

## **Documentation**

# **OTRS 5 - 系统管理员手册**

**Build Date:**

**2015-05-26**

---

---

## OTRS 5 - 系统管理员手册

版权 © 2003-2015 OTRS AG

本作品版权归OTRS AG所有。

只要您保留此版权声明，就可以拷贝文档。

All trade names are used without the guarantee for their free use and are possibly registered trade marks. All products mentioned in this manual may be trade marks of the respective manufacturer.

本文档的源代码可以在[github](#)上找到，具体位置是[doc-admin](#)，欢迎贡献更多内容。您也可以在[Transifex](#)上帮助翻译本文档为您的语言。



# 目录

前言 .....	xi
1. 介绍 .....	1
1. 问题单系统 — 基本介绍 .....	1
1.1. 问题单系统是什么，为何需要它？ .....	1
1.2. 什么事问题单？ .....	1
2. OTRS 服务台 .....	2
2.1. 基本介绍 .....	2
2.2. 功能介绍 .....	2
2.3. 硬件和软件要求 .....	7
2.4. 社区 .....	8
2.5. OTRS专业服务 .....	8
2. 安装 .....	9
1. 简单方法 - 预构建软件包安装 .....	9
1.1. 在SUSE Linux服务器上安装RPM包 .....	9
1.2. 在Red Hat Enterprise Linux或CentOS系统上安装OTRS .....	11
1.3. 在Debian或Ubuntu系统上安装OTRS .....	14
2. 从源码安装OTRS (Linux、Unix) .....	15
3. 使用Web安装器 .....	19
4. 在Windows上使用OTRS .....	24
4.1. 如何迁移现有的Windows安装到Linux .....	24
5. 升级OTRS版本 (从4到5) .....	35
6. 附加的应用程序 .....	41
6.1. FAQ 知识库 .....	41
3. OTRS第一步 .....	42
1. 服务人员WEB界面 .....	42
2. 客户WEB界面 .....	42
3. 公共WEB界面 .....	43
4. 首次登录 .....	43
5. WEB界面概览 .....	44
6. 仪表盘 .....	46
7. 一个队列是什么？ .....	50
8. 队列概览是什么？ .....	50
9. 用户偏好设置 .....	51
4. 管理 .....	54
1. OTRS系统管理区 .....	54
1.1. 基本介绍 .....	54
1.2. 服务人员、组和角色 .....	54
1.3. 客户与客户组 .....	61
1.4. 队列 .....	63
1.5. 问候语、签名、附件和模板 .....	64
1.6. 自动响应 .....	69
1.7. 系统邮件地址 .....	71
1.8. 工单通知 .....	72
1.9. S/MIME .....	74
1.10. PGP .....	75
1.11. 状态 .....	75
1.12. 系统配置 .....	76
1.13. 使用邮件帐户 .....	76
1.14. 过滤进入的邮件消息 .....	77
1.15. 使用GenericAgent (自动任务) 执行自动任务 .....	79
1.16. 管理员消息 .....	80
1.17. 会话管理 .....	81
1.18. 系统维护 .....	82
1.19. 系统日志 .....	83
1.20. 通过SQL查询窗口执行SQL查询 .....	83

1.21. 软件包管理器 .....	84
1.22. Web服务 .....	84
1.23. 动态字段 .....	85
2. 系统配置 .....	85
2.1. OTRS 配置文件 .....	85
2.2. 通过WEB界面配置系统 .....	86
3. 备份系统 .....	86
3.1. 备份 .....	86
3.2. 恢复 .....	87
4. 邮件设置 .....	88
4.1. 发送/接收邮件 .....	88
4.2. 用PGP加密邮件 .....	93
4.3. 用S/MIME加密邮件 .....	95
5. 使用外部后端 .....	99
5.1. 客户数据 .....	99
5.2. 客户联系人后端 .....	99
5.3. 服务人员和客户联系人身份认证后端 .....	106
5.4. 定制客户联系人自助注册 .....	110
6. 工单设置 .....	112
6.1. 工单状态 .....	112
6.2. 工单优先级 .....	115
6.3. 工单负责人 & 工单关注 .....	115
7. 时间相关的功能 .....	117
7.1. 设置上班时间、假期和时区 .....	117
7.2. 自动解锁 .....	118
8. 定制PDF输出 .....	119
9. 统计 .....	119
9.1. 统计配置和用法 .....	119
9.2. 统计系统管理员 .....	123
10. 动态字段 .....	124
10.1. 介绍 .....	124
10.2. 配置 .....	125
11. 通用接口 .....	140
11.1. 通用接口层 .....	140
11.2. 通用接口通讯流程 .....	142
11.3. Web服务 .....	145
11.4. WEB服务图形界面 .....	145
11.5. WEB服务命令行接口 .....	160
11.6. WEB服务配置 .....	160
11.7. 连接器 .....	166
12. OTRS守护进程 .....	182
12.1. OTRS守护进程图形界面 .....	183
12.2. OTRS守护进程命令行接口 .....	184
5. 定制 .....	186
1. 访问控制列表 (ACL) .....	186
1.1. 介绍 .....	186
1.2. 定义 .....	186
1.3. 示例 .....	187
1.4. ACL设置参考 .....	192
2. 流程管理 .....	195
2.1. 介绍 .....	195
2.2. 流程示例 .....	195
2.3. 实施示例 .....	196
2.4. 流程配置参考 .....	219
2.5. 导入样例流程 .....	240
3. 创建你自己的主题 .....	240
4. OTRS界面的本地化 .....	241
6. 性能优化 .....	242

1. OTRS .....	242
1.1. TicketIndexModule 工单索引模块 .....	242
1.2. SearchIndexModule .....	242
1.3. TicketStorageModule 工单存储模块 .....	243
1.4. Archiving Tickets .....	244
1.5. 缓存 .....	244
2. 数据库 .....	245
2.1. MySQL .....	245
2.2. PostgreSQL .....	245
3. WEB服务器 .....	245
3.1. 预建立的数据库连接 .....	246
3.2. 预装载的模块 - startup.pl .....	246
3.3. 当磁盘文件更新时重载Perl模块 .....	246
3.4. 选择正确的策略 .....	246
3.5. mod_gzip/mod_deflate .....	246
A. 其他资源 .....	247
B. 配置选项参考手册 .....	248
1. CloudService .....	248
2. Daemon .....	249
3. DynamicFields .....	260
4. Framework .....	264
5. GenericInterface .....	379
6. ProcessManagement .....	388
7. Ticket .....	397
C. GNU 自由文档许可证 .....	587
0. 导言 .....	587
1. 适用范围与约定 .....	587
2. 原样复制 .....	588
3. 大量发布 .....	588
4. 修正 .....	588
5. 合并文档 .....	589
6. 文档合集 .....	589
7. 独立作品汇集 .....	589
8. 翻译 .....	590
9. 协议终止 .....	590
10. 本协议的未来修订版 .....	590
你的文档怎样使用GNU自由文档许可证 .....	590

## 插图清单

2.1. 欢迎窗口 .....	20
2.2. GNU GENERAL PUBLIC LICENSE .....	20
2.3. 数据库选择 .....	21
2.4. 数据库认证 .....	21
2.5. 数据库设置 .....	22
2.6. 成功的数据库配置 .....	22
2.7. 系统设置 .....	23
2.8. 邮件配置 .....	23
2.9. WEB安装器的最后窗口 .....	24
2.10. 下载OTRSCloneDB - 屏幕截图 .....	25
2.11. 安装OTRSCloneDB - 屏幕截图 .....	26
2.12. 获取目标数据库的密码 - 屏幕截图 .....	28
2.13. 配置OTRSCloneDB的系统配置1 - 屏幕截图 .....	29
2.14. 配置OTRSCloneDB的系统配置2 - 屏幕截图 .....	30
2.15. 运行OTRSCloneDB脚本1 - 屏幕截图 .....	31
2.16. 运行OTRSCloneDB脚本2 - 屏幕截图 .....	32
2.17. 运行OTRSCloneDB脚本3 - 屏幕截图 .....	33
3.1. 服务人员界面登录窗口 .....	42
3.2. 客户界面登录窗口 .....	43
3.3. 公共WEB界面 .....	43
3.4. 请求新密码 .....	44
3.5. 服务人员界面仪表盘 .....	45
3.6. 页脚 .....	46
3.7. 仪表盘小部件 .....	46
3.8. 事件日历小部件 .....	48
3.9. 仪表盘设置 .....	49
3.10. 服务人员队列视图 (默认) .....	51
3.11. 服务人员队列视图的可视警告信号 .....	51
3.12. 服务人员的个人偏好设置 .....	52
3.13. 客户联系人的个人偏好设置 .....	53
4.1. OTRS系统管理概览窗口 .....	54
4.2. 服务人员管理 .....	55
4.3. 添加一个新服务人员 .....	55
4.4. 组管理 .....	56
4.5. 服务人员 <-> 组管理 .....	56
4.6. 修改一个服务人员所属的组 .....	57
4.7. 修改一个组关联的服务人员 .....	57
4.8. 角色管理 .....	59
4.9. 添加一个新的角色 .....	59
4.10. 修改一个服务人员关联的角色 .....	59
4.11. 修改一个特定角色关联的服务人员 .....	60
4.12. 管理角色-组的关系 .....	60
4.13. 修改一个角色的组关系 .....	60
4.14. 修改一个组的角色关系 .....	61
4.15. 客户管理 .....	61
4.16. 添加一个客户联系人 .....	62
4.17. 客户联系人-组 关系管理 .....	62
4.18. 修改一个客户联系人的组 .....	63
4.19. 修改一个组的客户联系人 .....	63
4.20. 队列管理 .....	63
4.21. 添加一个新队列 .....	64
4.22. 问候语管理 .....	65
4.23. 添加一条新的问候语 .....	65
4.24. 签名管理 .....	65
4.25. 添加一个新签名 .....	66

4.26. 附件管理 .....	66
4.27. 添加一个新附件 .....	66
4.28. 连接附件到模板中 .....	67
4.29. 修改一个模板的附件 .....	67
4.30. 修改一个附件关联的模板 .....	67
4.31. 模板管理 .....	68
4.32. 添加一个模板 .....	68
4.33. 模板-队列关系管理 .....	68
4.34. 修改一个模板的队列关系 .....	69
4.35. 修改一个队列的模板关系 .....	69
4.36. 自动响应管理 .....	69
4.37. 添加一个自动响应 .....	70
4.38. 队列<->自动响应 关系管理 .....	71
4.39. 修改一个队列的自动响应关系 .....	71
4.40. 系统邮件地址管理 .....	71
4.41. 添加一个系统邮件地址 .....	71
4.42. 工单通知管理 .....	72
4.43. 定制一个通知 .....	73
4.44. 定制一个通知的收件人 .....	73
4.45. 定制通知发送方法 .....	74
4.46. S/MIME管理 .....	75
4.47. PGP管理 .....	75
4.48. 状态管理 .....	75
4.49. 系统配置的图形界面 .....	76
4.50. 邮件账号管理 .....	77
4.51. 邮箱管理员过滤器管理 .....	77
4.52. 添加一个邮箱管理员过滤器 .....	79
4.53. GenericAgent (自动任务) 任务列表 .....	79
4.54. 创建一个新的GenericAgent (自动任务) 任务 .....	80
4.55. 管理员通知窗口 .....	81
4.56. 会话管理 .....	81
4.57. 会话详细信息 .....	82
4.58. 有一些计划周期的系统维护概览窗口 .....	82
4.59. 系统维护编辑窗口 .....	83
4.60. 系统日志 .....	83
4.61. SQL查询窗口 .....	84
4.62. 软件包管理器 .....	84
4.63. WEB服务的图形界面 .....	85
4.64. 有一些动态字段的动态字段概览窗口 .....	85
4.65. 系统配置的图形界面 .....	86
4.66. 新增一个邮件账号 .....	88
4.67. 在工单详情窗口变更一个工单的负责人 .....	116
4.68. 变更一个工单的负责人的弹出对话框 .....	116
4.69. 在一个工单的详情窗口添加关注 .....	117
4.70. 在一个工单的详情窗口取消关注 .....	117
4.71. “我关注的工单”视图 .....	117
4.72. 标准统计概览 .....	119
4.73. 查看一个特定的统计。 .....	120
4.74. 添加一个新的统计，第一步。 .....	120
4.75. 添加一个新的统计，第二步。 .....	121
4.76. 配置一个统计的X轴。 .....	122
4.77. 配置一个统计的Y轴。 .....	122
4.78. 配置一个统计的数据过滤器。 .....	122
4.79. 配置一个统计的数据过滤器。 .....	123
4.80. 导入统计 .....	123
4.81. Dynamic fields overview screen. ....	125
4.82. 文本动态字段配置对话框 .....	127
4.83. 多行文本动态字段配置对话框 .....	128

4.84. 复选框动态字段配置对话框 .....	128
4.85. 下拉列表框动态字段配置对话框 .....	129
4.86. 多选动态字段配置对话框 .....	130
4.87. 日期动态字段配置对话框 .....	131
4.88. 日期/时间动态字段配置对话框 .....	132
4.89. 填满样例数据的动态字段概览窗口 .....	133
4.90. 在新建电话工单窗口显示Field1 .....	134
4.91. 在新建电话工单窗口显示Field1为必填字段 .....	135
4.92. 在新建电话工单窗口显示数个字段为必填 .....	136
4.93. 在新建电话工单窗口不显示一些字段 .....	137
4.94. 在工单详情窗口显示Field1 .....	137
4.95. 在工单概览简洁模式视图显示Field1 .....	138
4.96. 在用户'偏好设置'中激活Field1 .....	140
4.97. 接口的'层'图解 .....	141
4.98. WEB服务概览 .....	145
4.99. 添加WEB服务 .....	146
4.100. 克隆WEB服务 .....	147
4.101. 导出WEB服务 .....	148
4.102. 导入WEB服务 .....	148
4.103. WEB服务历史 .....	149
4.104. 删除WEB服务 .....	150
4.105. WEB服务调试器 .....	151
4.106. 修改WEB服务 .....	152
4.107. WEB服务提供者网络传输 ( HTTP::SOAP ) .....	153
4.108. WEB服务提供者网络传输 ( HTTP::REST ) .....	154
4.109. WEB服务的操作 .....	155
4.110. WEB服务请求者的网络传输 ( HTTP::SOAP ) .....	156
4.111. WEB服务请求者网络传输 ( HTTP::REST ) .....	158
4.112. WEB服务调用程序 .....	159
4.113. WEB服务映射 .....	160
4.114. 守护进程通知 .....	183
4.115. 启动OTRS守护进程 .....	184
5.1. ACL 100-Example-ACL .....	188
5.2. ACL 102-Example-ACL .....	189
5.3. ACL 102-Second-Example-ACL .....	190
5.4. ACL 103-Third-ACL-Example .....	191
5.5. ACL 104-Only-Hardware-Services-for-HW-Queues-ACL .....	191
5.6. ACL 105-Disallow-Process-For-CustomerID .....	192
5.7. OTRS系统管理窗口 - 系统管理 .....	199
5.8. 创建新的流程按钮 .....	199
5.9. 添加新流程 .....	200
5.10. 创建新的活动对话框按钮 .....	200
5.11. 添加新的活动对话框 .....	201
5.12. 编辑字段详细信息(信件) .....	201
5.13. 创建新的转换按钮 .....	203
5.14. 创建新转换 .....	204
5.15. 创建新的转换操作按钮 .....	205
5.16. 创建新的转换操作 .....	206
5.17. 创建新活动按钮 .....	207
5.18. 将第一个活动拖放到画布中 .....	208
5.19. 拖动第二个活动到画布中 .....	208
5.20. 拖放一个转换到画布上 .....	209
5.21. 使用转换连接活动 .....	209
5.22. 分配转换操作 .....	210
5.23. 书籍采购完整的流程路径 .....	212
5.24. 导入样例流程小部件 .....	240
A.1. Bug 跟踪工具 .....	247



---

## 表格清单

4.1. 全新安装OTRS可用的默认组 .....	55
4.2. OTRS组相关的权限 .....	57
4.3. 附加的权限组 .....	58
4.4. 自动响应事件 .....	70
4.5. 不同X-OTRS-headers的功能 .....	77
4.6. 下列字段将被添加到系统中： .....	126
A.1. 邮件列表 .....	247



## 范例清单

4.1. 将垃圾邮件分类到一个指定队列 .....	79
4.2. 使用otrs.Console.pl通过Procmail路由 .....	89
4.3. .fetchmailrc .....	90
4.4. 过滤器模块Kernel::System::PostMaster::Filter::Match的任务示例 .....	90
4.5. 过滤器模块Kernel::System::PostMaster::Filter::CMD的任务示例 .....	91
4.6. 过滤器模 块Kernel::System::PostMaster::Filter::ExternalTicketNumberRecognition的 任务示例 .....	91
4.7. 配置客户联系人的数据库后端 .....	99
4.8. 通过数据库后端使用公司工单 .....	102
4.9. 配置一个LDAP客户后端 .....	102
4.10. 通过LDAP后端使用公司工单 .....	103
4.11. 在OTRS中利用多个客户数据后端 .....	104
4.12. 使用数据库后端认证服务人员 .....	106
4.13. 使用LDAP后端认证服务人员 .....	106
4.14. 使用HTTPBasic认证服务人员 .....	108
4.15. 用Radius后端认证服务人员 .....	108
4.16. 用数据库后端认证客户联系人 .....	108
4.17. 用LDAP后端认证客户联系人 .....	109
4.18. 为客户联系人配置HTTP基本认证 .....	109
4.19. 使用一个Radius后端的客户联系人身份认证 .....	110
4.20. 默认统计权限组 .....	121
4.21. 定制统计权限组 .....	121
4.22. 在新建电话工单窗口激活Field1。 .....	134
4.23. 在新建电话工单窗口激活Field1为必填字段。 .....	134
4.24. 在新建电话工单窗口激活数个字段。 .....	135
4.25. 在新建电话工单窗口去激活一些字段。 .....	136
4.26. 在工单详情窗口激活Field1。 .....	137
4.27. 在工单概览简洁模式视图激活Field1。 .....	138
4.28. 在工单创建事件激活Field1。 .....	138
4.29. 在用户'偏好设置'中激活Field1。 .....	139
4.30. 启动OTRS守护进程示例 .....	184
4.31. 列出所有配置的子守护进程的示例 .....	184
4.32. 汇总所有守护进程信息的示例 .....	185
5.1. 仅允许工单优先级5的工单转移到一个队列的ACL .....	187
5.2. ACL仅允许已保存到数据库中的优先级为5的工单移到到一个队列。 .....	188
5.3. ACL禁止关闭'Raw'队列的工单，并隐藏"关闭"按钮。 .....	189
5.4. ACL总是移除'成功关闭'状态。 .....	190
5.5. ACL为创建在以"HW"开头的队列中的工单仅显示硬件服务。 .....	191
5.6. ACL使用CustomerID禁用客户界面的一个流程 .....	192
5.7. 所有可能的重要ACL设置的参考 .....	193

# 前言

本书是为OTRS系统管理员准备的，也是OTRS新手的一本好参考书。

下面的章节描述了OTRS软件的安装、配置和管理。本书的前三分之一用来描述OTRS的主要功能，余下的部分是所有可配置参数的参考。

本书仍然是一个进展中的工作,正往新版本前进中。我们需要你的反馈以编制一本高质量的参考文档--可用、精确且完整的文档。如果你发现本书有内容缺失，或没有解释得足够清楚的事情，甚至发现了拼写错误、语法错误、错别字，都请写给我们。通过我们的BUG跟踪系统<http://bugs.otrs.org>提交任何反馈均高度赞赏。提前感谢你的贡献！



# 第 1 章 介绍

## 1. 问题单系统 — 基本介绍

本章会对问题单系统以及问题单系统的核心概念进行介绍。我们将通过一个虚拟案例来介绍系统的优点。

### 1.1. 问题单系统是什么，为何需要它？

下面的例子将叙述问题单系统是什么，公司将如何从这套系统中获益。

让我们假设Max是名录像机生产商。Max会收到许多来自客户咨关于备使用维修的咨询邮件。有时Max无法及时回复或查阅这些邮件。一些客户等的没了耐心，会重复发送邮件咨询相同问题。所有的支持请求邮件都保存在一个收件夹里。这些需求没有排序，所有邮件都由Max通过普通的邮件客户端回复。

由于Max无法足够快的回复所有的消息，由开发人员Joe和John来协助他处理邮件。Joe和John使用相同的邮件系统，访问同一个收件箱。他们并不知道Max经常会收到来自某个沮丧顾客发送的两封完全相同请求的邮件。还有有时候他们还会对同一个请求邮件各自做出自己的回复，客户则会收到两封不同回复的邮件。此外，Max无法知晓他们回复的细节。他也无法知道顾客问题和他们解决方案的细节，例如如哪些问题经常发生，或者他在客户支持上花掉了多少时间与金钱。

在一次会议上，一个同事告知Max，问题单系统可以帮助解客户支持的问题。在通过互联网了解相关信息后，Max决定在一台可以让他的顾客和雇员通过web访问的计算机上安装OTRS。现在，客户请求再也不用发送至Max的私人收件箱了，而是发送到OTRS使用的邮件账户。问题单管理系统将通过收件箱获取所有需求邮件，并将其转存至数据库。每个新请求，系统将生成自动回复发送给客户，这样客户知晓自己的需求已送达，服务人员会尽快被答复。OTRS系统会为每个请求生成唯一的ID号，即问题单号。现在，客户开心了，因为需求已经到达，只需耐心等待回复，也不用重复发送邮件了。Max，John和Joe则可以通过登录OTRS来回复客户请求。系统会锁定被回复过的问题单，这样就避免了重复回复。

想象一下，当Smith发送了一个请求邮件给Max的公司，被OTRS接收。John马上给了Smith一个简单回复，但是Smith反馈想要更加详细的解答。此时John刚好有别的事情要做，Max便回复了Smith的问题。历史查看功能使得Max可以很容易获知Smith的初始需求，以及John的回复内容，这样他可以补充更为详尽的解答信息。而Smith并不知道其实有两个人参与回复了此需求，而Max最后的回复，令他对问题得到解决而十分满意。

当然，这只是关于问题单系统的简单介绍。如果有大量客户请求（邮件和电话呼叫）需要响应，如果不同服务需要区分不同响应时间，问题单系统可以助力组织提高服务质量。问题单系统可以协助完善工作处理流程，提高效率并提高生产力。问题单系统可以协助组织创建并完善灵活的服务台支持环境。客户与服务人员之间的沟通会更透明。最终结果就是提高服务效率。毋庸置疑，让客户满意，意味着公司将获得更高的收益。

### 1.2. 什么事问题单？

问题单就像病人的病例。当病人第一次访问医院，医院会创建体检报告以保存他所有必要的个人和医疗信息。多次看病后，因为他可能由相同或多名的医生诊治过，主诊医生会更新通过增加病人的健康状况与持续治疗的相关信息来完善记录。这样一来，任何其他医生或护理人员都能获得一手完整的诊疗记录。当患者恢复并离开医院后，所有的医疗信息都会被归档，报告也会关闭。

诸如OTRS这样的问题单系统，通过邮件处理工单。所有消息都会保存在系统中。当客户发送请求，新问题单被系统自动创建，就像新开启病例一样。对新问题单的回复就像医生的诊断记录。当问题解决，系统会发送关闭通知给客户，或者由系统关闭工单。如果客户往已关闭的工单中添加内容，则该单会重新打开，新的信息被记录。所有的信息都会完整保存并归档。所有内容，如email、附件、注解也会作为每封email的附加信息保存。不仅如此，其他附加信息如日期、参与人员、工作量等也都有记录。在之后的任何阶段，工单可以排序归类，也可以通过搜索和过滤来分析所有信息。

## 2. OTRS 服务台

本章介绍OTRS Help Desk (OTRS) 的服务台功能。您将了解有关OTRS硬件与软件要求的信息。此外，本章中您还能了解如何获得OTRS的社区和商业支持。

### 2.1. 基本介绍

OTRS Help Desk (OTRS) 是一款安装在服务器上的网站应用程序，可以通过浏览器访问。

OTRS由几个部分组成。OTRS的主模块是OTRS框架，它包含了应用程序和工单系统的所有核心功能。还可以安装额外的应用，如ITSM模块，与网络监控解决方案集成，知识库（FAQ）等等。

### 2.2. 功能介绍

OTRS有许多功能。下面的列表给出了关于OTRS框架包含的主要功能概述。

#### 2.2.1. 用户界面

- OTRS针对服务人员和客户分别提供了独立、现代的Web界面。
- 其可以被用于任何现代网页浏览器，包括移动平台并支持视网膜屏分辨率。
- WEB界面还可以自定义主题和皮肤。
- 拥有功能强大、可定制的服务人员仪表盘，还有个人工单的总览与图形化的统计支持功能。
- 一个可扩展的报表引擎能提供多种统计和报告的调度计划选项。
- 使用流程管理(ProcessManagement)功能，可以定制基于工单的窗口和流程(工单 workflow)。
- OTRS内置有权限管理功能，可以配合细粒度的访问控制列表(ACL)扩展权限管理功能。
- 支持超过30种语言和不同时区。

#### 2.2.2. 邮件界面

- 支持带附件的MIME邮件
- 自动将HTML转化为普通文本信息(提高敏感内容安全性并提高搜索速度)。
- 新进邮件可以被过滤并按复杂规则预处理，如处理垃圾邮件或按队列分发。
- 支持PGP和S/MIME标准的密钥/证书管理和邮件处理。
- 自动响应，可以为每个队列单独设置
- 新工单、跟进与工单解锁可以邮件通知服务人员。
- 可以定义工单标识符，以识别跟进，如根据Call#、Ticket#或Request#。工单编号可以根据多种规则生成(基于时间和随机等。)，还可以按自己的要求进行设计。跟进同样可以通过In-Reference-To头部信息或扩展工单编号进行识别。

#### 2.2.3. 工单

- OTRS使用工单将内部与外部沟通信息整合在一起。这些工单以队列形式组织。
- 有多种方式查看系统中的工单(基于队列、状态、升级情况等)，还可以根据级别查看详细信息(简洁/基本/预览)。

- 工单历史记录一张工单的所有变更。
- 工单可以用很多方式修改，如回复、转发、退回、转换到另外一个队列、更新属性(状态、优先级等)、锁定和计算工时。可以对多张工单进行批量修改（批量操作）。
- 挂起时间与升级时间/SLA管理支持针对工单基于时间的调度与限制。
- 工单可以被链接到其他工单或其他对象如FAQ文档。
- 可以通过自动任务("GenericAgent")自动化与定时操作工单。
- OTRS拥有强大的搜索引擎，支持针对工单的复杂和全文搜索。

## 2.2.4. 系统

- OTRS能运行于多种操作系统（Linux, Solaris, AIX, FreeBSD, OpenBSD, Mac OS 10.x），并且后端支持多种数据库（MySQL, PostgreSQL, Oracle, MSSQL）。
- 可以通过安装OTRS软件包扩展核心系统。有很多免费软件包可以使用(如FAQ、OTRS::ITSM等)，OTRS团队同样还提供了商业付费软件包，有需要的话可以联系相关商务。
- 客户数据还支持与其他后端验证方式集成，如AD、eDirectory或OpenLDAP。客户可以通过数据库、LDAP、HTTPAuth或Radius进行验证。
- 使用通用接口可以很容易连接OTRS与其他的WEB服务。简单的WEB服务可以直接集成无须编码，复杂的场景，可以自定义扩展插件。OTRS工单连接器(The OTRS Ticket connector)允许第三方应用程序通过WEB服务创建、更新和搜索工单。

让我们看看最新几个版本的OTRS都有什么变化。

## 2.2.5. OTRS 5的新功能

### 2.2.5.1. 生产力

- OTRS现已针对不同类型和不同尺寸的移动设备进行了优化。
- 单选和多选输入字段更现代化，并提供高级搜索和过滤功能（感谢Mühlbauer的Dusan Vuckovic）。
- 现在能通过复制/粘贴和从任意位置拖放的方式（在所有浏览器中均可支持，无需额外插件）添加/上传图片到WYSIWYG（所见即所得）编辑器。
- 改进的工单通知系统。现在能够使用自己的触发器条件和收件人配置定制工单通知。在OTRS商业版（OTRS Business Solution™）中，还能通过短信和/或网页发送通知
- 统计有了一个新的图形界面，更好理解并有助于更快更容易地创建好的统计。
- 此外，统计支持新的时间周期“一刻钟”和“半年”。
- 现在支持工单详情窗口菜单项分组。使用频率低的菜单项被分组到子菜单中，以改进窗口使用和清晰度。
- 工单概览现在能显示客户单位信息，感谢 Renée Bäcker#
- 工单流程转换操作“TicketCreate（创建工单）”现在能创建没有信件的工单。

### 2.2.5.2. 可扩展性和性能

- 新的OTRS Daemon处理所有的异步和定期任务，替换了所有之前的OTRS cron任务。在群集环境中负载自动分配到所有节点。

- 现在可以指定多个只读镜像（从）数据库用于开销大的计算如统计或全文搜索，以在这些数据库中分配负载。

### 2.2.5.3. 安全

- 一种新的双因素身份验证层允许添加登录安全措施。
- 如果输入固定的用户名和密码不能满足您的需求，您现在能够使用开放标准基于时间的一次性密码（[RFC6238](#)，即谷歌身份验证器）。
- 在启用了双因素身份验证后，服务人员和客户能够添加一个共享密钥到他们的偏好设置中，并能使用他们选择的兼容方法（例如安卓手机上的谷歌身份验证器APP）生成的一次性密码立即开始登录系统。

### 2.2.5.4. 与外部系统协同工作

- 一个新的基于XSLT的通用接口映射模块允许任意复杂的用户定义数据映射。

### 2.2.5.5. 安装和管理

- 新的OTRS控制台使得命令行操作简单而有趣。所有的命令拥有一致的接口、实用的文档，并提供有用的彩色输出结果。
- 系统管理员现在能够指定最低的日志级别，以减少日志容量，感谢Renée Bäcker#
- 系统管理概览窗口现在显示无效条目为灰色，更容易将注意力集中在活动元素上。

## 2.2.6. OTRS 4新功能

### 2.2.6.1. 生产力

- 实现了一种更清晰的平面设计。
- 服务人员现在能够直接回复到一个工单备注，原始备注的正文内容在新的备注中引用。
- 服务人员现在能够在具有内部备注的所有窗口使用模板。
- 工单操作窗口（如备注、所有者等）现在允许执行操作而不用始终创建新信件（可配置）。
- 基于“我的服务”（服务人员能够关注的服务）的新的工单概览。新工单和跟进工单的通知选项现在能基于“我的队列”、“我的服务”或两者组合来配置。
- OTRS现在能显示多达数千信件的工单。
- 仪表板中的客户在线列表现在直接链接到客户信息中心页面。
- 服务人员现在能够通过拖放永久地重排主菜单。
- 服务人员和客户现在能够通过附件名搜索工单。
- 新的仪表板小部件--执行中的流程工单。
- 新的搜索选项--工单的最后修改时间。
- 为没有回复的工单添加了新的外出邮件窗口。

### 2.2.6.2. 可扩展性和性能

- OTRS 4在相同的硬件配置下能够处理更多的并发用户/请求，单个请求的响应时间更短，特别是有大量数据的页面。

### 2.2.6.3. 与其他系统协同工作。

- 通用接口现在还支持HTTP REST 作为网络传输协议。

### 2.2.6.4. 安装和管理

- 邮件管理员过滤器不再限制4条匹配字段，可以有大量可配置的字段的（默认为12，最大99）。
- 新的配置选项Ticket::MergeDynamicFields可以指定在合并一个工单到另一工单时哪个动态字段也将合并。
- 添加了新的选项来检查动态字段输入的文本，如果不匹配相关的模式则显示翻译了的错误消息。
- 添加了新的选项来限制动态字段输入将来或过去的日期/日期时间。
- 在添加了信件或所有者不在办公室时，OTRS能够配置自动解锁一个工单。
- 特定类型的链接工单（例如合并或已删除）现在能够通过系统配置选项隐藏起来。
- 改进了的ACL处理更加一致且更容易调试。
  - 添加了新的ACL选项-PossibleAdd（可能添加），添加条目到可能的列表而不重置。
  - 添加了新的ACL值修改器，[Not-非]、 [NotRegExp-非正则表达式]、 [Notregexp-非正则表达式]，适用于所有ACL。
- 改进了的流程处理更加一致且更容易调试。
  - OTRS流程配置新的基于GUID（全球唯一标识符）的实体命名方案，可以将流程从一个系统安全地传输到另一系统，而不用复制实体。
  - 添加了新的转换操作来创建一个新工单。
  - 添加了定义可变转换操作属性（基于当前流程工单值）的可能性。
- 从管理员界面的系统管理面板可以计划系统维护期了。
  - 系统维护期到来前的某个时间（可以配置）会显示一条通知消息。
  - 如果在系统维护期内，在服务人员和客户界面会显示一条通知消息，并且只有系统管理员能够登录到系统。
  - 一个关于活动会话的概览窗口，系统管理员可以一键结束所有会话或逐个结束会话。
- 通过配置可以禁止导入系统配置。
- 添加了Apache MD5作为一个新的哈希密码后端，感谢Norihiro Tanaka。
- 添加了客户自注册的邮件地址白名单或黑名单，感谢Renée Bäcker#
- 添加了新的仪表板模板来显示外部命令的结果，感谢ib.pl。

### 2.2.6.5. 开发

- 基于Template::Toolkit的新的强力模板引擎。
- 一个新的中央对象管理器使得创建和使用全局对象更容易，感谢noris network公司的Moritz Lenz。
- 扩展了OPM软件包格式，增加了一个软件包已被合并到另一软件包的标志，允许软件包管理器在安装或升级时能正确处理这种情形。
- 缓存被集中到一个全局缓存对象，也执行所有数据的内存缓存。



- 添加了缓存速度测试脚本，感谢ib.pl。

## 2.2.7. OTRS 3.3的新功能

### 2.2.7.1. 生产力

- 仪表盘工单列表和定期工单概览现在可以用合适的工单列来过滤，并可以配置要显示的列。
- 工单基本概览和预览概览视图现在可以排序了。
- 仪表盘新增了日历小部件，可以将工单作为事件展示。
- 仪表盘新增了小部件，在矩阵表单中按状态和按队列显示工单数。
- 服务人员现在可以标记重要信件。
- 新的树形选择小部件使操作树形数据(选择队列、服务等)更快速、更简单。
- 在日期和日期/时间动态字段中支持搜索相对日期(如“一个月之前”)。
- 现在还可以在创建新工单或工单转发时指定模板（之前的“标准响应”）。
- 可用流程列表现在可以通过ACL过滤。
- 支持从客户界面发起初始流程。
- 许多地方的文本不再使用缩短了的固定数量字符串(如“Queue1...”),而是占用可用的屏幕空间，以看到更多的信息。
- OTRS已支持视网膜分辨率。图片资源已经为了支持更高分辨率而优化，大部分图像图标都已经利用来自FontAwesome的图标字体替换。
- 新增仪表盘管理功能，可以让仪表盘显示状态图。请注意IE8不支持该功能。

### 2.2.7.2. 与其他系统协同工作。

- OTRS现在可以支持多个客户单位数据库，感谢belnet-ict公司的Cyrille。
- OTRS现在可以用工单动态字段自动存储客户联系人数据以便永久存储到工单中，这些数据在报表中 useful。
- OTRS现在可以根据来自外部系统的工单编号，正确地将进入邮件分配到已有的工单。
- OTRS现在支持通过POP3/TLS连接收取邮件。

### 2.2.7.3. 安装和管理

- Web安装程序在原有MySQL数据库之外，还新增了PostgreSQL、Oracle与SQL Server数据库支持。
- OTRS现在已完全支持MySQL 5.6。
- 现在可以按配置的工单事件触发执行自动任务。
- 新的图形化ACL编辑器让编辑ACL更简单。
- 邮件管理员过滤器现在可使用否定条件，感谢Renée Bäcker#
- 邮件管理员过滤器现在可基于入站邮件，设置相关的挂起日期、所有者与负责人。
- 客户与服务人员密码现在支持强壮的bcrypt算法加密，比SHA算法更优秀。

- 许多图标都已改成图标字体，方便基于不同颜色自定义皮肤。通过加载更小尺寸的图片文件还改善了整体性能。

## 2.3. 硬件和软件要求

OTRS can be installed on many different operating systems. OTRS can run on linux and on other unix derivates (e.g. OpenBSD or FreeBSD). OTRS does not have excessive hardware requirements. We recommend using a machine with at least a 3 GHz Xeon or comparable CPU, 8 GB RAM, and a 256 GB hard drive.

运行OTRS同样还需要使用web服务器和数据库服务器。除此之外，还需要在OTRS主机上安装perl以及其他相关的perl模块。web服务器与Perl必须安装在运行OTRS的主机上。数据库后端可以安装在本地，也可以独立安装在另外主机上。

对于web服务器，我们建议使用Apache HTTP Server，因为可以利用mod\_perl模块极大地提升OTRS性能。除此之外，OTRS可以在任何支持执行Perl脚本的web服务器上运行。

你可以用不同的数据库部署OTRS，可以选择MySQL、PostgreSQL或Oracle。如果你使用MySQL或PostgreSQL，就有通过WEB界面在安装过程中配置数据库和某些系统设置的优势。

对于Perl，你需要一些额外的模块，这些模块可以通过Perl shell和CPAN安装，也可以通过操作系统的软件包管理器（rpm、yast、apt-get）安装。

软件要求

### 2.3.1. Perl支持

- Perl 5.10或更高版本

### 2.3.2. Web服务器支持

- Apache2 + mod\_perl2或更高版本(建议)
- 支持CGI的Web服务器(不建议使用CGI)

### 2.3.3. 数据库支持

- MySQL 5.0 or higher (MySQL 8+ is [not supported](#))
- MariaDB
- PostgreSQL 9.0 or higher
- Oraclr 10g或更高版本

本手册的安装额外的perl模块部分有如何安装OTRS所需Perl模块的详细介绍。

如果使用为你的操作系统编译的二进制程序包安装OTRS，要么是包含所有所需Perl模块的软件包，要么是操作系统的软件包管理器，均需要注意Perl模块的依赖性。

### 2.3.4. 浏览器支持

To use OTRS, you'll be OK if you use a modern browser with JavaScript support enabled. Supported browsers are:

- Google Chrome
- Firefox version 10 and higher
- Safari version 5 and higher

- Internet Explorer version 10 and higher, Microsoft Edge

我们推荐你的浏览器保持更新。新版本的浏览器一直在改进JavaScript和渲染的性能。使用老版本浏览器时大型系统会出现戏剧性的性能问题。我们乐于在此问题上为您提供咨询。

## 2.4. 社区

OTRS拥有一个大型的用户社区。用户和开发人员通过邮件列表讨论OTRS并交换相关问题的信息。你可以使用邮件列表来讨论OTRS的安装、配置、使用、本地化和开发方面的问题。你也可以通过我们的bug跟踪系统报告软件BUG。

OTRS社区主页是：<http://www.otrs.com/open-source/>。

## 2.5. OTRS专业服务

我们的OTRS Business Solution™（OTRS商业版）为您提供来自OTRS团队最佳的专业支持、可靠的安全和定期自由更新，以及**独家的附加商业功能集**，你可以根据不同的部署场景灵活地激活或去激活某些功能。

**OTRS集团** 为不同国家提供专业的 **培训项目**。可以加入我们定期举行的OTRS系统管理员培训，或者联系我们安排根据需求定制的公司内部培训。

## 第 2 章 安装

本章描述了OTRS中央框架的安装和基本配置，涵盖了从源码或从二进制软件包如RPM安装OTRS的信息。

这里涵盖的主题包括WEB和数据库服务器的配置、OTRS和数据库之间的接口、附件的Perl模块的安装、设置合适的OTRS访问权限、设置OTRS的cron任务以及OTRS配置文件的一些基本设置。

跟随本章的详细步骤来安装OTRS到你的服务器中，然后你就可以使用它的WEB界面来登录并管理系统了。

### 1. 简单方法 - 预构建软件包安装

如果你使用的系统平台有OTRS预构建软件包，你就应该优先用它来安装，因为这是最简单、最便捷的方法。你可以在官网下载区[www.otrs.com](http://www.otrs.com)找到可用的预构建软件包。

#### 1.1. 在SUSE Linux服务器上安装RPM包

本部分描述了在SUSE Linux服务器上安装RPM包。

##### 1.1.1. 准备OTRS所需的数据库环境

你可以使用不同的数据库后端：MySQL、PostgreSQL或Oracle。部署OTRS最流行的数据库是MySQL。本节描述了在基于SUSE的服务器上配置MySQL需要的步骤。当然如果你有伸缩性或其它方面的需要，也可以将数据库安装到一台单独的数据库服务器上。

##### 注意

如果你在openSUSE 12.3以后的版本参照本节的步骤，实际上你不是安装MySQL而是MariaDB，一个与MySQL代码兼容的MySQL分支。这并没有任何问题，它可以同样工作得很好（在某些点上甚至更好那么一点点）。

以root用户执行下列的命令来安装MySQL：

```
linux:~ # zypper install mysql perl-DBD-mysql
```

这会按默认选项安装MySQL到系统中。你需要修改一些默认设置以便更适合OTRS运行。用文本编辑器打开文件/etc/my.cnf并添加下列行到[mysqld]节：

```
max_allowed_packet = 20M
query_cache_size = 32M
innodb_log_file_size = 256M
```

##### 注意

Please note that OTRS requires utf8 as database storage encoding.

现在执行命令**systemctl restart mysql.service**来重启数据库服务器并激活刚才的修改内容。然后运行命令**/usr/bin/mysql\_secure\_installation**并根据窗口提示来设置数据库的root用户密码、移除匿名访问和移除test数据库。最后，运行命令**systemctl enable mysql.service**来确保MySQL在服务器启动后自动运行。

##### 1.1.2. 安装 OTRS

使用**zypper**通过命令行来安装OTRS，它还会拉入一些依赖包如Apache WEB服务器和一些Perl模块。确保你已经将OTRS RPM文件复制到了当前目录。

```
otrs-sles:~ # zypper install otrs*.rpm
....
Retrieving package otrs-x.x.x-01.noarch (1/26), 17.5 MiB (74.3 MiB unpacked)
Installing: otrs-x.x.x-01 [done]
Additional rpm output:
Check OTRS user ... otrs added.

...
otrs-sles:~ #
```

现在使用命令 **systemctl restart apache2.service** 重启 Apache 以便载入为 OTRS 修改的配置。

### 1.1.3. 安装额外的 perl 模块

OTRS 每次通过软件包管理器默认安装不会安装所有需要的模块，你可以在后面手动安装它们。运行位于目录 `/opt/otrs/bin/` 下的脚本 `otrs.CheckModules.pl`，可以知道缺失了哪些必须安装的模块。可选的模块可能包括通过 IMAP(S) 与 MTA 通讯或生成 PDF 格式的输出。

在 SLES 操作系统中需要添加外部软件仓库来获得缺失的模块。从这里选择符合操作系统版本的软件仓库：<http://download.opensuse.org/repositories/devel:/languages:/perl/>。例如，SLES 11 SP 3 的软件仓库应按如下方式添加：

```
zypper ar -f -n perl http://download.opensuse.org/repositories/devel:/languages:/perl/
SLE_11_SP3_Perl
```

在 openSUSE 12.3 中只有 `Mail::IMAPClient` 才需要额外的软件仓库，这个模块只有当你要从用 TLS 加密的 IMAP 邮件服务器收取邮件时才需要。相应的命令行像这个样子：

```
zypper ar -f -n perl http://download.opensuse.org/repositories/devel:/languages:/perl/
openSUSE_12.3/ Perl
```

在添加了这个软件仓库后首次使用 `zypper` 时会提示添加它的密钥。现在你就可以像下面这样安装缺失的模块。

```
otrs-sles:/opt/otrs # zypper install -y "perl(YAML::LibYAML)"
Refreshing service 'susecloud'.
Retrieving repository 'perl' metadata [\]

New repository or package signing key received:
Key ID: DCCA98DDDEF338C
Key Name: devel:languages:perl OBS Project <devel:languages:perl@build.opensuse.org>;
Key Fingerprint: 36F0AC0BCA9D8AF2871703C5DCCA98DDDEF338C
Key Created: Wed Oct 10 22:04:18 2012
Key Expires: Fri Dec 19 22:04:18 2014
Repository: perl

Do you want to reject the key, trust temporarily, or trust always? [r/t/a/?] (r): a
Retrieving repository 'perl' metadata [done]
Building repository 'perl' cache [done]
Loading repository data...
Reading installed packages...
'perl(YAML::LibYAML)' not found in package names. Trying capabilities.
Resolving package dependencies...

The following NEW package is going to be installed:
 perl-YAML-LibYAML

The following package is not supported by its vendor:
```

```
perl-YAML-LibYAML
```

```
Retrieving package perl-YAML-LibYAML-0.38-12.4.x86_64 (1/1), 75.0 KiB (196.0 KiB unpacked)
Retrieving: perl-YAML-LibYAML-0.38-12.4.x86_64.rpm [done (55.7 KiB/s)]
Installing: perl-YAML-LibYAML-0.38-12.4 [done]
```

下一步是使用WEB安装器来配置OTRS，其过程本章另有描述。

现在你可以启动OTRS守护进程并激活相应的监控此进程的cron任务（必须以otrs用户执行）：

```
shell> /opt/otrs/bin/otrs.Daemon.pl start
shell> /opt/otrs/bin/Cron.sh start
```

就这样了，恭喜搞定！

## 1.2. 在Red Hat Enterprise Linux或CentOS系统上安装OTRS

本部分描述了在一台Red Hat Enterprise Linux (RHEL) 或 CentOS 服务器上RPM软件包的安装。

### 1.2.1. 准备：禁用SELinux

#### 注意

如果你的系统使用了SELinux，你需要禁用它，否则OTRS无法正常工作。

下面是如何在RHEL/CentOS/Fedora中禁用SELinux：

- 在文件/etc/selinux/config中配置SELINUX=disabled

```
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#     enforcing - SELinux security policy is enforced.
#     permissive - SELinux prints warnings instead of enforcing.
#     disabled - No SELinux policy is loaded.
SELINUX=disabled
# SELINUXTYPE= can take one of these two values:
#     targeted - Targeted processes are protected,
#     mls - Multi Level Security protection.
SELINUXTYPE=targeted
```

- 重启系统。重启后确认命令getenforce返回为Disabled：

```
shell> getenforce
Disabled
```

### 1.2.2. 准备OTRS所需的数据库环境

你可以使用不同的数据库后端：MySQL、PostgreSQL或Oracle。部署OTRS最流行的数据库是MySQL。本节描述了在基于RHEL的服务器上配置MySQL需要的步骤。当然如果你有伸缩性或其它方面的需要，也可以将数据库安装到一台单独的数据库服务器上。

以root用户执行下列的命令来安装MySQL（或MariaDB）：

```
shell> yum -y install mariadb-server
```

这会在你的系统上使用默认选项安装MySQL，你需要修改默认设置以适用于OTRS。使用文本编辑器来创建一个新文件/etc/my.cnf.d/zotrs.cnf，包含如下内容：

```
[mysqld]
max_allowed_packet = 20M
query_cache_size = 32M
innodb_log_file_size = 256M
```

现在执行**systemctl start mariadb**来重启数据库服务器并激活刚才的修改内容。然后运行命令**usr/bin/mysql\_secure\_installation**，并按照屏幕上的指令来设置数据库的root密码、移除匿名访问及删除test数据库。

## 注意

Please note that OTRS requires utf8 as database storage encoding.

### 1.2.3. 安装 OTRS

使用**yum**通过命令行来安装OTRS，它还会拉入一些依赖包如Apache WEB服务器和一些Perl模块。确保你已经将OTRS RPM文件复制到了当前目录。

```
shell> yum install --nogpgcheck otrs-x.x.*.rpm
...
Dependencies Resolved

=====
Package                Arch   Version                Repository              Size
=====
Installing:
otrs                   noarch x.x.x-01               /otrs-x.x.x-01.noarch 74 M
Installing for dependencies:
apr                    x86_64 1.3.9-5.el6_2         updates                 123 k
...
procmail               x86_64 3.22-25.1.el6         base                    163 k

Transaction Summary
=====
Install      26 Package(s)

Total size: 80 M
Total download size: 6.0 M
Installed size: 88 M
Downloading Packages:
(1/25): apr-1.3.9-5.el6_2.x86_64.rpm           | 123 kB   00:00
...
(25/25): procmail-3.22-25.1.el6.x86_64.rpm     | 163 kB   00:00
-----
Total                                           887 kB/s | 6.0 MB   00:06
Running rpm_check_debug
Running Transaction Test
Transaction Test Succeeded
Running Transaction
  Installing : apr-1.3.9-5.el6_2.x86_64         1/26
  ...
  Installing : otrs-x.x.x-01.noarch             26/26
Check OTRS user ... otrs added.
...
shell>
```

现在使用命令 `systemctl restart httpd.service` 重启 Apache 以载入为 OTRS 修改的配置。

## 1.2.4. 安装额外的 perl 模块

除了通过 RPM 包安装的 Perl 模块外，OTRS 还需要一些其它的 Perl 模块，你可以手动安装。通过执行位于目录 `/opt/otrs` 下的文件 `bin/otrs.CheckModules.pl` 来检查缺失的模块。一些模块只是可选的功能才需要，比如与 IMAP 服务器通讯或生成 PDF。在 Red Hat 或 CentOS 上我们推荐从 EPEL 软件仓库安装这些模块，EPEL 是由 Fedora 项目维护的一个软件仓库，为 RHEL 及其分支提供高质量的软件包。可在 [EPEL 网站](#) 上查看更多信息。

如果你使用的是 RHEL 7 或 CentOS 7，你可以从 [这个网址](#) 获得最新的 EPEL 软件包。你可以执行下面的命令添加 EPEL 软件仓库到 yum 源：

```
shell> yum -y install http://download.fedoraproject.org/pub/epel/7/x86_64/e/epel-release-7-8.noarch.rpm
...
Installed:
  epel-release.noarch 0:7-8
Complete!
```

添加这个软件仓库后首次使用 yum，会提示你要添加它的密钥。现在你可以像下面这样安装缺失的模块了。

```
shell> yum -y install "perl(Text::CSV_XS)"
...
Installed:
  perl-Text-CSV_XS.x86_64 0:0.85-1.el6
Complete!
shell>
```

下一步是使用 WEB 安装器来配置 OTRS，其过程本章另有描述。

现在你可以启动 OTRS 守护进程并激活相应的监控此进程的 cron 任务（必须以 `otrs` 用户执行）：

```
shell> /opt/otrs/bin/otrs.Daemon.pl start
shell> /opt/otrs/bin/Cron.sh start
```

就这样了，恭喜搞定！

## 1.2.5. 在 Red Hat 或 CentOS 上安装 Oracle 数据库驱动

如果你想在 ORACLE 数据库上部署 OTRS，你需要编译并安装 `DBD::Oracle` 数据库驱动。这比安装其它软件稍微复杂了一点儿；这是因为 Oracle 是一个私有数据库，Red Hat 或 CentOS 项目都不允许在他们的 RPM 软件仓库中分发这个驱动。

首先，我们需要安装 `gcc`、`make` 和 `CPAN`，这样我们才能编译和安装这个驱动。你下面看到的命令适用于 CentOS，其它的版本可能有一点点不同。



```
shell> yum -y install gcc make "perl(CPAN)"
```

下一步是获取并安装数据库客户端。你需要到Oracle官网注册一个免费帐号，从网页：<http://www.oracle.com/technetwork/database/features/instant-client/index-097480.html>上可以下载客户端驱动。请选择对应的'Linux x86 or x86-64'版本。你可以用命令**uname -i**检查架构，要么是'x86\_64'，要么是'i386'即x86。你应该下载软件包'Instant Client Package - Basic'、'Instant Client Package - SQL\*Plus'和'Instant Client Package - SDK'。将它们保存到磁盘的一个目录。现在用下面的命令来安装这些软件包：

```
shell> yum install oracle-instantclient*
```

安装完成后你要设置两个环境变量并编译DBD::Oracle驱动。还是以root用户执行这些任务。步骤如下。请注意：命令输出结果被省略了一些行。

```
shell> export ORACLE_HOME=/usr/lib/oracle/11.2/client64
shell> export LD_LIBRARY_PATH=$ORACLE_HOME/lib
shell> cpan
cpan[1]> look DBD::Oracle
...
Fetching with LWP:
  http://www.perl.org/CPAN/authors/id/P/PY/PYTHIAN/CHECKSUMS
Checksum for /root/.cpan/sources/authors/id/P/PY/PYTHIAN/DBD-Oracle-1.62.tar.gz ok
Scanning cache /root/.cpan/build for sizes
DONE
...
Working directory is /root/.cpan/build/DBD-Oracle-1.62-ZH6LNy
[root@localhost DBD-Oracle-1.62-ZH6LNy]# perl Makefile.PL
...
[root@localhost DBD-Oracle-1.62-ZH6LNy]# make
...
[root@localhost DBD-Oracle-1.62-ZH6LNy]# make install
...
cpan[2]> exit
Terminal does not support GetHistory.
Lockfile removed.
```

现在就该编辑文件Kernel/Config.pm来提供环境变量ORACLE\_HOME了。下一步是使用WEB安装器来配置OTRS，其过程本章另有描述。

## 注意

Please note that OTRS requires utf8 as database storage encoding.

## 1.3. 在Debian或Ubuntu系统上安装OTRS

### 重要

请从源码安装OTRS，不要使用Debian或Ubuntu提供的OTRS软件包。

如果你使用现成的软件来安装Perl模块会更容易一些：

```
apt-get install libapache2-mod-perl2 libdbd-mysql-perl libtimedate-perl libnet-dns-perl \
libnet-ldap-perl \
libio-socket-ssl-perl libpdf-api2-perl libdbd-mysql-perl libsoap-lite-perl libtext-csv- \
xs-perl \
libjson-xs-perl libapache-dbi-perl libxml-libxml-perl libxml-libxslt-perl libyaml-perl \
libarchive-zip-perl libcrypt-eksblowfish-perl libencode-hanextra-perl libmail- \
imapclient-perl \
```

## 2. 从源码安装OTRS ( Linux、 Unix )

### 2.1. 准备：禁用SELinux

#### 注意

如果你的系统使用了SELinux，你需要禁用它，否则OTRS无法正常工作。

下面是如何在RHEL/CentOS/Fedora中禁用SELinux：

- 在文件/etc/selinux/config中配置SELINUX=disabled

```
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#     enforcing - SELinux security policy is enforced.
#     permissive - SELinux prints warnings instead of enforcing.
#     disabled - No SELinux policy is loaded.
SELINUX=disabled
# SELINUXTYPE= can take one of these two values:
#     targeted - Targeted processes are protected,
#     mls - Multi Level Security protection.
SELINUXTYPE=targeted
```

- 重启系统。重启后确认命令getenforce返回为Disabled：

```
shell> getenforce
Disabled
```

### 2.2. 步骤1：安装 .tar.gz

如果你要从源码安装OTRS，首先从<https://www.otrs.com/download-open-source-help-desk-software-otrs-free/>下载.tar.gz、.tar.bz2或.zip格式的源文件。

解包这个档案文件（例如使用**tar**）到/opt目录，并将目录otrs-x.x.x 重命名为otrs（脚本如下）。

```
shell> tar xzf /tmp/otrs-x.x.x.tar.gz
shell> mv otrs-x.x.x /opt/otrs
```

### 2.3. 步骤2：安装附加的Perl模块

使用下面的脚本来得到所有已安装的和必需的CPAN模块概览。

```
shell> perl /opt/otrs/bin/otrs.CheckModules.pl
o CGI.....ok (v3.60)
o Crypt::PasswdMD5.....ok (v1.3)
o Crypt::SSLeay.....Not installed! (Optional - Required for Generic Interface
SOAP SSL connections.)
o CSS::Minifier.....ok (v0.01)
o Date::Format.....ok (v2.22)
o Date::Pcalc.....ok (v1.2)
...
```

## 注意

请注意：OTRS需要一个安装了所有核心模块（如version）、能工作的Perl。这些模块不在脚本的检查范围，在某些系统上（如RHEL）默认没有安装这些核心模块，你可能需要安装一个perl-core软件包。

要安装缺失的Perl模块，你可以：

### 2.3.1. a) 通过你的Linux分支的软件包管理器来安装需要的软件包

- 对RedHat、CentOS、Fedora或兼容的系统：

```
shell> yum install "perl(Digest::MD5)"
```

- 对SUSE Linux Enterprise Server、openSUSE或兼容的系统：首先确定模块的软件包名称。通常模块My::Module的软件包叫做“perl-My-Module”。

```
shell> zypper search Digest::MD5
```

然后安装：

```
shell> zypper install perl-Digest-MD5
```

- 对Debian、Ubuntu或兼容的系统：首先确定模块的软件包名称。通常模块My::Module的软件包叫做“libmy-module-perl”。

```
shell> apt-cache search Digest::MD5
```

然后安装：

```
shell> apt-get install libdigest-md5-perl
```

请注意：你可能在你的Linux分支的软件仓库中找不到所有的模块或模块所需的版本，在这种情况下你只能选择通过CPAN安装这些模块（如下）。

### 2.3.2. b) 通过CPAN Shell安装需要的Perl模块

注意：如果是在Linux系统上，你要用超级用户（root）运行CPAN shell，因为这些模块需要由OTRS帐户和WEB服务器运行时使用的帐户共同访问。

```
shell> perl -MCPAN -e shell;
...
install Digest::MD5
install Crypt::PasswdMD5
...
```

如果系统的某些特殊功能依赖于可选的模块，则脚本中列出的可选模块也应该安装上。

## 2.4. 步骤3：创建OTRS用户

创建用户：

```
shell> useradd -d /opt/otrs -c 'OTRS user' otrs
```

将用户添加到WEB服务器所有组（如果WEB服务器不是以OTRS用户运行）：

```
shell> usermod -G www otrs  
(SUSE=www, Red Hat/CentOS/Fedora=apache, Debian/Ubuntu=www-data)
```

## 2.5. Step 4: Activate Default Config File

There is one OTRS config file bundled in \$OTRS\_HOME/Kernel/Config.pm.dist. You must activate it by copying it without the ".dist" filename extension.

```
shell> cp /opt/otrs/Kernel/Config.pm.dist /opt/otrs/Kernel/Config.pm
```

## 2.6. 步骤5：检查所有需要的模块是否都已安装

```
shell> perl -cw /opt/otrs/bin/cgi-bin/index.pl  
/opt/otrs/bin/cgi-bin/index.pl syntax OK  
  
shell> perl -cw /opt/otrs/bin/cgi-bin/customer.pl  
/opt/otrs/bin/cgi-bin/customer.pl syntax OK  
  
shell> perl -cw /opt/otrs/bin/otrs.Console.pl  
/opt/otrs/bin/otrs.Console.pl syntax OK
```

“syntax OK”表示所有必须的模块都已安装。

## 2.7. 步骤6：配置Apache WEB服务器

首先，你得安装Apache2 WEB服务器和mod\_perl模块；一般都通过系统的软件包管理器来安装。下面你可以找到主流Linux分支安装Apache需要的命令。

```
# rhel / centos:  
shell> yum install httpd mod_perl  
  
# suse:  
shell> zypper install apache2-mod_perl  
  
# debian/ubuntu:  
shell> apt-get install apache2 libapache2-mod-perl2
```

大部分的Apache安装都会包括一个conf.d目录。在Linux系统中这个目录一般在/etc/apache或/etc/apache2目录下。以root用户登录，进入conf.d目录，并在Apache配置目录创建一个到/opt/otrs/scripts/apache2-httpd.include.conf的软链接zzz\_otrs.conf（确保载入其它配置之后再载入OTRS配置）。

OTRS需要激活一些Apache模块来达到最佳运行状态。在大多数平台上你可以通过a2enmod工具来确认已经激活了这些模块。

```
shell> a2enmod perl  
shell> a2enmod version  
shell> a2enmod deflate  
shell> a2enmod filter  
shell> a2enmod headers
```

现在你可以重启WEB服务器来载入新的配置设置了。在大多数系统你可以用**systemctl restart apache2.service**命令来重启Apache。

## 2.8. 步骤7：文件权限

文件权限根据需要调整，以允许OTRS读取和写入文件：

```
otrs.SetPermissions.pl [ --otrs-user=OTRS用户，默认为'otrs' ] { --web-group= WEB服务器用户所属的组 }
```

例如：

- 以OTRS用户运行WEB服务器：

```
shell> bin/otrs.SetPermissions.pl --web-user=otrs
```

- 以wwwrun用户运行WEB服务器（如SUSE）：

```
shell> bin/otrs.SetPermissions.pl --web-group=wwwrun
```

- 以apache用户运行WEB服务器（如Red Hat、CentOS）：

```
shell> bin/otrs.SetPermissions.pl --web-group=apache
```

- 以www-data用户运行WEB服务器（如RDebian、Ubuntu）：

```
shell> bin/otrs.SetPermissions.pl --web-group=www-data
```

## 2.9. 步骤8：数据库设置和系统基本配置

请使用WEB安装器 <http://localhost/otrs/installer.pl>（用你的OTRS主机名或IP地址替换“localhost”）来设置数据库和系统基本配置如邮件帐户。

### 注意

下面的推荐配置设置是针对MySQL的。请添加下列行到文件 `/etc/my.cnf` 的[mysqld]部分：

```
max_allowed_packet = 20M
query_cache_size = 32M
innodb_log_file_size = 256M
```

### 注意

Please note that OTRS requires utf8 as database storage encoding.

## 2.10. 步骤9：首次登录

现在你可以在 <http://localhost/otrs/index.pl>登录系统了，使用在WEB安装程序中配置的认证信息（用户：root@localhost）。

通过本步骤，完成了系统的基本配置。

## 2.11. 步骤10：启动OTRS守护进程

新的OTRS守护进程负责处理OTRS中的所有异步和循环任务。之前在cron文件中定义的自动任务现在都由OTRS守护进程处理。这个守护进程还处理所有的'GenericAgent自动任务'任务，它必须以otrs用户运行。

```
shell> /opt/otrs/bin/otrs.Daemon.pl start
```

## 2.12. 步骤11：OTRS用户的Cron任务

OTRS自带了两个不同的cron文件：`/opt/otrs/var/cron/*.dist`，它们的目的是确保OTRS守护正常运行。你需要将它们复制为不带.dist扩展名的文件名来激活它们。

```
shell> cd /opt/otrs/var/cron
shell> for foo in *.dist; do cp $foo `basename $foo .dist`; done
```

要在系统中安排这些cron任务，你可以使用otrs运行脚本文件Cron.sh。

```
shell> /opt/otrs/bin/Cron.sh start
```

还可以用这个脚本来停止cron任务（系统维护时有用）：

```
shell> /opt/otrs/bin/Cron.sh stop
```

## 2.13. 步骤12：设置bash自动补齐（可选）

所有的常规OTRS命令操作现在都通过OTRS控制台接口 `bin/otrs.Console.pl` 执行。它提供bash的自动补齐功能，在查找正确的命令和选项时更容易些。

你可以安装软件包**bash-completion**来激活bash的自动补齐功能。它能为otrs用户自动检测到并载入文件`/opt/otrs/.bash_completion`。

重启shell后，你可以在输入`bin/otrs.Console.pl`后按TAB键，就会显示所有可用的命令。如果你键入了命令的部分字符，按TAB键会列出所有匹配这些字符的命令。在输入完整的命令后，按TAB键会列出所有可用的选项和参数。

## 2.14. 步骤13：进一步信息

我们建议你参阅OTRS性能调优章节。

如果安装遇到了问题，你可以给我们的邮件列表`otrs@otrs.org`(<http://lists.otrs.org/>)发送一个消息。

你还可以请求OTRS集团帮助你规划或部署OTRS，或者检查你安装的OTRS系统。我们的**专业服务**就是用来帮助你更快地部署OTRS并从OTRS获得最大的收益。

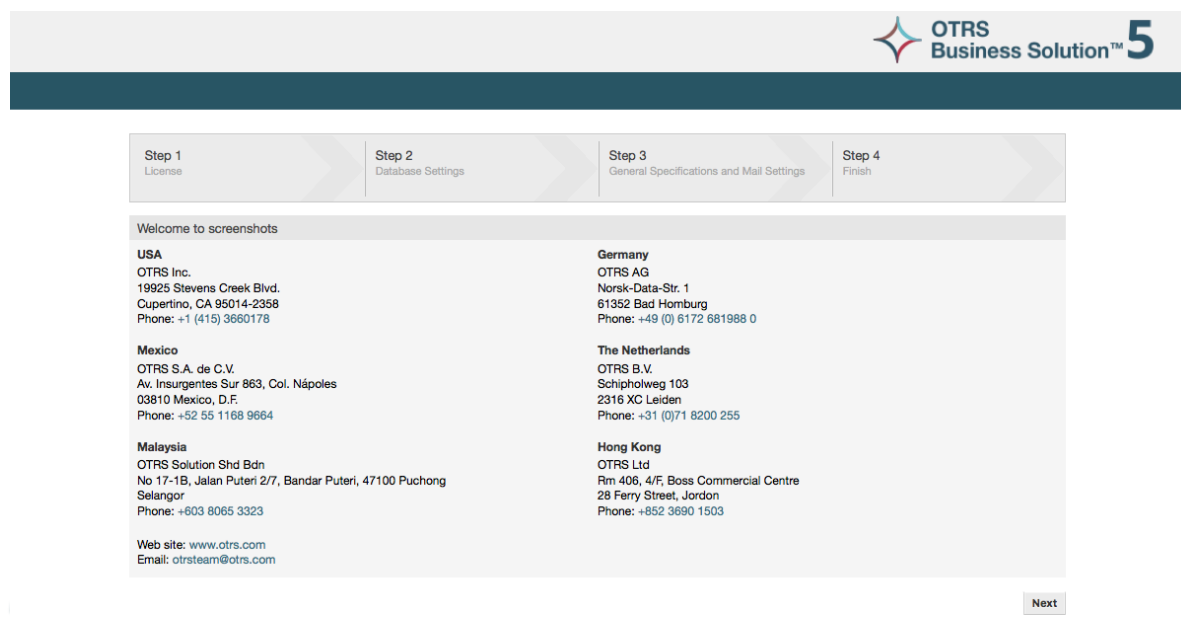
## 3. 使用Web安装器

在安装完OTRS软件后，你可以使用OTRS的WEB安装器来设置和配置OTRS数据库。WEB安装器是一个能通过浏览器访问的WEB页面。WEB安装器的地址是：<http://localhost/otrs/installer.pl>。

启动WEB安装器后，请跟随下面的步骤来设置你的系统：

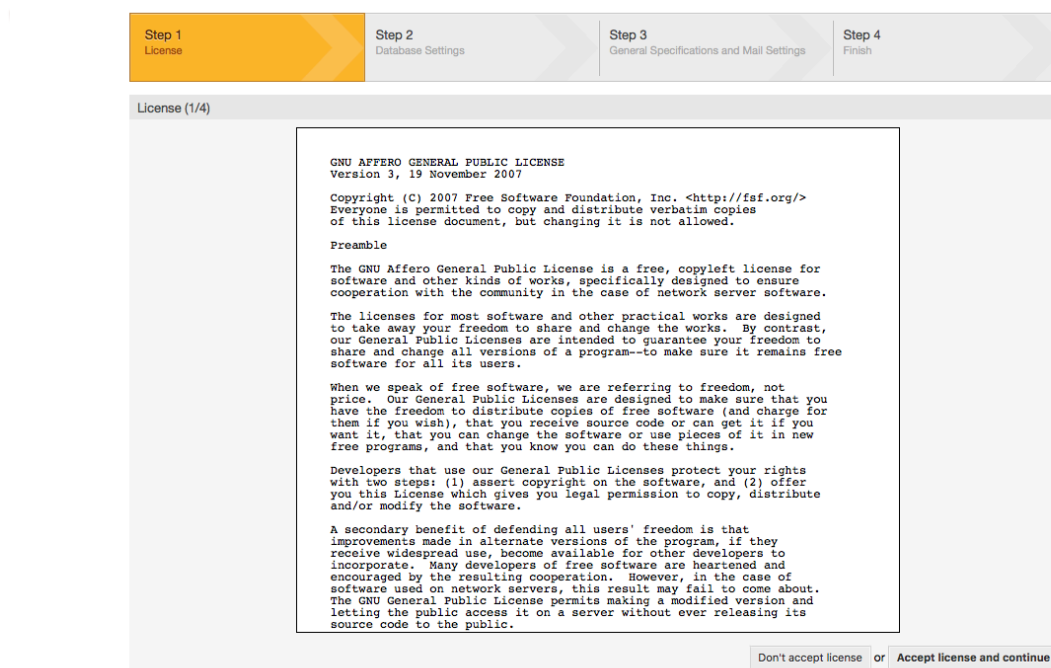
1、检验OTRS办公室信息并点击‘下一步’以继续（见下图）。

图 2.1. 欢迎窗口



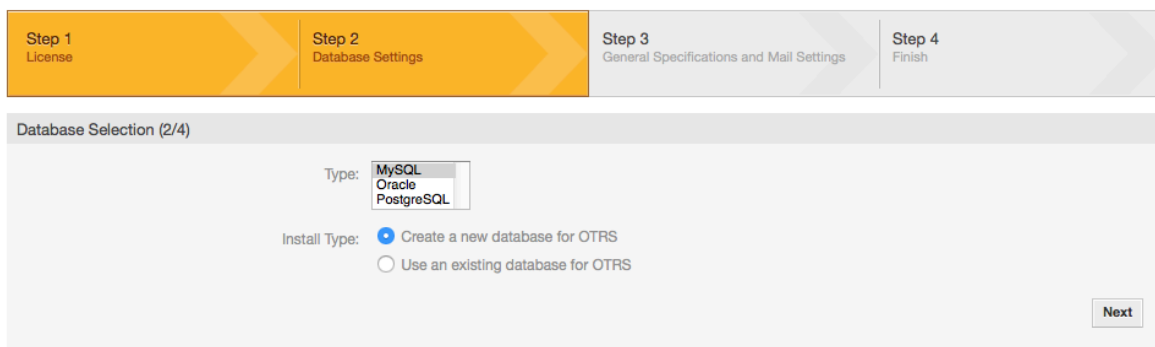
2. Read the GNU GENERAL PUBLIC LICENSE (see figure below) and accept it, by clicking the corresponding button at the bottom of the page.

图 2.2. GNU GENERAL PUBLIC LICENSE



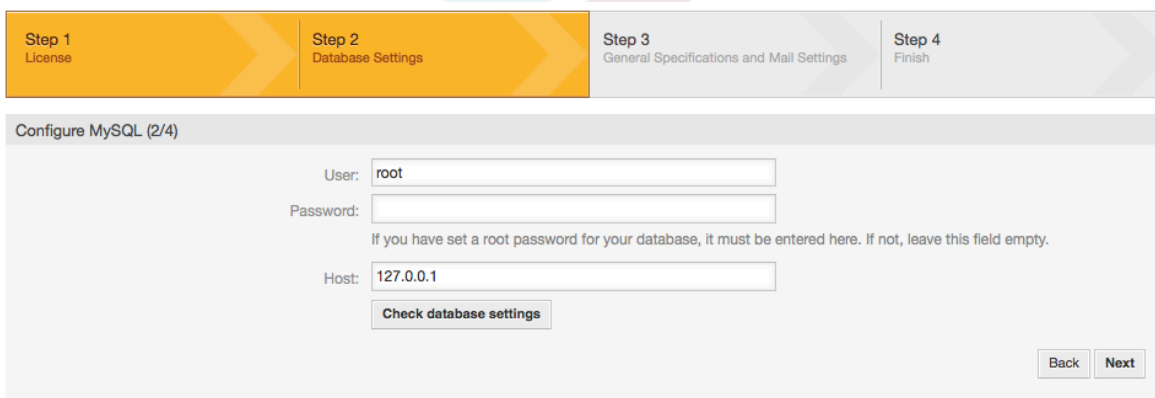
3、选择你要在OTRS中使用的数据库。如果你选择MySQL或PostgreSQL，你还能在这里选择是通过WEB安装器新建一个数据库还是使用你的数据库管理员已经创建好的空数据库。

图 2.3. 数据库选择



4、根据你选择的数据库的不同，以及在上一步中是用WEB安装器新建数据库还是使用已有数据库，这个窗口可用有一点点差异。在这个窗口输入数据库认证信息。

图 2.4. 数据库认证



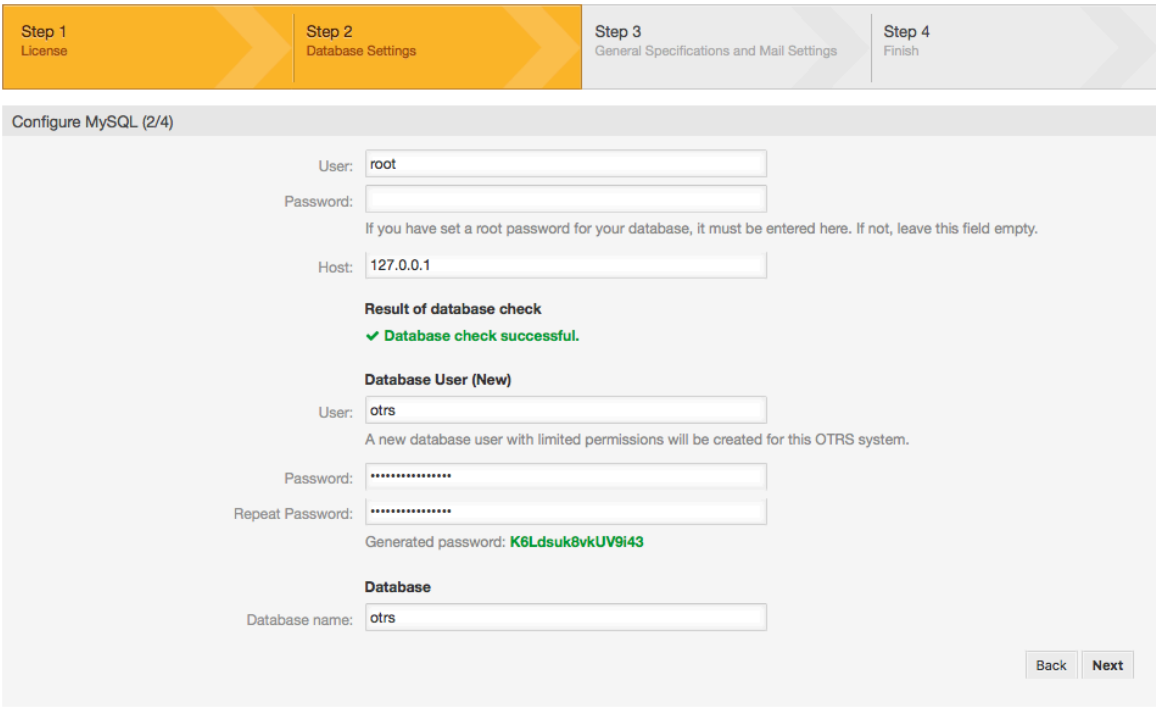
5、创建一个新的数据库用户，选择一个数据库名称，并点击‘下一步’（见下图）。

## 警告

OTRS会为你生成一个强密码，当然如果你愿意也可以输入你自己的密码。这个密码会写入到配置文件Kernel/Config.pm，所以无需记住这个密码。



图 2.5. 数据库设置



Step 1 License | **Step 2 Database Settings** | Step 3 General Specifications and Mail Settings | Step 4 Finish

Configure MySQL (2/4)

User:

Password:

If you have set a root password for your database, it must be entered here. If not, leave this field empty.

Host:

**Result of database check**  
✓ Database check successful.

**Database User (New)**

User:

A new database user with limited permissions will be created for this OTRS system.

Password:

Repeat Password:

Generated password: **K6Ldsuk8vkUV9i43**

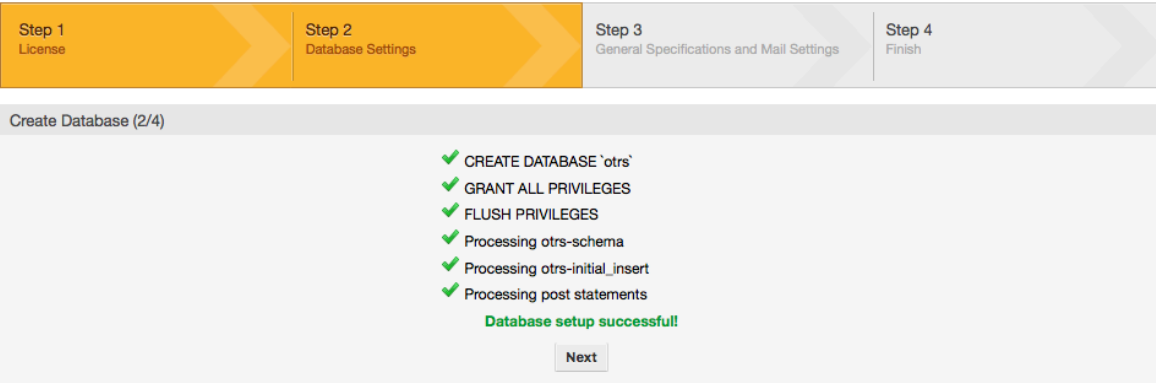
**Database**

Database name:

Back Next

6、如果需要会创建数据库，并填充相应数据，如图所示。点击‘下一步’进入下一个窗口。

图 2.6. 成功的数据库配置



Step 1 License | Step 2 Database Settings | **Step 3 General Specifications and Mail Settings** | Step 4 Finish

Create Database (2/4)

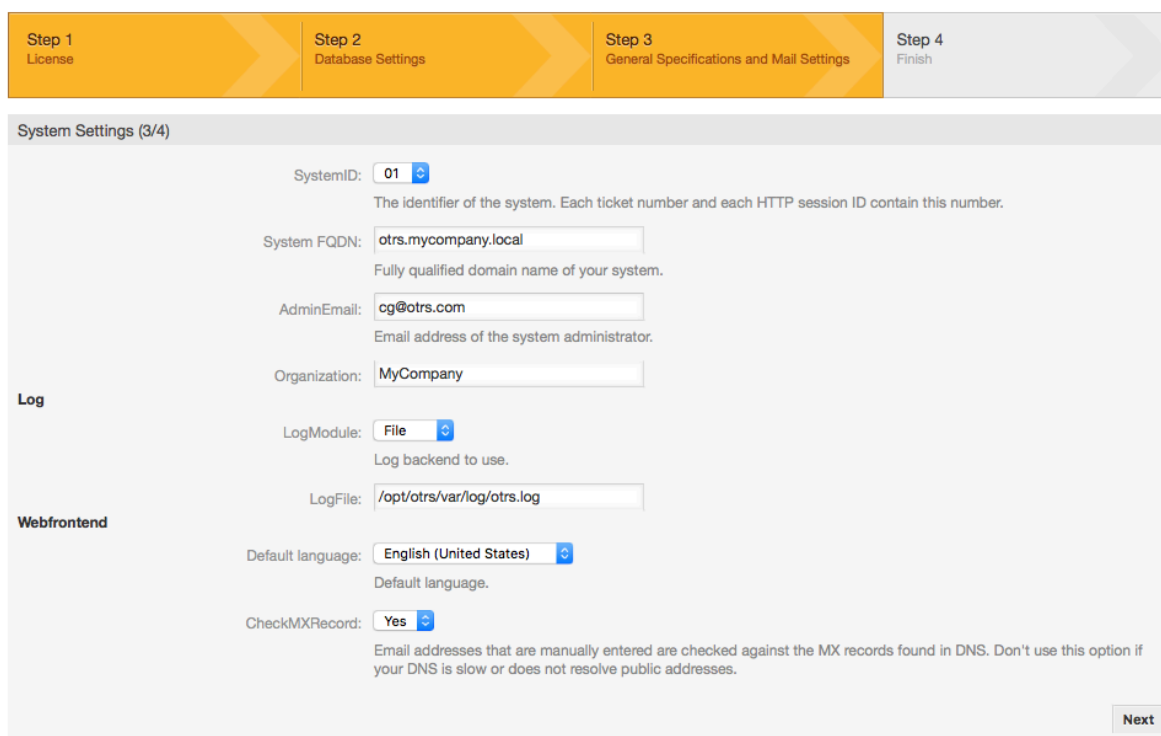
- ✓ CREATE DATABASE `otrs`
- ✓ GRANT ALL PRIVILEGES
- ✓ FLUSH PRIVILEGES
- ✓ Processing otrs-schema
- ✓ Processing otrs-initial\_insert
- ✓ Processing post statements

**Database setup successful!**

Next

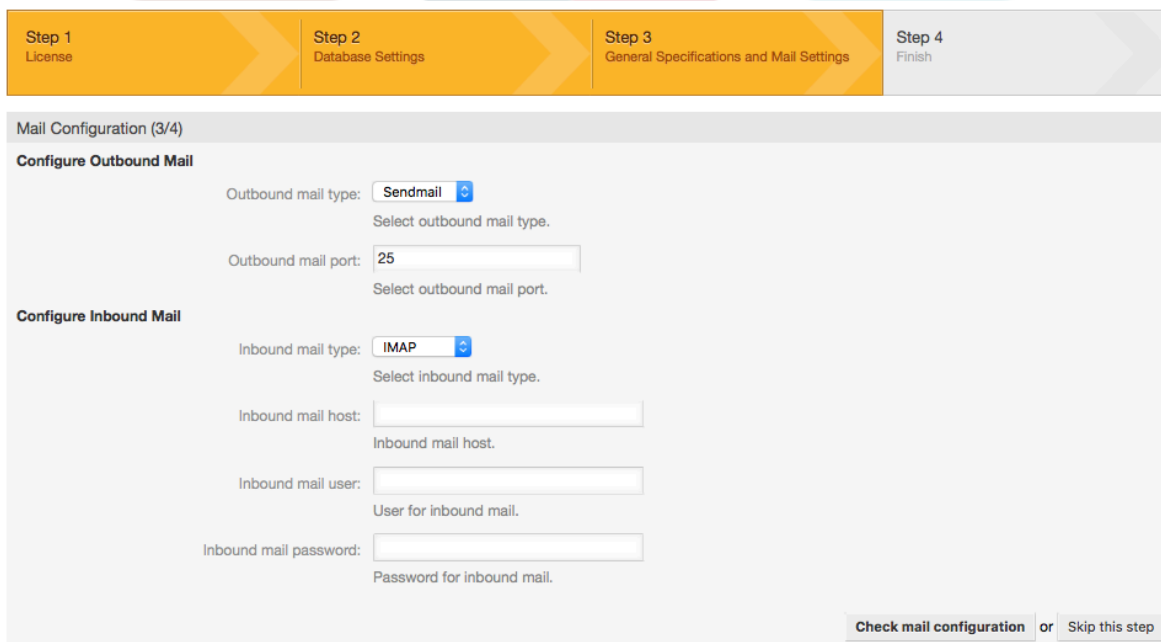
7、提供所有必填的系统设置，并点击‘下一步’（如下图）。

图 2.7. 系统设置



8、若需要，你可以提供需要的数据来配置收发邮件，或者点击窗口底部右边的按钮跳过这一步（如下图）。

图 2.8. 邮件配置

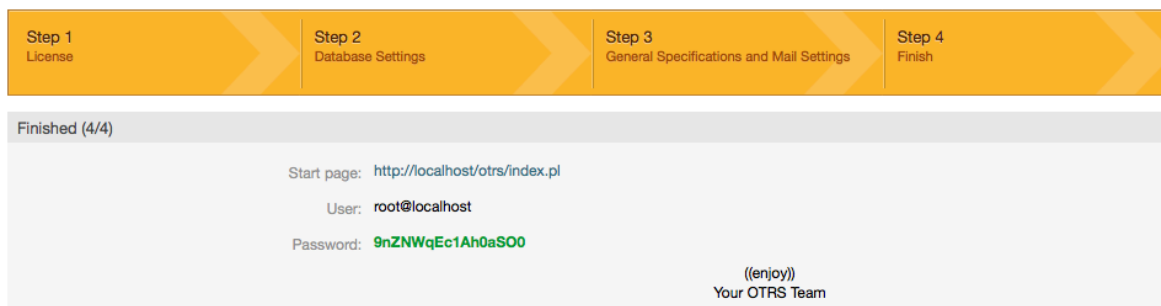


9、恭喜！现在OTRS的安装全部完成，你可以用这个系统工作了（见下图）。在浏览器中输入地址<http://localhost/otrs/index.pl>即可登录到OTRS WEB界面。使用用户名'root@localhost'和刚生成的密码以OTRS系统管理员登录，然后你就可以配置这个系统来满足你的需求了。

## 警告

请记住刚生成的'root@localhost'帐户密码。

图 2.9. WEB安装器的最后窗口



## 4. 在Windows上使用OTRS

OTRS可以在各种不同的系统平台上运行，包括企业级Linux平台如OTRS on Windows和SUSE Linux Enterprise Server及一系列其它的Linux分支。

然而在Windows平台上运行OTRS时我们遭遇了反复的性能损失，尽管做过详尽的分析，但是由于技术的差异，好像不可能解决这些性能问题到令人满意的程度。由于目前缺少其他供应商提供的必要的第三方组件，因此带着一颗沉重的心,我们已经停止开发Windows安装程序和OTRS Appliance (OTRS一体机，部署在虚拟机中无需安装可直接使用的OTRS系统)。

在这些情况下，我们不能保证OTRS能持续地运行于Windows平台，所以推荐将OTRS迁移到上面提及的Linux平台或使用托管的**OTRS Business Solution™** (OTRS商业版)

为了更容易地将OTRS从Windows迁移到Linux以获得最佳性能，我们在这里为你准备了详细的说明。

### 4.1. 如何迁移现有的Windows安装到Linux

#### 4.1.1. 说明和准备

如果你有了一个基于Windows的OTRS安装并想迁移到Linux系统，你需要安装一台Linux服务器或虚拟主机并在系统中安装OTRS (参阅安装说明章节)，这将是迁移的目标系统。

#### 4.1.2. 取得OTRSCloneDB脚本以克隆数据库

进入Windows版的系统管理菜单，并安装最新版的OTRSCloneDB软件包到OTRS：

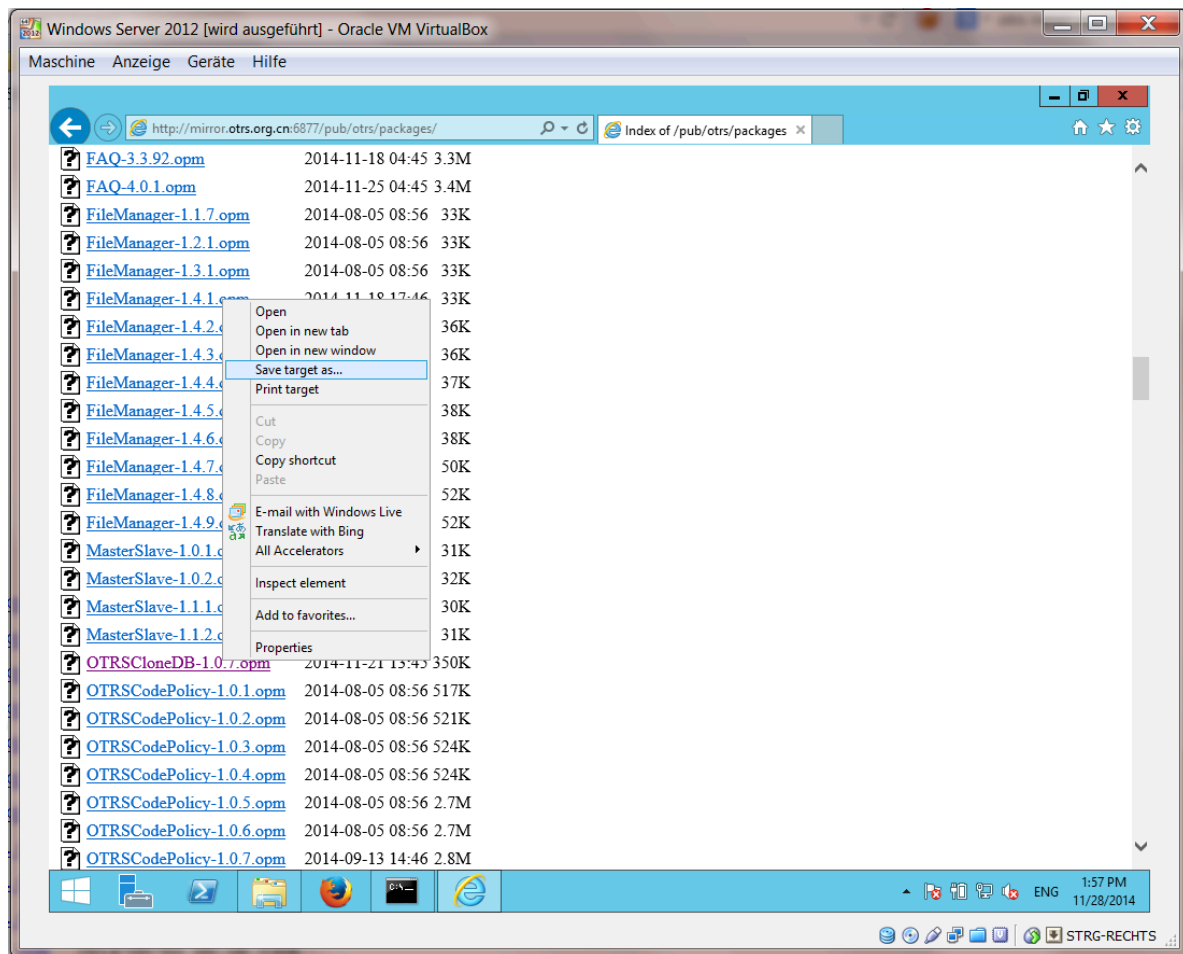
你可以从软件包管理器中直接安装OTRSCloneDB软件包。从左侧的下拉列表中选择“OTRS Extensions”并点击“更新软件仓库信息”按钮，然后会在一个列表中显示OTRSCloneDB软件包，点击“安装”即可。

你还可以从OTRS的FTP服务器上手动下载这个软件包，并用下面的方法手动安装这个软件包。<http://ftp.otrs.org/pub/otrs/packages/>

请下载最高版本号的软件包：

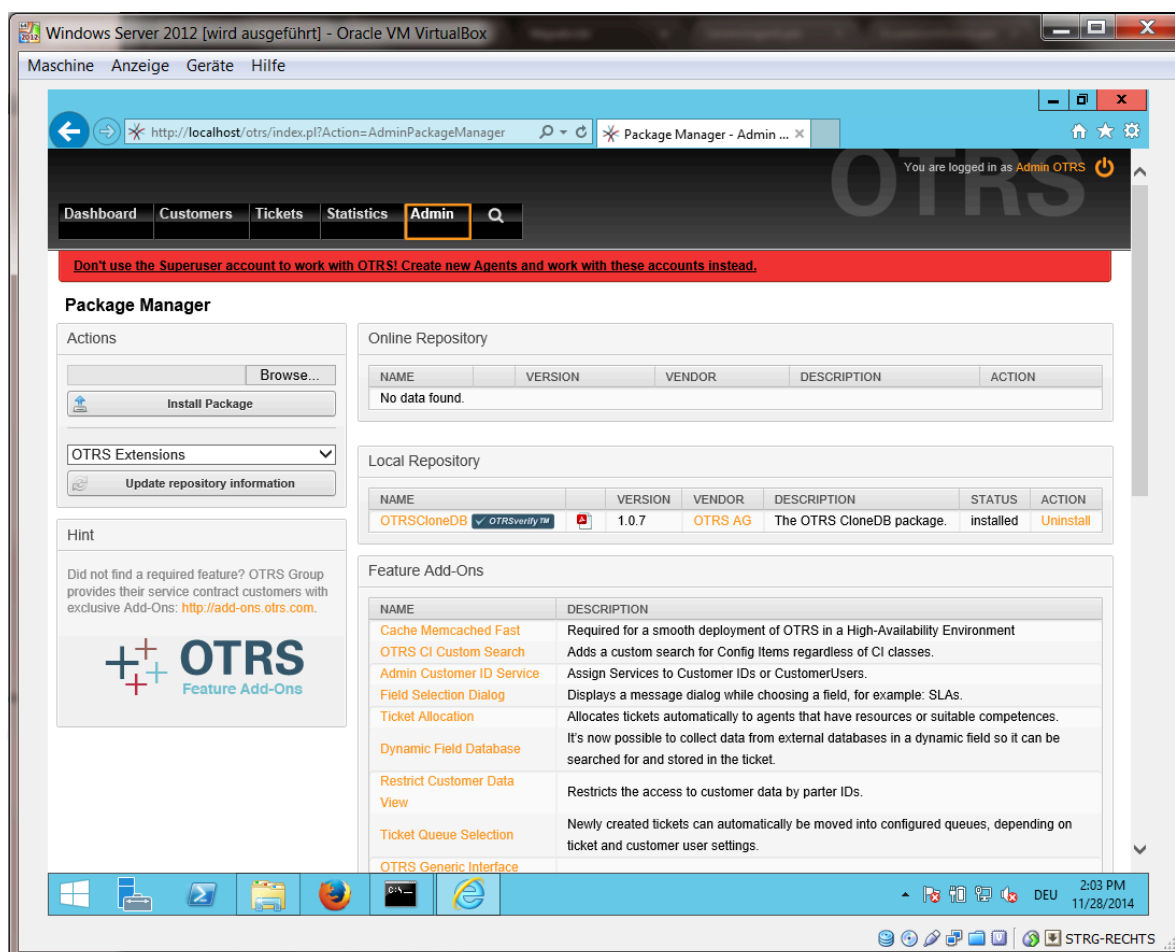
```
OTRSCloneDB-1.0.13.opm
```

图 2.10. 下载OTRSCloneDB - 屏幕截图



安装到Windows版的OTRS中：

图 2.11. 安装OTRSCloneDB - 屏幕截图



如果你安装了一些附加功能或对OTRS进行了定制开发也没有问题。你只需注意所有安装的软件也能与Unix类的系统兼容就可以了，由OTRS提供的软件包就是这样。

### 4.1.3. 允许目标系统PostgreSQL数据库的远程访问

OTRSCloneDB脚本通过网络复制数据库的数据，所以我们需要允许到数据库的远程访问。不同的数据库设置方法不一样，我们在这里描述的是打开PostgreSQL数据库的远程访问。

通过SSH登录到目标系统后，切换到postgresql目录：

```
shell> cd /etc/postgresql/9.4/main
shell> vi postgresql.conf
```

将下列行添加到文件末尾：

```
listen_addresses = '*'
```

保存文件

```
shell> vi pg_hba.conf
```

将下列行添加到文件末尾：

```
host all all 0.0.0.0/0 md5
```

保存文件

重启PostgreSQL服务器

```
shell> service postgresql restart
```

#### 4.1.4. 停止OTRS服务

停止目标系统所有运行的服务：

```
shell> service cron stop
shell> service apache2 stop
shell> su - otrs
shell> cd /opt/otrs/
shell> bin/Cron.sh stop
shell> bin/otrs.Daemon.pl -a stop
shell> exit
```

#### 4.1.5. 删除目标系统现有的数据库，为克隆数据腾空数据库

OTRSCloneDB脚本不会移除目标系统存在的otrs数据库的数据，所以我们需要手动删除：

切换到postgresql用户：

```
shell> su - postgres
```

删除已有的otrs数据库：

```
shell> dropdb otrs
```

为otrs用户创建一个新的otrs数据库：

```
shell> createdb --owner=otrs --encoding=utf8 otrs
```

返回到root用户：

```
shell> exit
```

#### 4.1.6. 获取数据库的PostgreSQL密码

进入目标系统的OTRS目录：

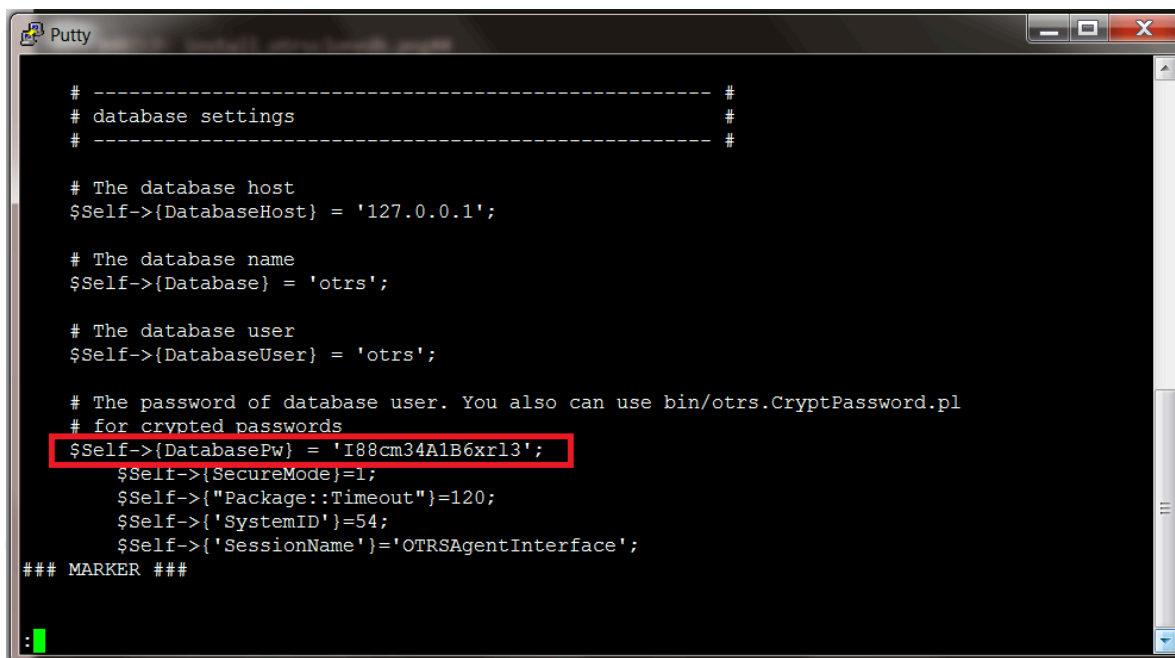
```
shell> cd /opt/otrs
```

查看目录系统的配置文件：

```
shell> less Kernel/Config.pm
```

你往下拉一点就能找到你的数据库密码：

图 2.12. 获取目标数据库的密码 - 屏幕截图



```

# ----- #
# database settings #
# ----- #

# The database host
$Self->{DatabaseHost} = '127.0.0.1';

# The database name
$Self->{Database} = 'otrs';

# The database user
$Self->{DatabaseUser} = 'otrs';

# The password of database user. You also can use bin/otrs.CryptPassword.pl
# for crypted passwords
$Self->{DatabasePw} = 'I88cm34A1B6xrl3';
$Self->{SecureMode}=1;
$Self->{"Package::Timeout"}=120;
$Self->{'SystemID'}=54;
$Self->{'SessionName'}='OTRSagentInterface';
### MARKER ###

```

在我们的示例中：

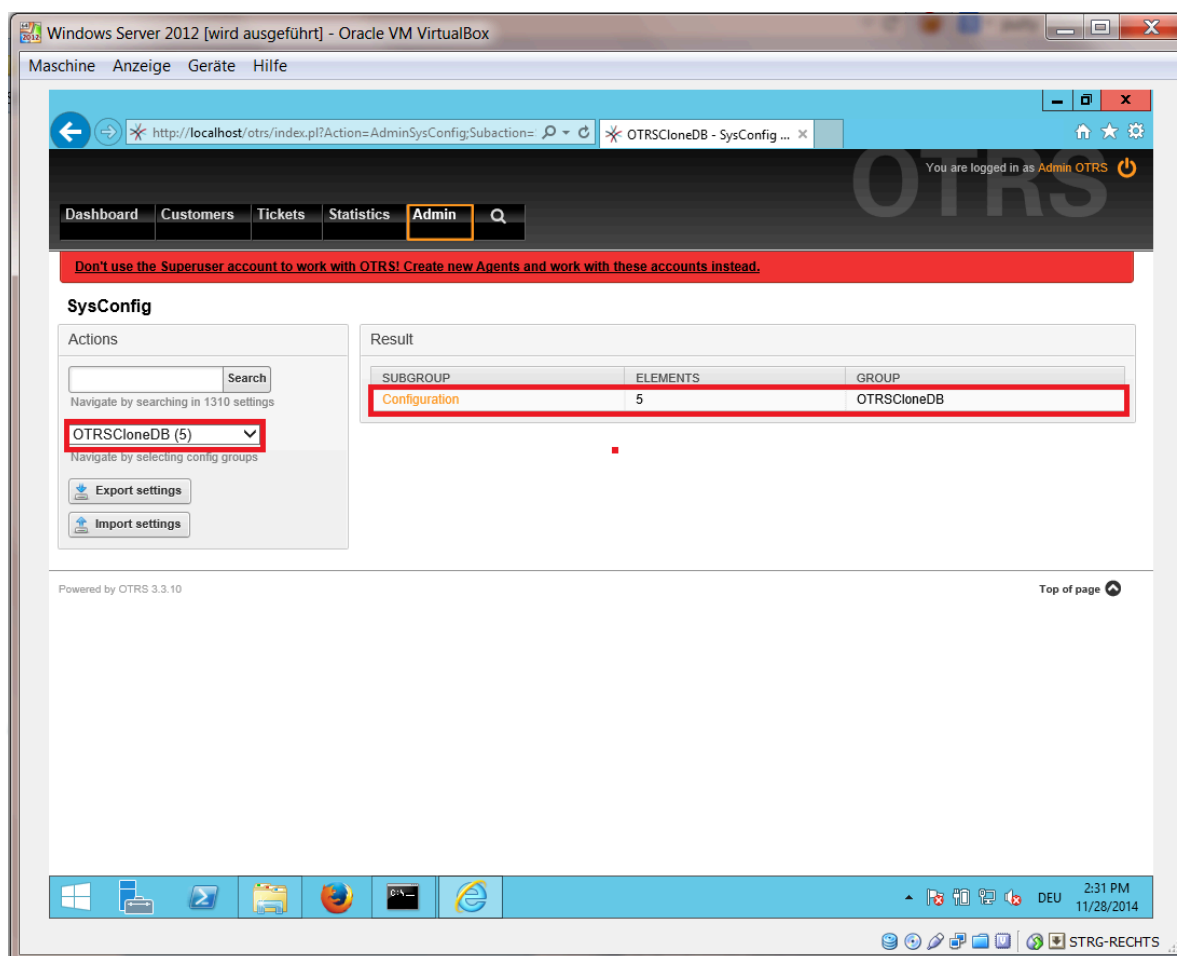
```
I88cm34A1B6xrl3
```

把密码写在一张纸上。

#### 4.1.7. 克隆数据库到目标系统

切换回Windows版的OTRS，打开系统管理菜单中的系统配置。选择“OTRSCloneDB”组-“Configuration”子组：

图 2.13. 配置OTRSCloneDB的系统配置1 - 屏幕截图



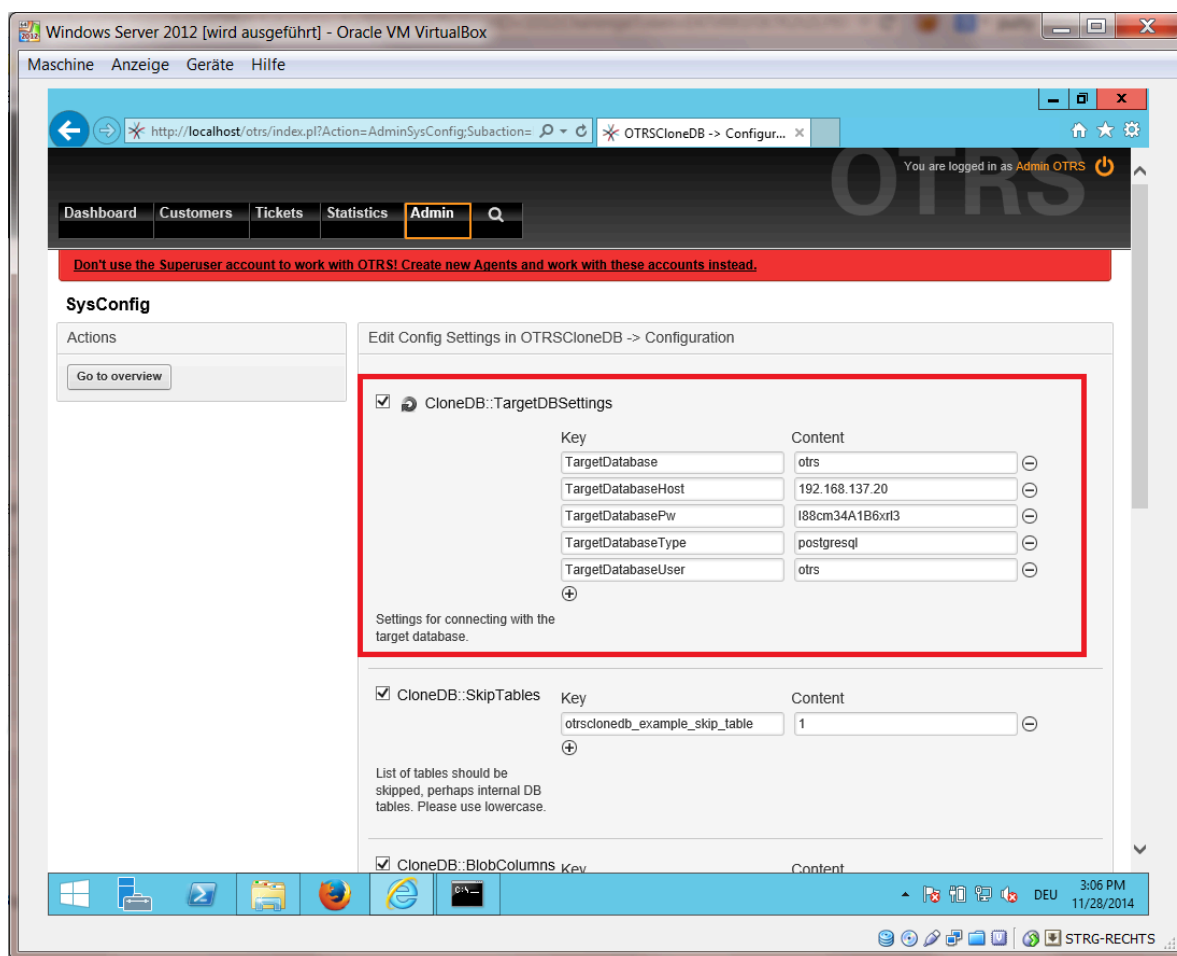
我们需要使用下列值来配置系统配置选项OTRSCloneDB::TargetSettings :

```

TargetDatabaseHost => 192.168.137.20 (Here you need to enter the ip address of your target
system)
TargetDatabase => otrs
TargetDatabaseUser => otrs
TargetDatabasePw => I88cm34A1B6xrl3 (Here you need to set the password of your target
system)
TargetDatabaseType => postgresql
  
```



图 2.14. 配置OTRSCloneDB的系统配置2 - 屏幕截图



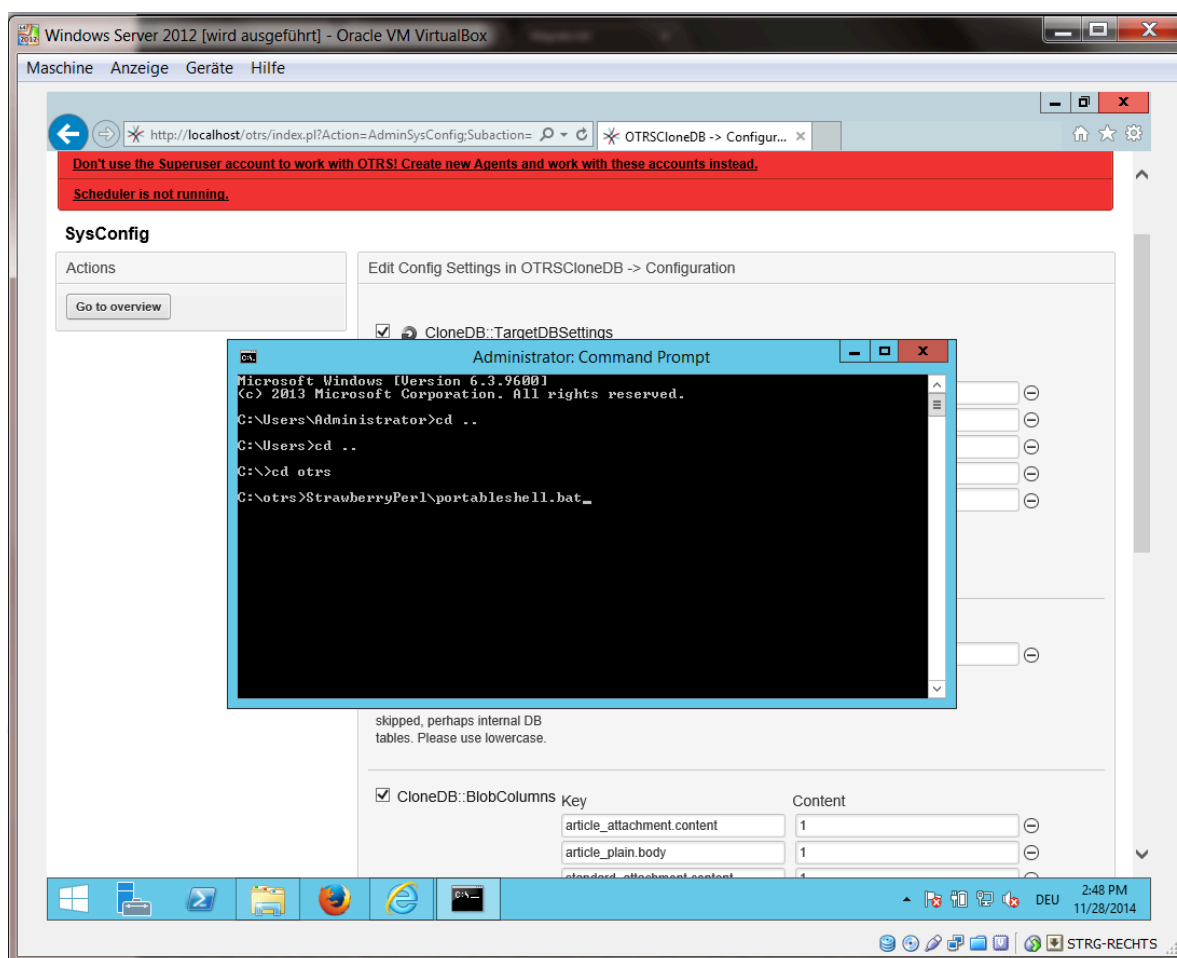
要运行OTRSCloneDB脚本我们需要切换到Windows版本的OTRS的命令行窗口，并进入OTRS安装的目录：

```
shell> cd "C:\otrs"
```

如果你使用的是StrawberryPerl，则你需要激活Perl shell：

```
shell> StrawberryPerl\portableshell.bat
```

图 2.15. 运行OTRSCloneDB脚本1 - 屏幕截图



OTRSCloneDB脚本位于OTRS目录的bin目录。

```
shell> cd "OTRS\bin"
```

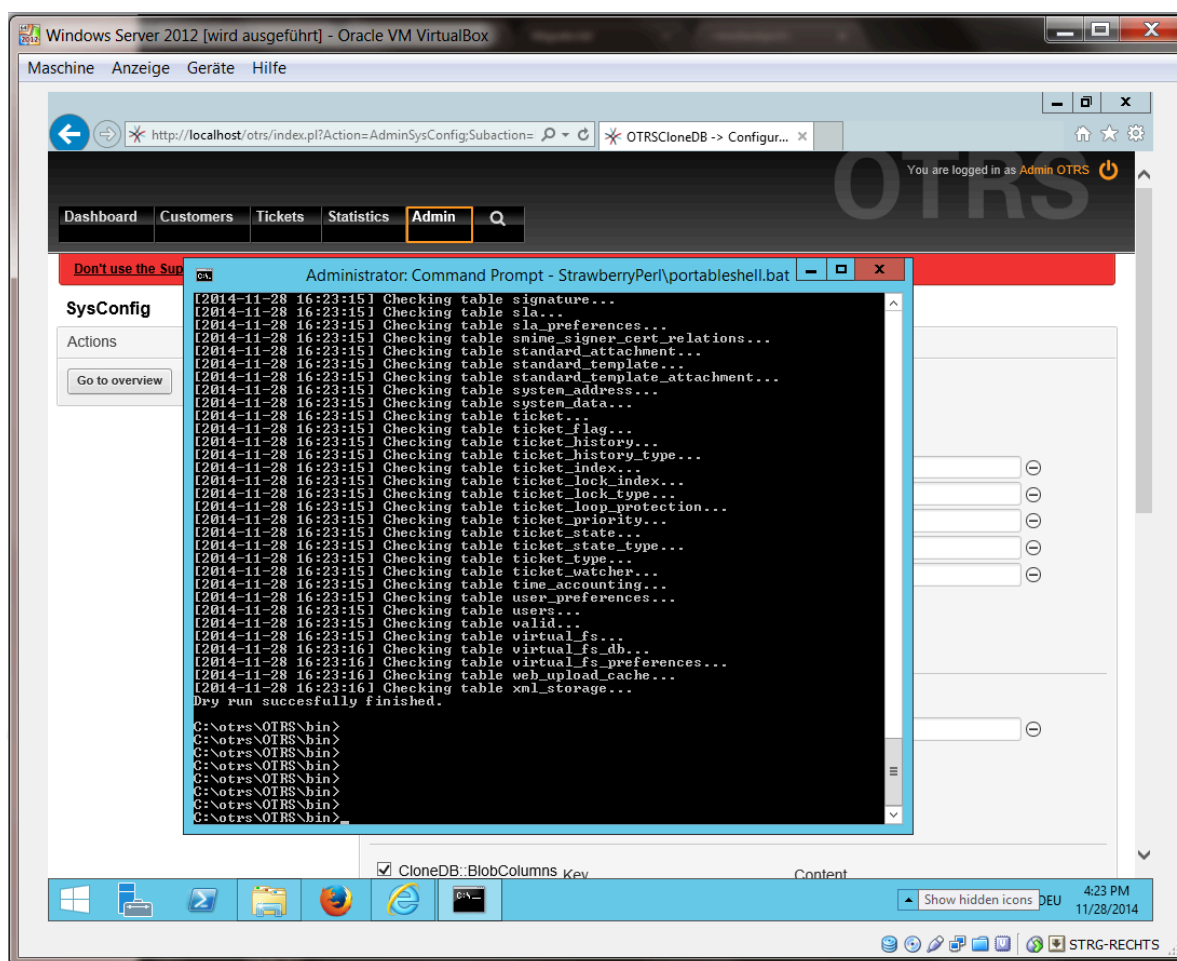
运行OTRSCloneDB脚本：

```
shell> perl otrs.CloneDB.pl
```

现在你应该能够看到关于这个脚本及其参数的一些信息。



图 2.17. 运行OTRSCloneDB脚本3 - 屏幕截图



开始克隆数据库，交叉手指祈祷好运吧：

```
shell> perl otrs.CloneDB.pl -r
```

一个成功的运行就像这样：

```
Generating DDL for OTRS.
Generating DDL for package OTRSCloneDB.
Creating structures in target database (phase 1/2)
...
...
Creating structures in target database (phase 2/2)
...
done.
```

#### 4.1.8. 从Windows系统复制下列文件到目标系统：

你需要从Windows系统复制一些文件到目标系统，你可以用一个免费工具软件如“WinSCP”（直接在网搜索“WinSCP”就行）。从Windows版本的OTRS中复制下列文件：

```
C:\otrs\OTRS\Kernel\Config\GenericAgent.pm
C:\otrs\OTRS\Kernel\Config\Files\ZZZAuto.pm
C:\otrs\OTRS\var\article\*
C:\otrs\OTRS\var\log\TicketCounter.log
```

到目标系统：

```
/opt/otrs/Kernel/Config/GenericAgent.pm  
/opt/otrs/Kernel/Config/Files/ZZZAuto.pm  
/opt/otrs/var/article/*  
/opt/otrs/var/log/TicketCounter.log
```

在目标系统中打开文件 `/opt/otrs/Kernel/Config/Files/ZZZAuto.pm`，并将所有类似“`C:/otrs/OTRS/`”格式的路径替换为“`/opt/otrs/`”格式的路径!

如果你手动修改过文件`Kernel/Config.pm`，请复制这些修改内容到目标系统的`Kernel/Config.pm`。不要复制原样复制，因为你现在的数据库设置不同，并且文件路径在目标系统和原Windows系统不一样！

#### 4.1.9. 重新安装所有软件包

用软件包管理器重新安装所有软件包，以获取所有定制的文件。

```
shell> bin/otrs.SetPermissions.pl --otrs-user=otrs --otrs-group=otrs --web-user=www-data --  
web-group=www-data /opt/otrs  
shell> su - otrs  
shell> cd /opt/otrs  
shell> perl bin/otrs.Console.pl Maint::Cache::Delete  
shell> perl bin/otrs.Console.pl Maint::Loader::CacheCleanup  
shell> perl bin/otrs.Console.pl Admin::Package::ReinstallAll  
shell> exit
```

再次修复OTRS系统的所有权限：

```
shell> bin/otrs.SetPermissions.pl --otrs-user=otrs --otrs-group=otrs --web-user=www-data --  
web-group=www-data /opt/otrs
```

#### 4.1.10. 禁用目标系统PostgreSQL数据库的远程访问

撤消在目标系统启用PostgreSQL数据库远程访问的所有步骤。

切换到`postgresql`目录：

```
shell> cd /etc/postgresql/9.4/main  
shell> vi postgresql.conf
```

移除这个文件末尾的下列行：

```
listen_addresses = '*'
```

保存文件

```
shell> vi pg_hba.conf
```

移除这个文件末尾的下列行：

```
host all all 0.0.0.0/0 md5
```

保存文件

重启PostgreSQL服务器

```
shell> service postgresql restart
```

### 4.1.11. 启动OTRS服务

启动目标系统的服务：

```
shell> service cron start
shell> service apache2 start
shell> su - otrs
shell> cd /opt/otrs/
shell> bin/Cron.sh start
shell> bin/otrs.Daemon.pl -a start
```

现在你就能在浏览器中打开复制了Windows版本数据的OTRS系统。

## 5. 升级OTRS版本（从4到5）

这些说明适用于想升级OTRS版本的人群，从4到5或从5到版本5最近的补丁集。

如果你正运行在一个低版本的OTRS上，首先需要按如下升级顺序先升级到OTRS 4 : 1.1->1.2->1.3->2.0->2.1->2.2->2.3->2.4->3.0->3.1->3.2->3.3->4！ 在每个版本间需要执行完全升级，包括修改数据库和升级的Perl脚本。

请注意：如果你从OTRS 2.2之前的版本升级，还需要参照[执行额外的步骤](#)。

在大版本内你可以跳过小版本升级，比如你可以直接从OTRS 5 补丁级别2直接升级到版本5补丁级别6。如果你只是升级补丁级别，可以跳过步骤6、10、11、14、17和18。

强烈建议在一台独立的测试主机上先进行升级测试。

### 5.1. 步骤1：停止所有相关的服务

请确保没有任何运行中的服务或CRON计划任务还在试图访问OTRS。取决于你的服务配置，下面是一个例子：

```
shell> /etc/init.d/cron stop
shell> /etc/init.d/postfix stop
shell> /etc/init.d/apache stop
```

取决于从哪个版本升级，按以下顺序停止OTRS CRON计划任务或守护进程：

```
shell> cd /opt/otrs/
shell> bin/Cron.sh stop
shell> bin/otrs.Scheduler.pl -a stop
```

或

```
shell> cd /opt/otrs/
```

```
shell> bin/Cron.sh stop  
shell> bin/otrs.Daemon.pl stop
```

## 5.2. 步骤2：备份/opt/otrs/下面的所有内容

- Kernel/Config.pm
- Kernel/Config/GenericAgent.pm (仅供参考, 这个文件不再需要)
- Kernel/Config/Files/ZZZAuto.pm
- var/\*
- 当然还有数据库

## 5.3. 步骤3：确保你已经备份了所有需要备份的 ;-)

## 5.4. 步骤4：安装新版本（源码或RPM包）

### 5.4.1. 步骤4.1：使用源码：

```
shell> cd /opt  
shell> mv otrs otrs-old  
shell> tar -xzf otrs-x.x.x.tar.gz  
shell> mv otrs-x.x.x otrs
```

#### 5.4.1.1. 恢复原配置文件

- Kernel/Config.pm
- Kernel/Config/Files/ZZZAuto.pm

#### 5.4.1.2. 恢复TicketCounter.log

为了让OTRS继续使用正确的工单编号，恢复文件TicketCounter.log到目录/opt/otrs/var/log/。这对于使用自动增长的工单编号特别有用。

#### 5.4.1.3. 恢复工单数据

如果你配置OTRS将信件数据存储存储在文件系统，你必须恢复article目录到/opt/otrs/var/或在系统配置中指定的目录。

#### 5.4.1.4. 设置文件权限

请执行

```
shell> cd /opt/otrs/  
shell> bin/otrs.SetPermissions.pl
```

参数为你的系统设置需要的权限。例如：

- 以OTRS用户运行WEB服务器：

```
shell> bin/otrs.SetPermissions.pl --web-group=otrs
```

- 以wwwrun用户运行WEB服务器（如SUSE）：

```
shell> bin/otrs.SetPermissions.pl --web-group=wwwrun
```

- 以apache用户运行WEB服务器（如Red Hat、CentOS）：

```
shell> bin/otrs.SetPermissions.pl --web-group=apache
```

- 以www-data用户运行WEB服务器（如RDebian、Ubuntu）：

```
shell> bin/otrs.SetPermissions.pl --web-group=www-data
```

## 5.4.2. 步骤4.2：使用RPM包：

```
shell> rpm -Uvh otrs-x.x.x.-01.rpm
```

在这种情况下通过RPM升级会自动恢复原配置文件并设置文件权限。

## 5.5. 步骤5：检查所需的Perl模块

验证所有需要的Perl模块都已经安装到系统中，并安装可能缺失的模块。

```
shell> /opt/otrs/bin/otrs.CheckModules.pl
```

## 5.6. 步骤6：应用数据库修改

### 5.6.1. 步骤6.1：数据库schema升级

#### 5.6.1.1. MySQL：

##### 注意

注意：在MySQL升级过程中会在默认的表存储引擎创建新表。在MySQL 5.5中，新的默认存储引擎是 InnoDB。如果现有的表如‘users’表的表存储引擎是MyISAM，则在创建外键约束时会报错。在这种情况下，我们推荐将所有的表切换到InnoDB引擎，可通过执行命令行脚本切换：**bin/otrs.Console.pl Maint::Database::MySQL::InnoDBMigration**。

任何与存储引擎相关的问题都可以通过命令检查出来：**bin/otrs.Console.pl Maint::Database::Check**，所以请运行这个命令来检查可能的问题。

```
shell> cd /opt/otrs/  
shell> cat scripts/DBUpdate-to-5.mysql.sql | mysql -p -f -u root otrs  
shell> bin/otrs.Console.pl Maint::Database::Check
```

#### 5.6.1.2. PostgreSQL：

```
shell> cd /opt/otrs/
```



```
shell> cat scripts/DBUpdate-to-5.postgresql.sql | psql --set ON_ERROR_STOP=on --single-transaction otrs otrs
```

## 5.6.2. 步骤6.2：数据库迁移脚本

运行迁移脚本（用otrs用户，而不是root用户）：

```
shell> scripts/DBUpdate-to-5.pl
```

如果这个脚本运行不正常，请不要继续升级，否则可能丢失数据。

## 5.7. 步骤7：刷新配置缓存并删除缓存

请运行（用otrs用户，而不是root用户）：

```
shell> cd /opt/otrs/  
shell> bin/otrs.Console.pl Maint::Config::Rebuild  
shell> bin/otrs.Console.pl Maint::Cache::Delete
```

## 5.8. 步骤8：重启服务

例如（取决于使用的服务，可能有差异）：

```
shell> /etc/init.d/apache start  
shell> /etc/init.d/postfix start  
shell> /etc/init.d/cron start
```

现有你可以登录到系统了。

## 5.9. 步骤9：检查安装的软件包

### 注意

OTRS 4的软件包与OTRS 5不兼容，所以你还需要升级软件包！

下列软件会在升级过程中自动卸载（如果之前安装了的话）：

- OTRSGenericInterfaceMappingXSLT

## 5.10. 步骤10：配置NodeID（仅适用于多前端的群集安装）

### 注意

这一步仅在你有多台前端主机的群集安装时才需要。

从OTRS 5开始，每个前端服务器需要有它自己唯一的NodeID（节点编号）。对单前端安装来说默认就是1，不需要配置。如果有多台主机，每台主机都需要设置这个唯一值，可为从1到999的任一值。这个配置在文件Kernel/Config.pm中。

```
$Self->{'NodeID'} = '2'; # 为每个前端服务器分配一个唯一的值
```

## 5.11. 步骤11：检查跟进检测配置

跟进检测设置被重新组织了。现在OTRS默认在邮件主题和引用内容中搜索来检测是否为跟进邮件。如果你想修改跟进检测设置（比如在正文、附件或原始邮件内容中搜索），请在系统管理的系统配置Ticket -> Core::PostMaster中检查设置。

## 5.12. 步骤12：启动OTRS守护进程

新的OTRS守护进程负责处理OTRS中的所有异步和循环任务。之前在cron文件中定义的自动任务现在都由OTRS守护进程处理。这个守护进程还处理所有的'GenericAgent自动任务'任务，它必须以otrs用户运行。

```
shell> /opt/otrs/bin/otrs.Daemon.pl start
```

## 5.13. 步骤13：升级和激活cron任务

在/opt/otrs/var/cron/\*.dist有两个默认的cron文件，它们的目的是确保OTRS守护进程正常运行。它们需要复制为没有“.dist”扩展名的新文件来激活。

```
shell> cd /opt/otrs/var/cron
shell> for foo in *.dist; do cp $foo `basename $foo .dist`; done
```

要在系统中安排这些cron任务，你可以使用otrs运行脚本文件Cron.sh。

```
shell> /opt/otrs/bin/Cron.sh start
```

请注意：如果你有定制的cron任务，应该考虑将它们移到系统配置Daemon -> Daemon::SchedulerCronTaskManager::Task中，以便由OTRS守护进程来统一执行。你可能还需要修改你的定制脚本，因为现在大部分OTRS命令都通过bin/otrs.Console.pl统一管理，而不是单个的脚本。

### 注意

控制台命令Dev::Code::Generate::ConsoleCommand可以帮助创建新的控制台命令用来自制脚本。这个命令会创建一个模板，可以调整其中的脚本逻辑和相关参数。

## 5.14. 步骤14：回顾你的工单通知

在OTRS 5中工单通知配置与之前的版本不一样。它们现在是“事件通知”（以前的版本也有）的一部分（现在被称为“工单通知”）。你原有的工单通知已经迁移到工单通知列表中，但没有激活。你还会接收到新的默认工单通知（默认激活的）。

如果你没有修改老版本中的工单通知你就不需要再做任何事情。如果你做了本地的修改或翻译，可有两个选择：a) 你可以迁移你的修改内容到新的工单通知（推荐），或b) 你可以不激活新的通知并重新激活老版本的通知。

还要检查升级通知收件人设置，看看是否匹配你的期望值（工单在其队列中服务人员 vs 拥有读取权限的所有服务人员）。这个设置之前是通过文件Kernel/Config/GenericAgent.pm（现已废弃）来配置的，现在可以通过图形界面中每个通知类型单独控制。

## 5.15. 步骤15：升级系统注册（可选）

如果系统已经注册到OTRS集团，强烈推荐这个时候更新注册信息。这将在OTRS集团的记录中更新注册的版本信息及其它变化了的信息，以便从云服务中获得更准确的信息。

如果你不手动更新注册信息，则会在定期连接时自动更新，但这可能是几个小时或几天后的事了，在此期间可能会从云服务得到一些错误信息比如：**OTRS Business Solution™**更新。

```
shell> cd /opt/otrs/  
shell> bin/otrs.Console.pl Maint::Registration::UpdateSend --force  
shell> bin/otrs.Console.pl Maint::Cache::Delete
```

## 5.16. 步骤16：迁移基于Perl的定制自动任务（可选）

这一步仅与你有任何定制自动任务使用了OTRS 4的Perl配置文件如Kernel/Config/GenericAgent.pm有关，这些任务不再能由定期的工单通知处理。这些任务现在需要在系统配置（Daemon -> Daemon::SchedulerCronTaskManager::Task）中注册为OTRS守护进程的cron任务，以便能够定期执行。

在系统配置中为此准备了5个设置参数

（Daemon::SchedulerCronTaskManager::Task###GenericAgentFile1到Daemon::SchedulerCronTaskManager::Task###GenericAgentFile5）如果还不够，可以添加到一个定制的系统配置文件中。

请用<ModuleName>替换包含定制的自动任务的配置文件名，如用Kernel::Config::GenericAgent替换Kernel/Config/GenericAgent.pm，按以前的执行周期设置计划时间，选中这个参数（激活）并保存修改。

### 注意

要获取关于运行自动任务参数的更多信息，请执行以下语句：

```
shell> bin/otrs.Console.pl Maint::GenericAgent::Run --help
```

## 5.17. 步骤17：设置bash自动补齐功能（可选）

所有的常规OTRS命令操作现在都通过OTRS控制台接口 bin/otrs.Console.pl执行。它提供bash的自动补齐功能，在查找正确的命令和选项时更容易些。

你可以安装软件包bash-completion来激活bash的自动补齐功能。它能为otrs用户自动检测到并载入文件/opt/otrs/.bash\_completion。

重启shell后，你可以在输入bin/otrs.Console.pl后按TAB键，就会显示所有可用的命令。如果你键入了命令的部分字符，按TAB键会列出所有匹配这些字符的命令。在输入完整的命令后，按TAB键会列出所有可用的选项和参数。

## 5.18. 步骤18：回顾工单操作窗口配置（可选）

一些工单操作窗口比如“备注”在OTRS 4中有默认的主题（‘备注’的默认主题可通过系统配置参数Ticket::Frontend::AgentTicketNote###Subject设置）。这些默认的主题已经被移除了，以便减少潜在的冗余或无意义的数据库。如果这些对你很重要，可以重新添加。

## 5.19. 步骤19：审查工单事件模块配置

由于在 [OTRS 5s的修正BUG](#)，一些工单事件模块的配置名称必须被修改。任何可能的定制都需要重新应用。请检查您在SysConfig -> Ticket -> Core::Ticket 的设置并检查以 Ticket::EventModulePost开头的所有定制设置是否需要重新应用。例如，你已经激活并配置的事件处理程序以将客户数据保存在工单动态字段中的设置。

## 5.20. 步骤20：搞定！

---

## 6. 附加的应用程序

你可以安装附件的应用程序包来扩展OTRS框架的功能。系统管理页面的软件包管理器能在线软件仓库下载应用程序包，并管理软件包的依赖性，还可以从本地文件安装软件包。

### 6.1. FAQ 知识库

FAQ知识库是知识基础组件。它支持知识文章的编辑和查看。知识文章可以按服务人员、客户联系人或匿名用户来限制查看权限，还可以按‘组’来组织并用不同的语言查看。



## 第 3 章 OTRS 第一步

本章的目的是提供OTRS简短的概述及其web界面的结构，介绍术语‘服务人员’、‘客户’和‘系统管理员’。还以OTRS系统管理员登录并仔细看看每个用户帐户可用的偏好设置。

### 1. 服务人员WEB界面

#### 注意

Before logging on for the first time, please consider activating the HTTPS on your web server in order for OTRS to be served as a secure app over SSL/TLS protocol. For detailed instructions on how to do this, please consult the documentation of your web server (e.g. Apache2).

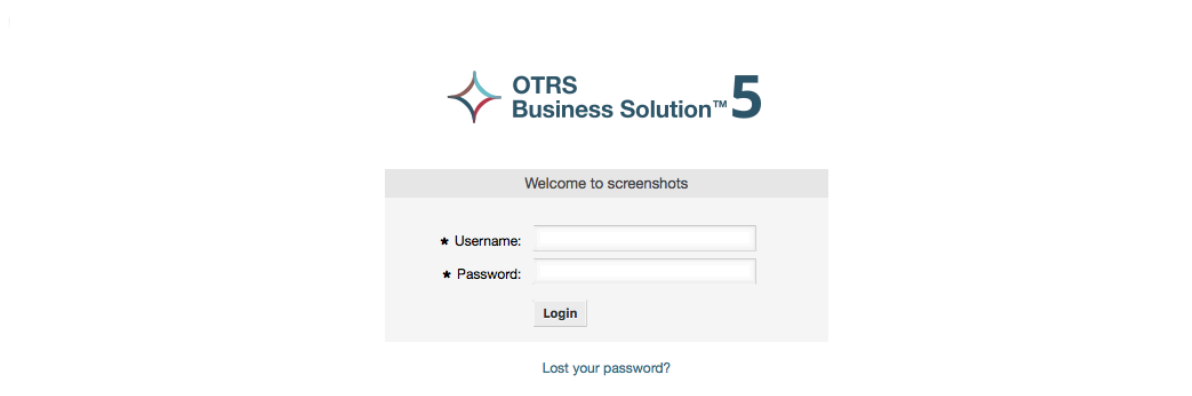
After you activate HTTPS, please set the configuration option `HttpType` to `https` in `SysConfig`. This will make sure that all internal links in OTRS are indeed using HTTPS.

Good practice is to also redirect all HTTP traffic to HTTPS, in case someone tries to access the OTRS via an insecure link. This should be done on web server configuration level for maximum effect.

服务人员WEB界面允许服务人员答复客户请求、为客户或其它服务人员创建新工单、创建客户的电话工单、编写FAQ知识库条目、编辑客户信息数据等等。

假设你的OTRS主机可以通过网址<http://www.example.com>访问，则可以在WEB浏览器中输入地址：<http://www.example.com/otrs/index.pl>访问OTRS登录窗口。

图 3.1. 服务人员界面登录窗口

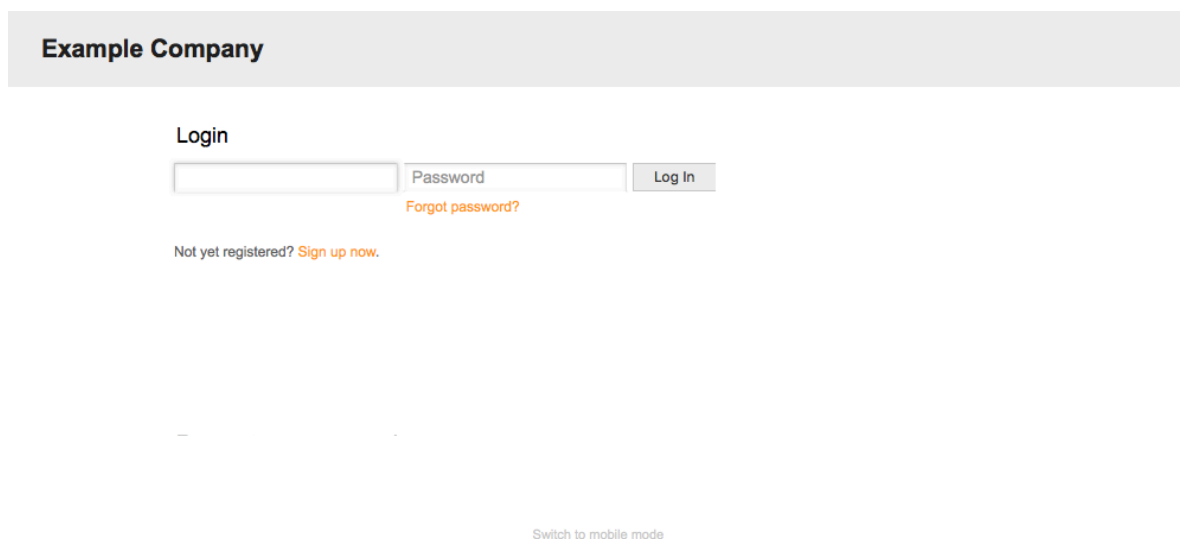


### 2. 客户WEB界面

在OTRS中客户有一个单独的WEB界面，可以创建新帐户、修改帐户设置、创建和编辑工单、查看他们创建的工单概览等等。

继续上面的示例，通过在浏览器中输入网址<http://www.example.com/otrs/customer.pl>即可访问客户登录窗口。

图 3.2. 客户界面登录窗口

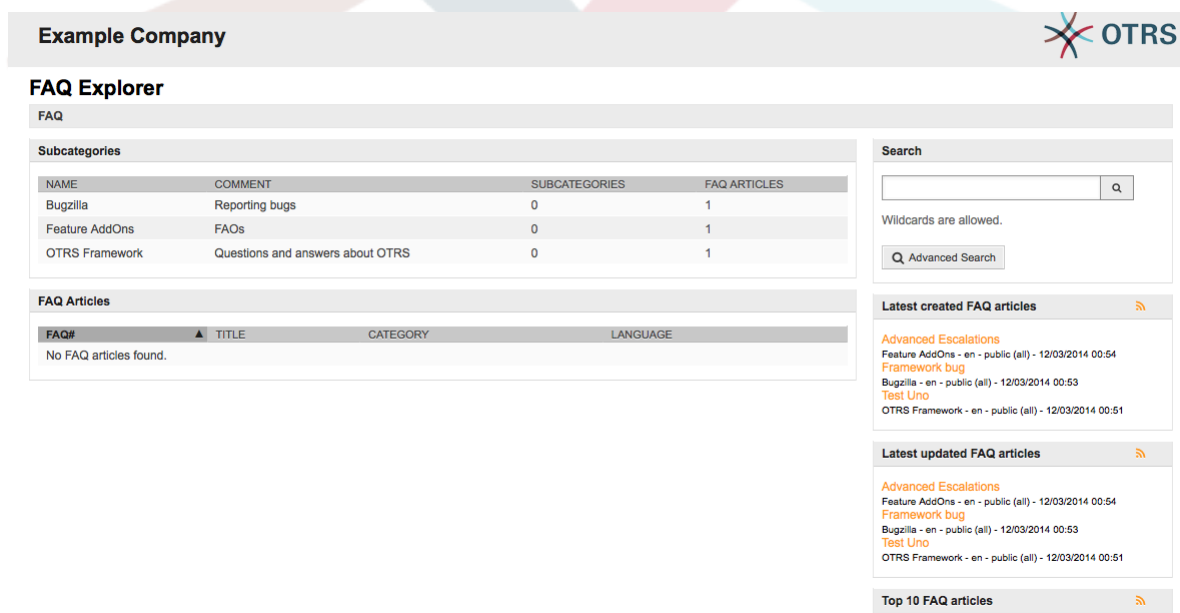


## 3. 公共WEB界面

作为服务人员和客户WEB界面的补充，OTRS还有一个访问FAQ知识库的公共WEB界面。FAQ知识库模块需要单独安装。它提供了到FAQ知识库系统的公共访问，并让来访者无需特别授权即可搜索FAQ知识库文章。

在我们的示例中，可以通过网址<http://www.example.com/otrs/faq.pl>或<http://www.example.com/otrs/public.pl>。

图 3.3. 公共WEB界面



## 4. 首次登录

按服务人员界面 部分介绍的方法访问OTRS登录窗口，输入用户名和密码。因为系统刚安装完毕且还没有创建任何用户，首先以OTRS系统管理员登录，使用'root@localhost'作为用户名，密码是安装程序最后一步生成的。

## 警告

这个帐户数据在每个新安装的OTRS系统都是有效的。你应该尽快修改这个OTRS系统管理员的密码！可以通过OTRS系统管理员帐户的偏好设置来修改密码。

如果你不想以OTRS系统管理员登录，只需输入普通的服务人员帐户信息即可。

如果你忘记了你的密码，可以向系统请求一个新密码。只需点击“登录”按钮下方的链接，输入注册的OTRS帐户的用户名，并点击“请求新密码”按钮即可（见下图）。

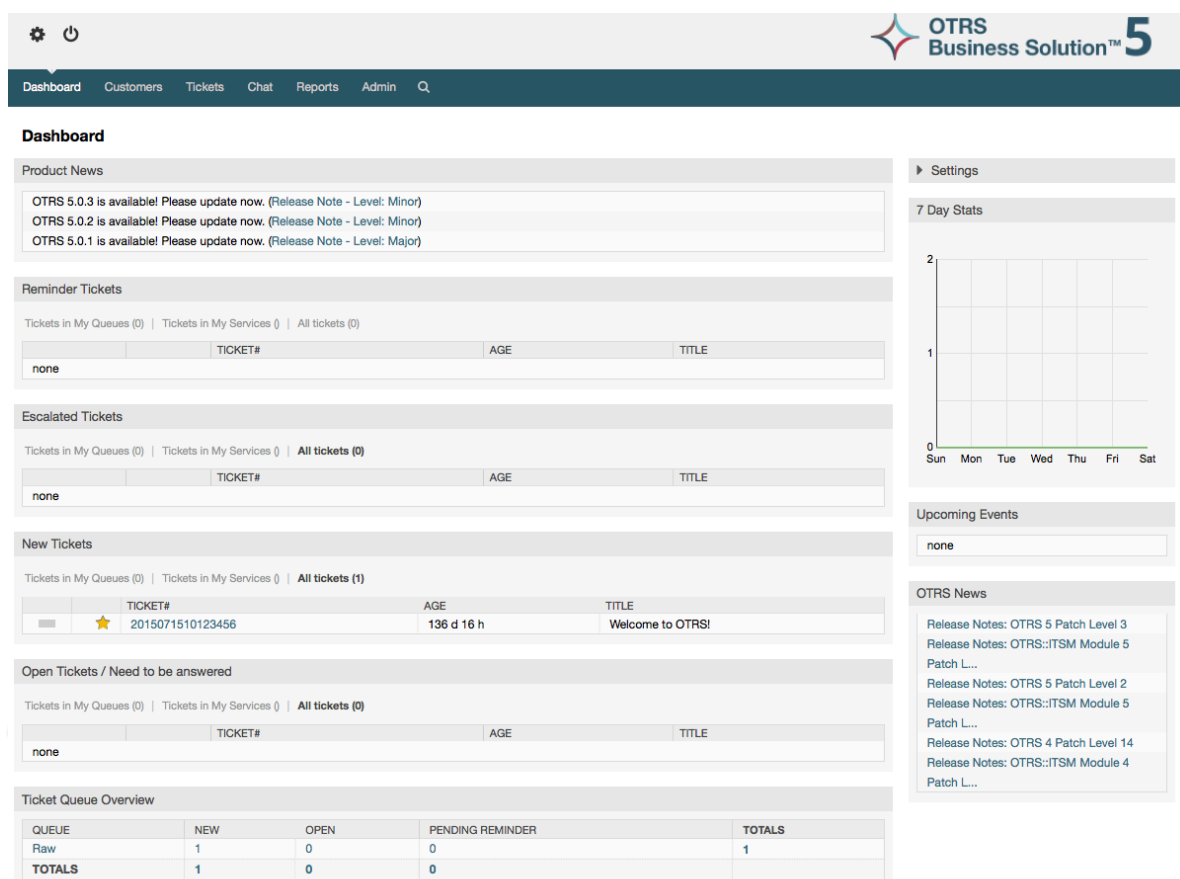
图 3.4. 请求新密码



## 5. WEB界面概览

一旦成功登录到系统中，首先看到的的就是仪表盘页面（见下图）。它显示了你锁定的工单，允许通过菜单直接访问队列视图、状态视图和升级视图，还保存了创建电话和邮件工单的选项。还提供了不同条件的工单快速汇总数。

图 3.5. 服务人员界面仪表盘



为了提高清晰度，通用WEB界面分成了不同的区域。每个页面的第一行显示一些通用信息比如退出按钮、标有锁定工单数的图标（可直接访问锁定的工单）、创建新的电话/邮件工单的链接等等。还有些图标可以进入队列视图、状态视图和升级视图。

在这个图标行的下面就是导航栏。导航栏显示一个菜单，让你导航到系统中的不同区域或模块，及执行一些全局操作。点击‘仪表盘’按钮就进入仪表盘页面。如果你点击‘工单’按钮，会得到一个子菜单，让你切换工单的视图、创建一个新的电话/邮件工单或搜索一个特定的工单。点击‘统计’按钮后会提供一个菜单让你从已注册的统计概览中选择统计、创建一个新的统计或从已有文件中导入统计。‘客户’按钮让你进入客户管理窗口。通过点击‘系统管理’按钮，可能可以访问所有的系统管理员模块，允许你创建新的服务人员、队列等等。还有一个‘搜索’按钮用来搜索工单。

如果还安装相关的模块如FAQ知识库或调查，还会显示访问这些模块的按钮。

在导航栏下方区域中，可以显示不同的系统消息。如果你用默认的OTRS系统管理员帐号登录，就会得到一个红色的消息，警告你不要使用该系统帐号干活。

在你当前所在区域/模块的标题下面，有几个包含你正在操作的窗口的相关信息子区域，每个子区域都在一个单独的盒子中。

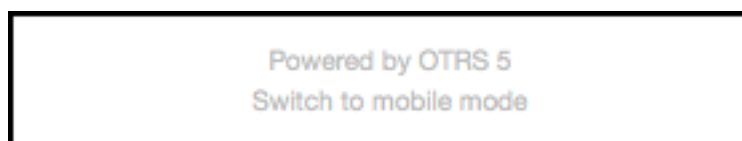
这些盒子包含了每个窗口的主要部分，通常它们显示为一个或几个列，每个盒子都储存当前窗口的相关信息，比如使用说明、建议、概述等等。还会显示执行每个窗口相关操作所需的表单或工具，比如添加/更新/删除记录、检查日志、修改配置设置等等。

最后，在页面的底端，显示的是网页脚（见下图）。它包含一个链接以便直接访问OTRS官方网站或切换到移动模式。

一般地，图标行、导航栏和页脚在WEB界面的所有窗口都是相同的。



图 3.6. 页脚



## 6. 仪表板

仪表板是OTRS系统的主页，在这里你可以得到关于工单和工单活动的其它相关资料的概览。仪表板是一个服务人员日常工作的起点，默认展示各类工单的快速汇总（处于挂起、升级、新建和处理中状态的工单数），以及其它信息。

仪表板最重要的功能就是：它是完全可定制的。这意味着你可以根据需要配置每一个元素的显示或隐藏。甚至在当前列内随意调整各部分的位置，只需点击元素顶部并拖动，然后放到任意位置即可。每个元素叫做“小部件”，OTRS系统有一些开箱即用的小部件，但是模块化的设计使得在仪表板中集成定制小部件很容易。

仪表板窗口被分成了两个主要的列，左边的列通常是你看到工单按状态分类的信息，如提醒、升级、新建和处理中。每个小部件上你可以过滤结果，以分别查看有权访问的所有工单、你锁定的工单、你所属队列中的工单以及其它过滤条件的结果。这个列中还有其它类型的小部件，都在下面介绍。

图 3.7. 仪表板小部件

**Dashboard**

Product News  
Can't connect to Product News server!

Reminder Tickets  
Tickets in My Queues (0) | Tickets in My Services (0) | All tickets (0)

TICKET#	AGE	TITLE
none		

Escalated Tickets  
Tickets in My Queues (0) | Tickets in My Services (0) | All tickets (0)

TICKET#	AGE	TITLE
none		

New Tickets  
Tickets in My Queues (0) | Tickets in My Services (0) | All tickets (1)

TICKET#	AGE	TITLE
★ 2015071510123456	136 d 16 h	Welcome to OTRS!

Open Tickets / Need to be answered  
Tickets in My Queues (0) | Tickets in My Services (0) | All tickets (0)

TICKET#	AGE	TITLE
none		

Ticket Queue Overview

QUEUE	NEW	OPEN	PENDING REMINDER	TOTALS
Raw	1	0	0	1
<b>TOTALS</b>	<b>1</b>	<b>0</b>	<b>0</b>	

仪表板左边列的小部件。

- 工单列表小部件

在这个类别下的小部件共享一些相同的整体行为、外观和感觉。这些小部件显示不同状态的工单列表。每个列表页面显示的工单数量可以在小部件设置（当你把鼠标移到小部件的右上方时就会显示）里配置。这些小部件支持下面的过滤器：

- 我锁定的工单

被当前登录的服务人员锁定的工单。

- 我关注的工单

在当前登录的服务人员的关注清单中的工单，需要打开Ticket::Watcher设置才会显示。

- 我负责的工单

当前登录的服务人员被设置为工单负责人的工单，需要打开Ticket::Responsible设置才会显示。

- 我队列中的工单

服务人员在偏好设置中定义“我的队列”，在“我的队列”出现的队列中的工单。

- 我服务的工单

服务人员在偏好设置中定义“我的服务”，关联了“我的服务”出现的服务的工单，并且服务人员需要对工单所在的队列至少有只读权限。

- 所有的工单

这个服务人员能访问的所有工单。

这些小部件是：

- 提醒的工单

被设置为挂起并且提醒时间已到工单。

- 升级的工单

升级了的工单。

- 新的工单

状态为“新建”的工单。

- 处理中/需要回复的工单

状态为“处理中”并且准备好处理的工单。

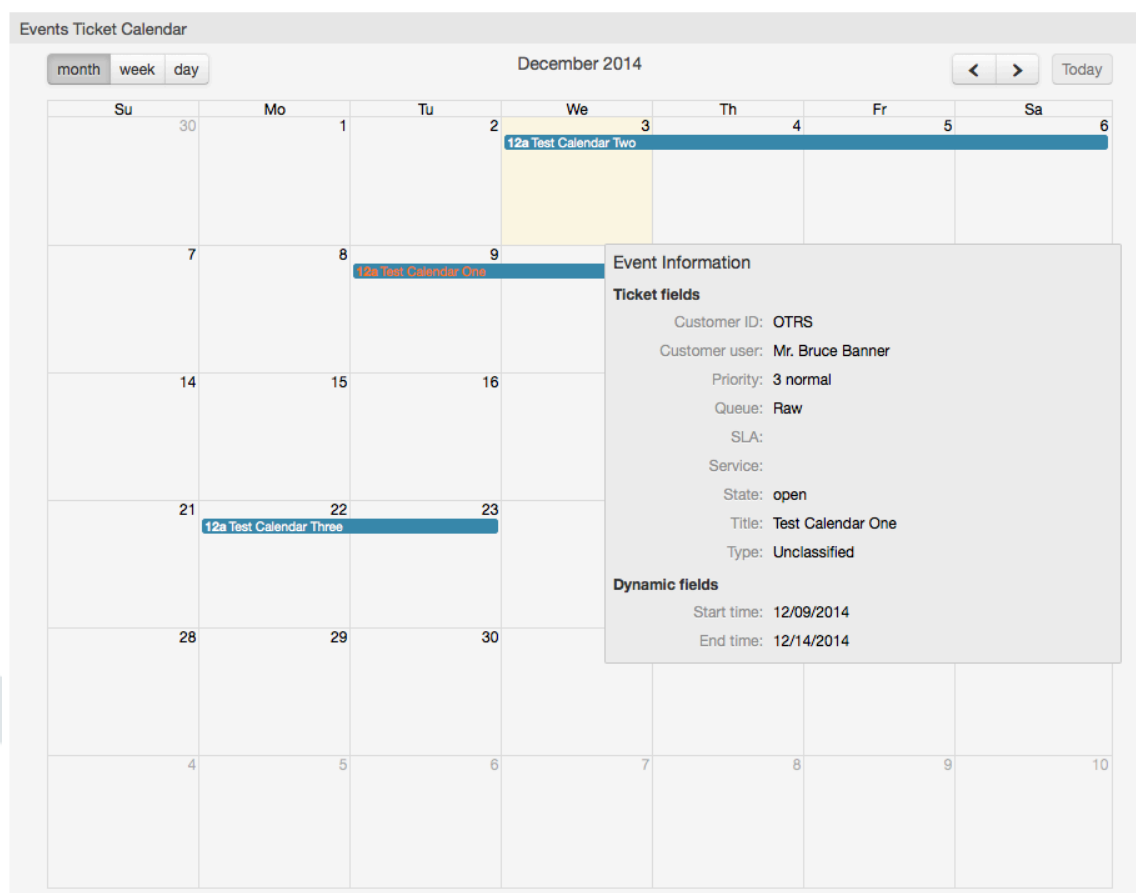
- 事件日历

在创建一个新工单后就定义了一个日历事件（对这个小部件而言），需要启用事件日历功能，并且还需要在工单创建窗口显示两个新的字段，一个是事件开始时间，另一个是事件结束时间，这两个时间用来确定事件的持续时间。

这个小部件包括下列视图：月、周和天。服务人员可能通过左右箭头按钮来滚动日历内容。

仅仅如前所述那样启用小部件还不够，还要添加一组工单的“日期/时间”动态字段到系统中（通过“系统管理”面板的“动态字段”链接），并在系统配置中为这个小部件设置动态字段。既要配置创建工单窗口中显示动态字段，还要在工单创建窗口或任何其它操作窗口（如自定义字段窗口）填写它们的日期/时间值（日历事件的开始和结束时间）；如果你认为有必要，还需要配置为在工单详情窗口显示这些动态字段。

图 3.8. 事件日历小部件



这个小部件进一步的配置可以在系统配置的子组 Frontend::Agent::Dashboard::EventsTicketCalendar 中找到。

- **CalendarWidth** 日历宽度  
定义日历的宽度（%），默认为95%。
- **DynamicFieldStartTime** 动态字段开始时间  
定义开始时间的动态字段名称。
- **DynamicFieldEndTime** 动态字段结束时间  
定义结束时间的动态字段名称。
- **队列**  
只有在这个设置中指定的队列中的工单才会显示在事件日历中。
- **DynamicFieldsForEvents** 事件的动态字段  
定义日历中事件信息窗口显示的动态字段。
- **TicketFieldsForEvents** 事件的工单属性  
定义日历中事件信息窗口显示的工单属性。
- **工单队列概览**

这个小部件显示一个计数矩阵，行代表队列，列代表工单数统计，每个单元格显示的是指定状态的工单在特定队列中的数量。

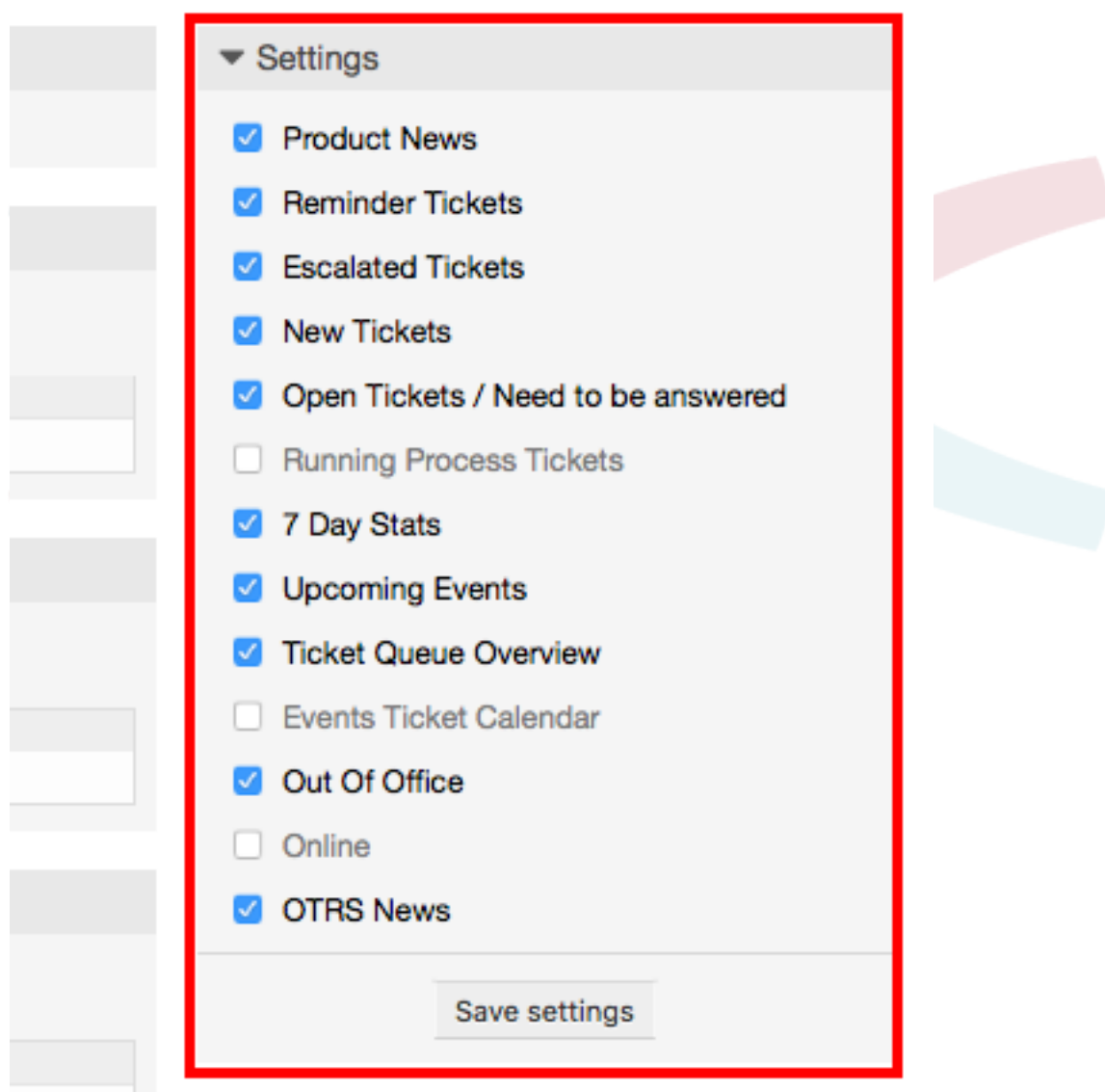
这个小部件还显示一个‘合计’行和一个‘合计’列，‘合计’行显示每个状态在所有队列中的汇总数，‘合计’列代表每个队列中各种状态的工单汇总数。

显示的队列和状态可以通过‘系统配置’修改。

点击小部件上的任何一个数字，都会打开一个工单搜索结果页面，以便让服务人员了解它们更详尽的概览。

在仪表板的右边列放置了一个特殊的小部分，允许你控制这些小部件的显示或隐藏，这就是‘设置’小部件。点击它的顶部来展开区域，可以看到所有可用的小部件，如图所示。每个小部件的名称前有一个复选框。使用这些复选框来定义对应小部件在仪表板的可见性（未选中的小部件不会显示），在你定义完可见性选项后单击“保存设置”按钮来存储你的修改内容。这个小部件是固定在窗口中的，意即你不能拖放或关闭它。

图 3.9. 仪表板设置



仪表板小部件右边的列。

- 最近7天统计

它用包括2条线的图形来显示最近7天的工单活动情况。一条线通常是绿色的，代表每天创建的工单数，另一条线通常是橙色的，代表每天关闭的工单数。

- 即将发生的事件

这里列出很快就要升级或已经升级的工单，这个小部件显示的信息很有帮助，因为你有机会知道哪些工单需要你特别注意，你也可以决定集中精力在哪个工单上，设置优先级或只是检查即将会发生的事。

- OTRS新闻

关于OTRS活动的完整列表，关于新产品版本或补丁的重要信息。

- 在线

这里会显示一个关于当前登录的服务人员的汇总信息，还包括在线客户联系人，请注意这个小部件通常是隐藏着的，可以通过前面描述的‘设置’小部件来显示它。

## 7. 一个队列是什么？

在许多邮件系统中，所有消息流入一个收件箱文件（存储消息的地方）很常见。新消息附加到这个收件箱文件的后面。用于读写邮件的邮件客户端程序读取这个收件箱文件并将内容呈现给用户。

OTRS中的队列在某种程序上就好比一个收件箱文件，因为它也能存储许多消息，一个队列还有一些超越收件箱文件的功能。作为一个OTRS服务人员或用户，你需要记住一个工单存储在哪个队列。服务人员能够处理或编辑一个队列中的工单，也能将工单从一个队列转移到另一个队列。但是他们为什么要转移工单？

要解释得更实际些，想想Max公司的例子（在一个工单系统的例子。Max安装了OTRS来让他的团队为购买了公司录像机的客户提供更好的支持。

在这个情况下一个队列就足够处理所有的请求了。但是，过了一段时间后，Max决定还要销售DVD录像机。现在客户不仅有关于录像机的问题，还有关于新产品的问题。越来越多的邮件进入到Max的OTRS系统的唯一队列中，很难对发生了什么有一个清晰的描述。

Max决定重建他的支持系统，并添加了两个新的队列。所以现有使用了3个队列。到达工单系统的新消息都存储到原来名为“raw”的队列中。两个新的队列中，一个命名为“录像机”专门用来处理录像机的请求，另外一个命名为“DVD录像机”则专门用于DVD录像机的请求。

Max要求Sandra关注“raw”队列，根据客户请求将消息分类（分派）到“录像机”或“DVD录像机”队列，John只能访问“录像机”队列，而Joe只能答复“DVD录像机”队列中的工单，Max能够编辑所有队列中的工单。

OTRS支持对用户、组和角色的访问管理，很容易设置只能由某些用户帐户访问的队列。Max还可以使用另外一种方法来将请求分派到不同队列，那就是过滤器规则。否则如果使用了两个不同的邮件地址，Sandra只需要分派这些邮件到两个其它不能被自动分派的队列。

将进入的消息分类到不同的队列有助于保持支持系统的结构化和整齐。因为你的服务人员被安排到不同的组中，有不同的队列访问权限，系统还能进一步优化。队列可以用来定义工作流程或创建一个公司的结构。例如Max可以部署另外一个叫“销售”的队列，包含“requests请求”、“offers报价”、“orders订单”、“billing开票”等子队列。这样一个队列结构可以帮助Max优化他的订单事务。

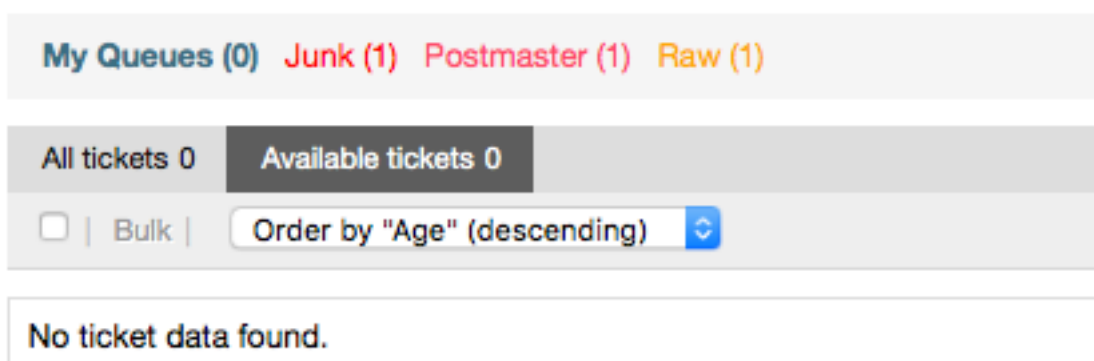
改进的系统结构，例如通过合适的队列设计，可以带来明显的时间和成本节约。队列可以帮助优化你公司的流程。

## 8. 队列概览是什么？

队列概览提供了一个在其中存在工单且当前用户对其有RW权限的所有队列的视图。

图 3.10. 服务人员队列视图（默认）

## QueueView: My Queues



队列概览提供了OTRS中日常工作的多种选项。第一个选项是‘我的队列’。在服务人员偏好设置中，或当登录为系统管理员时可以定义一个服务人员所属的多个队列。当访问‘工单’->‘队列视图’菜单时，所有的工单都会出现在这个默认视图。

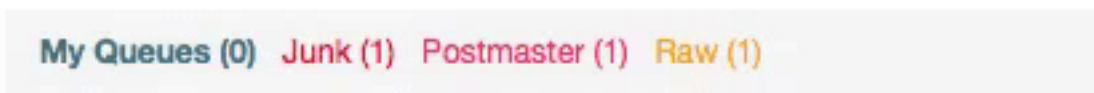
队列视图的第二个选项是一个向下导航到单独的队列和子队列，包含了在队列中的工单。

在这两种类型的视图中，用户还能看到所有锁定的工单（这是默认的过滤器），或者所有可用的工单。工单必须是下面的可视状态类型才能在队列视图中显示：‘处理中’、‘新建’、‘挂起提醒’、‘挂起自动（等待成功/失败关闭）’。

有可视警告信号可以帮助用户。

图 3.11. 服务人员队列视图的可视警告信号

## QueueView: My Queues



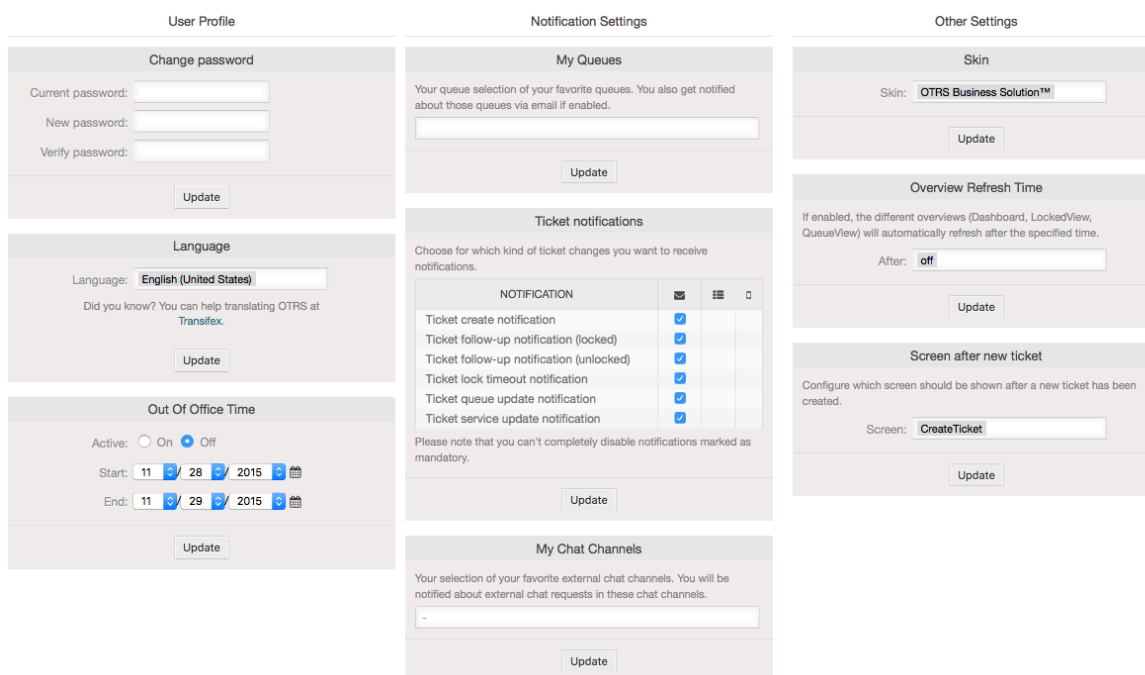
### 可视警告信号

- 第1个高亮阶段：设置（一级）寿命时间（单位：分钟）来高亮显示包含未处理工单的队列。如上图中的“Raw”队列。
- 第2个高亮阶段：设置（一级）寿命时间（单位：分钟）来高亮显示包含未处理工单的队列。如上图中的“Postmaster”队列。
- 闪烁：为包含最老的工单的队列激活闪烁机制。不是所有浏览器都支持这个闪烁功能，不支持闪烁时会显示为红色，如上图中的“Junk”队列。
- 加粗：当前队列名称会被加粗显示，如上图中的“我的队列”。

## 9. 用户偏好设置

OTRS用户如客户联系人、服务人员 and OTRS系统管理员可以按他们的需要配置帐户偏好设置。服务人员可以通过点击WEB界面顶部的齿轮图标（如下图）来访问配置窗口，客户联系人则必须点击“偏好设置”链接（见下图）。

图 3.12. 服务人员的个人偏好设置



The screenshot shows the OTRS user profile settings page, organized into three main sections:

- User Profile:**
  - Change password:** Fields for Current password, New password, and Verify password, with an Update button.
  - Language:** A dropdown menu set to 'English (United States)', a note about translation, and an Update button.
  - Out Of Office Time:** Radio buttons for 'Active' (Off) and 'Inactive' (On), and date/time pickers for Start (11:28, 2015) and End (11:29, 2015), with an Update button.
- Notification Settings:**
  - My Queues:** A text input field for selecting favorite queues, with an Update button.
  - Ticket notifications:** A table to select notification types. All checkboxes are checked.

NOTIFICATION	☐	☑	☑	☑
Ticket create notification	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Ticket follow-up notification (locked)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Ticket follow-up notification (unlocked)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Ticket lock timeout notification	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Ticket queue update notification	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Ticket service update notification	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
  - My Chat Channels:** A text input field for selecting external chat channels, with an Update button.
- Other Settings:**
  - Skin:** A dropdown menu set to 'OTRS Business Solution™', with an Update button.
  - Overview Refresh Time:** A text input field set to 'off', with an Update button.
  - Screen after new ticket:** A dropdown menu set to 'CreateTicket', with an Update button.

一个服务人员可以配置3个类型的偏好设置：个人设置、邮件设置和其它设置。默认可用项如下：

### 个人设置

- 修改当前密码。
- 切换界面语言。
- 配置与激活“不在办公室”的时间。
- 修改前端界面主题。

### 通知设置

- 选择你想要监控的队列到“我的队列”中。
- 选择你想要监控的服务到“我的服务”中。
- 配置你想要接收的工单通知（按传输方法）。

### 其它设置

- 切换前端界面皮肤。
- 设置各种概览视图的刷新周期（仪表盘、队列视图等等）。
- 配置创建新工单后显示的界面。

图 3.13. 客户联系人的个人偏好设置

### Example Company

Tickets

#### Interface language

Language

#### Ticket overview

Refresh interval

#### Number of displayed tickets

Tickets per page

#### Change password

Current password

New password

Verify password

一个客户联系人可以选择WEB界面的语言、设置工单概览的刷新周期和选择显示工单的最大数，还可以设置一个新的密码。



# 第 4 章 管理

## 1. OTRS 系统管理区

### 1.1. 基本介绍

通过OTRS WEB界面访问系统管理页面，下列的系统配置设置可用于OTRS系统管理员：添加服务人员、客户和队列，工单和邮件设置，安装附加软件包如FAQ知识库和ITSM等等。

admin组成员的服务人员能够通过导航栏的系统管理链接访问系统管理区（如下图）。没有足够权限的服务人员不能访问这个链接。

图 4.1. OTRS系统管理概览窗口

<b>Agent Management</b> <b>Agents</b> Create and manage agents. <b>Agents &lt;-&gt; Groups</b> Link agents to groups. <b>Agents &lt;-&gt; Roles</b> Link agents to roles. <b>Groups</b> Create and manage groups. <b>Roles</b> Create and manage roles. <b>Roles &lt;-&gt; Groups</b> Link roles to groups.	<b>Customer Management</b> <b>Customer User</b> Create and manage customer users. <b>Customer User &lt;-&gt; Groups</b> Link customer user to groups. <b>Customers</b> Create and manage customers. <b>Customer User &lt;-&gt; Services</b> Link customer user to services.	<b>Email Settings</b> <b>PostMaster Mail Accounts</b> Manage POP3 or IMAP accounts to fetch email from. <b>Email Addresses</b> Set sender email addresses for this system. <b>PGP Keys</b> Manage PGP keys for email encryption. <b>PostMaster Filters</b> Filter incoming emails. <b>S/MIME Certificates</b> Manage S/MIME certificates for email encryption.
<b>Queue Settings</b> <b>Queues</b> Create and manage queues. <b>Templates &lt;-&gt; Queues</b> Link templates to queues. <b>Auto Responses &lt;-&gt; Queues</b> Link queues to auto responses. <b>Attachments &lt;-&gt; Templates</b> Link attachments to templates. <b>Signatures</b> Create and manage signatures. <b>Templates</b> Create and manage templates. <b>Auto Responses</b> Create and manage responses that are automatically sent. <b>Attachments</b> Create and manage attachments. <b>Salutations</b> Create and manage salutations.	<b>Ticket Settings</b> <b>Ticket Notifications</b> Create and manage ticket notifications. <b>Access Control Lists (ACL)</b> Configure and manage ACLs. <b>Priorities</b> Create and manage ticket priorities. <b>Dynamic Fields</b> Create and manage dynamic fields. <b>Types</b> Create and manage ticket types. <b>States</b> Create and manage ticket states. <b>Services</b> Create and manage services. <b>Service Level Agreements</b> Create and manage Service Level Agreements (SLAs).	<b>System Administration</b> <a href="#">Online Admin Manual</a> <b>GenericAgent</b> Manage tasks triggered by event or time based execution. <b>OTRS Business Solution™</b> Deploy and manage OTRS Business Solution™. <b>Cloud Services</b> Manage OTRS Group cloud services. <b>Session Management</b> Manage existing sessions. <b>Performance Log</b> View performance benchmark results. <b>SQL Box</b> Execute SQL statements. <b>SysConfig</b> Edit the system configuration settings. <b>Package Manager</b> Update and extend your system with software packages. <b>System Registration</b> Manage system registration. <b>Support Data Collector</b> Manage support data. <b>Admin Notification</b> Send notifications to users. <b>System Maintenance</b> Schedule a maintenance period. <b>System Log</b> View system log messages. <b>Process Management</b> Configure Processes. <b>Web Services</b> Create and manage web services. <b>Chat Channel</b> Create and manage chat channels.

### 1.2. 服务人员、组和角色

#### 1.2.1. 服务人员

点击服务人员链接后，你就OTRS的服务人员管理窗口（如下图）。系统管理员可以添加、修改或‘去激活’服务人员帐户。此外还能管理服务人员的偏好设置包括个别服务人员界面的语言和通知设置。

#### 注意

一个OTRS服务人员帐户可以‘去激活’但不能删除。‘去激活’通过设置有效性标记为无效或临时无效来实现。

图 4.2. 服务人员管理

**Agent Management**

Actions

Wildcards like "\*" are allowed.

+ Add agent

Hint

Agents will be needed to handle tickets.  
Attention: Don't forget to add a new agent to groups and/or roles!

USERNAME	NAME	EMAIL	LAST LOGIN	VALIDITY	CHANGED	CREATED
carlos.garcia	Carlos Garcia	carlos.garcia@mycompany...	11/26/2015 12:10	valid	11/25/2015 13:25	11/25/2015 13:25
carlos.rodriguez	Carlos Rodriguez	carlos.rodriguez@mycomp...		valid	11/25/2015 13:25	11/25/2015 13:25
dennis.schmelter	Dennis Schmelter	dennis.schmelter@mycomp...		valid	11/25/2015 13:25	11/25/2015 13:25
dominik.klein	Dominik Klein	dominik.klein@mycompany...		valid	11/25/2015 13:25	11/25/2015 13:25
jan.steinweg	Jan Steinweg	jan.steinweg@mycompany.com		valid	11/25/2015 13:25	11/25/2015 13:25
jens.pfeifer	Jens Pfeifer	jens.pfeifer@mycompany.com		valid	11/25/2015 13:25	11/25/2015 13:25
manuel.hecht	Manuel Hecht	manuel.hecht@mycompany.com		valid	11/25/2015 13:25	11/25/2015 13:25
marc.bonsels	Marc Bonsels	marc.bonsels@mycompany.com		valid	11/25/2015 13:25	11/25/2015 13:25
marco.buchholz	Marco Buchholz	marco.buchholz@mycompan...		valid	11/25/2015 13:25	11/25/2015 13:25
martin.gruner	Martin Gruner	martin.gruner@mycompany...		valid	11/25/2015 13:25	11/25/2015 13:25
oliver.rottges	Oliver Rottges	oliver.rottges@mycompan...		valid	11/25/2015 13:25	11/25/2015 13:25
patrick.brischler	Patrick Brischler	patrick.brischler@mycom...		valid	11/25/2015 13:25	11/25/2015 13:25
root@localhost	Admin OTRS	root@localhost	11/26/2015 11:58	valid	11/25/2015 13:25	11/25/2015 13:25
udo.bretz	Udo Bretz	udo.bretz@mycompany.com		valid	11/25/2015 13:25	11/25/2015 13:25

要注册一个服务人员，点击“添加服务人员”按钮，输入必填数据并点击窗口底部的‘提交’按钮，如下图所示。

图 4.3. 添加一个新服务人员

**Agent Management**

Actions

Go to overview

Hint

Agents will be needed to handle tickets.  
Attention: Don't forget to add a new agent to groups and/or roles!

**Add Agent**

Title: Mr.

\* Firstname: Carlos

\* Lastname: Garcia

\* Username: carlos.garcia

Password:   
Will be auto-generated if left empty.

\* Email: carlos.garcia@mycompany.com

Mobile:   
Validity: valid

Google Authenticator:   
Enter your shared secret to enable two factor authentication.

Language: English (United States)

Out Of Office Time:  On  Off

Start: 12 / 11 / 2015

End: 12 / 12 / 2015

Skin: OTRS Business Solution™

新的服务人员帐户创建后，你还设置服务人员到一个或多个组/角色。关于组和角色的信息在本章的组和角色部分。

### 1.2.2. 组

每个服务人员帐户应该至少属于一个组或角色。在一个全新安装的系统中，有三个预告定义的组可用，如表4-1所示。

表 4.1. 全新安装OTRS可用的默认组

组	描述
admin	允许执行系统中的管理任务。
stats	有资格访问OTRS的统计模块并生成统计报表。
users	服务人员应该属于这个组且有读写权限。他们能访问工单系统的所有功能。

### 注意

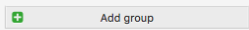
在全新安装的OTRS中，服务人员'root@localhost'默认属于'admin'、'stats'和'users'组。

你可以在系统管理区通过点击组链接访问组管理页面（见下图）

**图 4.4. 组管理**

**Group Management**

**Actions**



**Hint**

The admin group is to get in the admin area and the stats group to get stats area.  
 Create new groups to handle access permissions for different groups of agent (e. g. purchasing department, support department, sales department, ...).  
 It's useful for ASP solutions.

**List**

NAME	COMMENT	VALIDITY	CHANGED	CREATED
admin	Group for all administrators.	valid	11/19/2015 13:25	11/19/2015 13:25
stats	Group for statistics access.	valid	11/19/2015 13:25	11/19/2015 13:25
users	Group for default access.	valid	11/19/2015 13:25	11/19/2015 13:25

## 注意

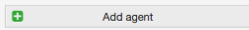
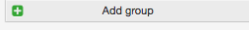
与服务人员一样，OTRS的组只能‘去激活’而不能删除。‘去激活’通过设置有效性标记为无效或临时无效来实现。

要添加一个服务人员到一个组中，或者修改这个服务人员所属的组，可以使用系统管理页面中的 服务人员 <-> 组链接（见下图）。

**图 4.5. 服务人员 <-> 组管理**

**Manage Agent-Group Relations**

**Actions**

**Filter for Agents**

Just start typing to filter...

**Filter for Groups**

Just start typing to filter...

**Overview**

**AGENTS**

- carlos.garcia (Carlos Garcia)
- carlos.rodriguez (Carlos Rodríguez)
- dennis.schmelter (Dennis Schmelter)
- dominik.klein (Dominik Klein)
- jan.steinweg (Jan Steinweg)
- jens.pfeifer (Jens Pfeifer)
- manuel.hecht (Manuel Hecht)
- marc.bonsels (Marc Bonsels)
- marco.buchholz (Marco Buchholz)
- martin.gruner (Martin Gruner)
- oliver.rottgies (Oliver Rottgies)
- patrick.brischler (Patrick Brischler)
- udo.bretz (Udo Bretz)

**GROUPS**

- admin
- stats
- users

系统中所有组和服务人员概览显示在这个页面上。你还可以使用过滤器来查找一个特定的条目。如果你想修改一个服务人员所属的组，只需点击服务人员的名称（见下图）。要修改一个组关联的服务人员，只需点击你想编辑的组（见下图）。

图 4.6. 修改一个服务人员所属的组

**Manage Agent-Group Relations**

Actions

Filter

Change Group Relations for Agent Dominik Klein (dominik.klein)

GROUP	<input type="checkbox"/> CHAT_OBSERVER	<input type="checkbox"/> CHAT_PARTICIPANT	<input type="checkbox"/> CHAT_OWNER	<input type="checkbox"/> RO	<input type="checkbox"/> MOVE_INTO	<input type="checkbox"/> CREATE	<input type="checkbox"/> NOTE	<input type="checkbox"/> OWNER
admin	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
stats	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
users	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

or

Reference

**ro**  
Read only access to the ticket in this group/queue.

**move\_into**  
Permissions to move tickets into this group/queue.

**create**  
Permissions to create tickets in this group/queue.

**note**  
Permissions to add notes to tickets in this group/queue.

**owner**  
Permissions to change the owner of tickets in this group/queue.

**priority**  
Permissions to change the ticket priority in this group/queue.

**rw**  
Full read and write access to the tickets in this group/queue.

图 4.7. 修改一个组关联的服务人员

**Manage Agent-Group Relations**

Actions

Filter

Change Agent Relations for Group users

AGENT	<input type="checkbox"/> CHAT_OBSERVER	<input type="checkbox"/> CHAT_PARTICIPANT	<input type="checkbox"/> CHAT_OWNER	<input type="checkbox"/> RO	<input type="checkbox"/> MOVE_INTO	<input type="checkbox"/> CREATE	<input type="checkbox"/> NOTE	<input type="checkbox"/> OWNER	<input type="checkbox"/> PRIORITY	<input type="checkbox"/> RW
carlos.garcia (Carlos Garcia)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
carlos.rodriguez (Carlos Rodriguez)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
dennis.schmelter (Dennis Schmelter)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
dominik.klein (Dominik Klein)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
jan.steinweg (Jan Steinweg)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
jens.pfeifer (Jens Pfeifer)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
manuel.hecht (Manuel Hecht)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
marc.bonsels (Marc Bonsels)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
marco.buchholz (Marco Buchholz)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
martin.gruner (Martin Gruner)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
oliver.rottges (Oliver Rottges)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
patrick.brischler (Patrick Brischler)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
root@localhost (Admin OTRS)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
udo.bretz (Udo Bretz)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

or

每个组都有一套相关的权限，每个组成员（服务人员）可能有这些权限的某种组合。权限清单如表 4-2 所示。

表 4.2. OTRS 组相关的权限

权限	描述
聊天观察者	服务人员可以在一个聊天中潜水（OTRS 商业版功能）。
聊天参与者	服务人员可以正常参与一个聊天（OTRS 商业版功能）。
聊天所有者	服务人员一个聊天的所有权并能接受聊天请求（OTRS 商业版功能）。
ro-只读	对这个组的工单、记录和队列有只读访问权限。
move into-转移到	在属于本组的队列或区域间转移工单或记录的权限。

权限	描述
create-创建	在这个组的队列或区域中创建工单或记录的权限。
note	Right to add notes to tickets or entries in the queues or areas of this group.
owner-所有者	在本组的队列或区域中更新工单或记录的所有者的权限。
priority-优先级	在本组的队列或区域中更改工单或记录的优先级的权限。
rw-读写	对属于这个组的队列或区域的工单、记录和队列完全读写访问权限。

## 注意

队列视图默认只列出一个服务人员有rw-读写权限的队列中工单。如果你想改变这个情况，你可以设置选项 `Ticket::Frontend::AgentTicketQueue###ViewAllPossibleTickets` 为是。

默认情况下不会显示所有可用的权限。这些附加的权限可以添加。

**表 4.3.** 附加的权限组

权限	描述
stats	访问统计页面的权限
退回	退回一个邮件消息的权限（使用工单详情窗口的‘退回’按钮）。
编写	给一个工单编写答复的权限。
客户	变更一个工单的客户权限。
转发	转发一个消息的权限（使用‘转发’按钮）。
挂起	设置一个工单为挂起状态的权限。
电话	添加一个电话沟通到工单的权限。
负责人	变更一个工单的负责人的权限。

## 注意

这些权限可以通过修改配置 `System::Permission` 添加进来。

### 1.2.3. 角色

角色是一个强大的功能，以一种非常简单、快速的方式管理大量服务人员的存取权限。在拥有大量服务人员、组和队列的大型、复杂的支持系统中特别有用。下面的例子解释了何时应该使用角色。

假设你的系统有100个服务人员，其中有90个需要访问一个处理所有支持请求、叫“support”的队列。这个“support”队列包含多个子队列。其它的10个服务人员有访问系统所有队列的权限。这10个服务人员分派工单，关注‘raw’队列并转移垃圾消息到‘junk’队列。

公司现在设置了一个销售某些产品的新部门。必须处理订单请求和接受、订单确认、开帐单等事宜，假定公司的一部分服务人员使用OTRS来做这些事情。不同的服务人员需要能够访问必须创建的新队列。

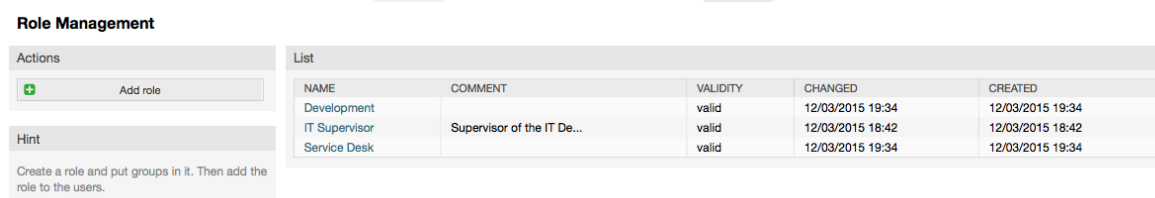
因为手动逐个修改服务人员的访问权限需要花费大量时间，可以创建角色来定义不同的访问级别。服务人员就可以添加到一个或多个角色中，然后他们的访问权限就自动修改了。如果创建了一个新的服务人员帐户，还能将这个帐户添加到一个或多个角色。

## 注意

当处理复杂的组织结构和维护大型的OTRS安装时，角色才真正好用。可是建议要适当注意：混合了服务人员-组和服务人员-角色的映射可能造成复杂的访问控制方案，难于理解和维护。如果你想只使用角色并禁用系统管理区服务人员<->组的选项，可以通过修改系统配置中的参数Frontend::Module###AdminUserGroup来实现。要注意这不会移除已经存在的服务人员到组的分配！

可以在系统管理页面点击角色链接来访问角色管理部分（见下图）。

图 4.8. 角色管理



**Role Management**

Actions

Hint

Create a role and put groups in it. Then add the role to the users.

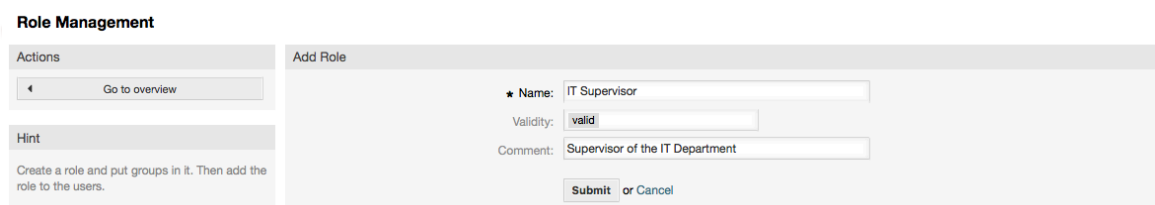
NAME	COMMENT	VALIDITY	CHANGED	CREATED
Development		valid	12/03/2015 19:34	12/03/2015 19:34
IT Supervisor	Supervisor of the IT De...	valid	12/03/2015 18:42	12/03/2015 18:42
Service Desk		valid	12/03/2015 19:34	12/03/2015 19:34

## 注意

和服务人员、组一样，角色一旦创建就只能‘去激活’而不能删除。要‘去激活’，设置有效性选项为无效或临时无效。

显示了系统中所有角色的一个概览。要编辑角色的设置，点击角色的名称即可。在一个全新安装的OTRS系统中，默认没有定义任何角色。要注册一个，点击“添加角色”按钮，提供需要的数据并提交（如下图）。

图 4.9. 添加一个新的角色



**Role Management**

Actions

Hint

Create a role and put groups in it. Then add the role to the users.

Add Role

★ Name:

Validity:

Comment:

or

要获得系统中所有角色和服务人员概览，在系统管理页面点击‘角色<->服务人员’链接。你还可以使用过滤器来查找一个特定元素。如果你想修改一个服务人员关联的角色，只需点击这个服务人员的姓名（见下图）。要修改一个角色关联的服务人员，点击你想编辑的角色名称（见下图）。

图 4.10. 修改一个服务人员关联的角色



**Manage Role-Agent Relations**

Actions

Filter

Change Role Relations for Agent Jan Steinweg (jan.steinweg)

ROLE	<input type="checkbox"/> ACTIVE
Development	<input checked="" type="checkbox"/>
IT Supervisor	<input type="checkbox"/>
Service Desk	<input type="checkbox"/>

or

图 4.11. 修改一个特定角色关联的服务人员

**Manage Role-Agent Relations**

Actions

Filter

Change Agent Relations for Role Development

AGENT	ACTIVE
carlos.garcia (Carlos Garcia)	<input checked="" type="checkbox"/>
carlos.rodriguez (Carlos Rodriguez)	<input checked="" type="checkbox"/>
dennis.schmelter (Dennis Schmelter)	<input checked="" type="checkbox"/>
dominik.klein (Dominik Klein)	<input checked="" type="checkbox"/>
jan.steinweg (Jan Steinweg)	<input checked="" type="checkbox"/>
jens.pfeifer (Jens Pfeifer)	<input checked="" type="checkbox"/>
manuel.hecht (Manuel Hecht)	<input checked="" type="checkbox"/>
marc.bonsels (Marc Bonsels)	<input checked="" type="checkbox"/>
marco.buchholz (Marco Buchholz)	<input checked="" type="checkbox"/>
martin.gruner (Martin Gruner)	<input checked="" type="checkbox"/>
oliver.rottges (Oliver Rottges)	<input checked="" type="checkbox"/>
patrick.brischler (Patrick Brischler)	<input checked="" type="checkbox"/>
udo.bretz (Udo Bretz)	<input checked="" type="checkbox"/>

or

要获得系统中所有角色和组的概览，在系统管理页面点击‘角色<->组’链接。你会看到一个类似下图的窗口。你还可以使用过滤器来查找一个特定的条目。

图 4.12. 管理角色-组的关系

**Manage Role-Group Relations**

Filter for Roles

Filter for Groups

Overview

ROLES	GROUPS
Development	admin
IT Supervisor	stats
Service Desk	users

要定义一个角色的不同访问权限，点击一个角色或一个组名称（分别如图4.13和4.14所示）。

图 4.13. 修改一个角色的组关系

**Manage Role-Group Relations**

Actions

Filter

Hint  
 Select the role/group permissions. If nothing is selected, then there are no permissions in this group (tickets will not be available for the role).

Change Group Relations for Role Service Desk

GROUP	<input type="checkbox"/> CHAT_OBSERVER	<input type="checkbox"/> CHAT_PARTICIPANT	<input type="checkbox"/> CHAT_OWNER	<input type="checkbox"/> RO	<input type="checkbox"/> MOVE_INTO	<input type="checkbox"/> CREATE	<input type="checkbox"/> NOTE	<input type="checkbox"/> OWNER	<input type="checkbox"/> PRIORITY
admin	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
stats	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
users	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

or

Reference

**ro**  
 Read only access to the ticket in this group/queue.

**move\_into**  
 Permissions to move tickets into this group/queue.

**create**  
 Permissions to create tickets in this group/queue.

**note**  
 Permissions to add notes to tickets in this group/queue.

**owner**  
 Permissions to change the owner of tickets in this group/queue.

**priority**  
 Permissions to change the ticket priority in this group/queue.

**rw**  
 Full read and write access to the tickets in this group/queue.

图 4.14. 修改一个组的角色关系

**Manage Role-Group Relations**

Change Role Relations for Group stats

Actions:

Filter:

Hint: Select the role/group permissions. If nothing is selected, then there are no permissions in this group (tickets will not be available for the role).

ROLE	<input type="checkbox"/> CHAT_OBSERVER	<input type="checkbox"/> CHAT_PARTICIPANT	<input type="checkbox"/> CHAT_OWNER	<input type="checkbox"/> RO	<input type="checkbox"/> MOVE_INTO	<input type="checkbox"/> CREATE	<input type="checkbox"/> NOTE	<input type="checkbox"/> OWNER	<input type="checkbox"/> PRIORITY
Development	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
IT Supervisor	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Service Desk	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

or

Reference

**ro**  
Read only access to the ticket in this group/queue.

**move\_into**  
Permissions to move tickets into this group/queue.

**create**  
Permissions to create tickets in this group/queue.

**note**  
Permissions to add notes to tickets in this group/queue.

**owner**  
Permissions to change the owner of tickets in this group/queue.

**priority**  
Permissions to change the ticket priority in this group/queue.

**rw**  
Full read and write access to the tickets in this group/queue.

## 1.3. 客户与客户组

### 1.3.1. 客户

OTRS支持不同类型的用户。使用“客户”链接（通过导航栏或系统管理页面），你可以管理你的客户联系人帐户（如下图），客户联系人用于通过客户界面（customer.pl）登录系统。你的客户联系人通过客户界面不仅能够创建工单，还能够回顾他们过去的工单以发起新的更新。重要的是要知道在系统的工单历史中需要一个客户联系人。

图 4.15. 客户管理

**Customer User Management**

Actions:

Wildcards like "\*" are allowed.

Hint: Customer user are needed to have a customer history and to login via customer panel.

USERNAME	NAME	EMAIL	CUSTOMERID	LAST LOGIN	VALIDITY
han.solo	Mr. Han Solo	han.solo@testcustomer.com	SWVII		valid
kylor.ren	Mr. Kylo Ren	kylor.ren@testcustomer.com	SWVII		valid
luke.skywalker	Mr. Luke Skywalker	luke.skywalker@testcustomer.com	SWVII		valid
poe.dameron	Mr. Poe Dameron	poe.dameron@testcustomer.com	SWVII		valid

你可以搜索一个已注册的客户联系人，或者通过点击他们的姓名来编辑他们的设置。你还可以修改客户后端，进一步信息请参考 外部后端部分。

要创建一个新的客户联系人帐户，点击“添加客户联系人”按钮（如下图）。一些字段是强制的，例如它们必须包含值，所以如果你将这些字段留空，就会用红色高亮显示错误信息。



图 4.16. 添加一个客户联系人

**Customer User Management**

Actions

Go to overview

Back to search results

Hint

Customer user are needed to have a customer history and to login via customer panel.

Add Customer User

Title: Mr

★ Firstname: Han

★ Lastname: Solo

★ Username: han.solo

Password: \*\*\*\*\*

★ Email: han.solo@testcustomer.com

★ CustomerID: SWVII han.solo@testcustomer.com x

Phone:

Fax:

Mobile:

Street:

Zip:

City:

Country:

Comment:

★ Valid: valid

Interface language: English (United States)

客户联系人提供他们的用户名和密码后就能访问OTRS系统。系统需要CustomerID来识别客户联系人和关联的工单。因为邮件地址是唯一的值，可以用来作为CustomerID。

## 注意

与服务人员、组和角色一样，客户联系人不能删除。只能通过设置有效性标记为无效或临时无效来去激活。

### 1.3.2. 客户联系人组

客户联系人还可以添加到一个组，如果你要添加同一个客户单位的联系人以访问一个或几个队列时有用。首先通过组管理模块创建你的客户联系人所属的组。然后添加队列并为此队列选择这个新的组。

下一步是激活客户联系人组支持。这可以通过系统管理中的系统配置参数CustomerGroupSupport来实现。使用参数CustomerGroupAlwaysGroups你可以指定一个新添加的客户联系人的默认组，以便每个新帐户会被自动添加到这些组中。

通过“客户联系人<->组”链接你可以管理一个客户联系人要属于的不同组（见下图）。

图 4.17. 客户联系人-组 关系管理

**Manage Customer-Group Relations**

Actions

Wildcards like "\*" are allowed.

Edit Customer Default Groups

These groups are automatically assigned to all customers. You can manage these groups via the configuration setting "CustomerGroupAlwaysGroups".

Filter for Groups

Just start typing to filter...

Search Results:

CUSTOMERS (4)

Mr. Han Solo <han.solo@testcustomer.com> (SWVII)

Mr. Kylo Ren <kylo.ren@testcustomer.com> (SWVII)

Mr. Luke Skywalker <luke.skywalker@testcustomer.com> (SWVII)

Mr. Poe Dameron <poe.dameron@testcustomer.com> (SWVII)

GROUPS

admin

stats

Customer Default Groups:

GROUPS

users

No changes can be made to these groups.

要给一个客户联系人定义不同的组或者给一个组分配不同的客户联系人，点击相应的客户联系人或组名称（分别如下图4.18和4.19）

图 4.18. 修改一个客户联系人的组

**Manage Customer-Group Relations**

Actions

Go to overview

Edit Customer Default Groups

These groups are automatically assigned to all customers. You can manage these groups via the configuration setting "CustomerGroupAlwaysGroups".

Filter for Groups

Just start typing to filter...

Hint

Select the customer:group permissions. If nothing is selected, then there are no permissions in this group (tickets will not be available for the customer).

Change Group Relations for Customer Mr. Kylo Ren (kylo.ren)

GROUP	<input type="checkbox"/> RO	<input type="checkbox"/> RW
admin	<input type="checkbox"/>	<input type="checkbox"/>
stats	<input type="checkbox"/>	<input type="checkbox"/>

Submit or Cancel

Customer Default Groups:

GROUPS

users

No changes can be made to these groups.

Reference

**ro**  
Read only access to the ticket in this group/queue.

**rw**  
Full read and write access to the tickets in this group/queue.

图 4.19. 修改一个组的客户联系人

**Manage Customer-Group Relations**

Actions

Go to overview

Wildcards like "\*" are allowed.

Hint

Select the customer:group permissions. If nothing is selected, then there are no permissions in this group (tickets will not be available for the customer).

Change Customer Relations for Group stats

CUSTOMER (4)	<input type="checkbox"/> RO	<input type="checkbox"/> RW
Mr. Han Solo <han.solo@testcustomer.com> (SWVII)	<input type="checkbox"/>	<input type="checkbox"/>
Mr. Kylo Ren <kylo.ren@testcustomer.com> (SWVII)	<input type="checkbox"/>	<input type="checkbox"/>
Mr. Luke Skywalker <luke.skywalker@testcustomer.com> (SWVII)	<input type="checkbox"/>	<input type="checkbox"/>
Mr. Poe Dameron <poe.dameron@testcustomer.com> (SWVII)	<input type="checkbox"/>	<input type="checkbox"/>

Submit or Cancel

Reference

**ro**  
Read only access to the ticket in this group/queue.

**rw**  
Full read and write access to the tickets in this group/queue.

## 1.4. 队列

在系统管理页面点击“队列”链接，可以管理系统中的队列（见下图）。在新安装的OTRS系统中有4个默认队列：Raw（原始）、Junk（垃圾）、Misc（杂项）和Postmaster（邮箱管理员）。如果没有定义过滤器规则，所有的进入邮件都将存储在“Raw（原始）”队列。用“Junk（垃圾）”队列存储垃圾邮件。

图 4.20. 队列管理

**Manage Queues**

Actions

Add queue

List

NAME	GROUP	COMMENT	VALIDITY	CHANGED	CREATED
Junk	users	All junk tickets.	valid	11/19/2015 13:25	11/19/2015 13:25
Misc	users	All misc tickets.	valid	11/19/2015 13:25	11/19/2015 13:25
Postmaster	users	Postmaster queue.	valid	11/19/2015 13:25	11/19/2015 13:25
Raw	users	All default incoming ti...	valid	11/19/2015 13:25	11/19/2015 13:25

你可以在这里添加队列（如下图）和修改队列。你可以指定使用这个队列的组，还可以设置本队列为一个现有队列的子队列。

图 4.21. 添加一个新队列

**Manage Queues**

Actions

Add Queue

★ Name:

Sub-queue of:

★ Group:

Unlock timeout minutes:

0 = no unlock - 24 hours = 1440 minutes - Only business hours are counted.  
 If an agent locks a ticket and does not close it before the unlock timeout has passed, the ticket will unlock and will become available for other agents.

Escalation - first response time (minutes):  (Notify by: )

0 = no escalation - 24 hours = 1440 minutes - Only business hours are counted.  
 If there is not added a customer contact, either email-external or phone, to a new ticket before the time defined here expires, the ticket is escalated.

Escalation - update time (minutes):  (Notify by: )

0 = no escalation - 24 hours = 1440 minutes - Only business hours are counted.  
 If there is an article added, such as a follow-up via email or the customer portal, the escalation update time is reset. If there is no customer contact, either email-external or phone, added to a ticket before the time defined here expires, the ticket is escalated.

Escalation - solution time (minutes):  (Notify by: )

0 = no escalation - 24 hours = 1440 minutes - Only business hours are counted.  
 If the ticket is not set to closed before the time defined here expires, the ticket is escalated.

★ Follow up Option:

Specifies if follow up to closed tickets would re-open the ticket, be rejected or lead to a new ticket

你可以定义一个队列的锁定超时时间，如果一个服务人员锁定了一个工单但没有在锁定超时时间内关闭它，这个工单自动解锁以便其它服务人员可以处理。

与队列级别相关的有3个升级时间：

### 首次响应时间升级

- 创建工单后，如果这里定义的时间内没有与客户进行任何邮件或电话沟通，这个工单就会升级。

### 更新时间升级

- 如果一个客户联系人通过邮件或客户门户跟进工单（记录到工单中的才算），更新时间会重置。如果这里定义的时间内没有客户联系，这个工单就会升级。

### 解决时间升级

- 如果这里定义的时间内没有关闭工单，这个工单就会升级。

使用‘一个跟进后锁定工单’，你可以规定一个工单在关闭后被一个跟进重新处理时是否‘锁定’给原来的所有者，这可以确保一个工单的跟进仍由原来处理此工单的服务人员继续处理。

系统邮件地址参数指定这个队列用于外发邮件工单的邮件地址。还可以给一个队列的邮件答复关联一个问候语和一个签名。详细信息请参考邮件地址、问候语和 签名。

## 注意

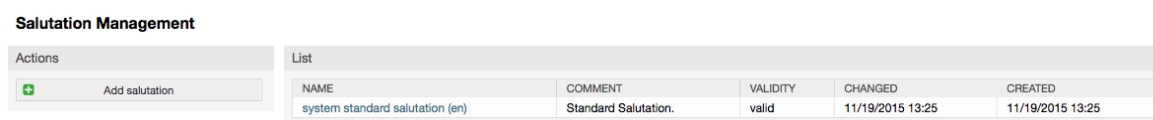
与服务人员、组、角色和客户联系人一样，队列不能删除。只能通过设置有效性标记为无效或临时无效来‘去激活’。

## 1.5. 问候语、签名、附件和模板

### 1.5.1. 问候语

问候语是一个模板的一个文本模块。问候语可以链接到一个或多个队列（参见 队列部分）。问候语仅用于其链接的队列在回复工单的时候。要管理系统中不同的问候语，可以使用系统管理页面中的“问候语”链接（见下图）。

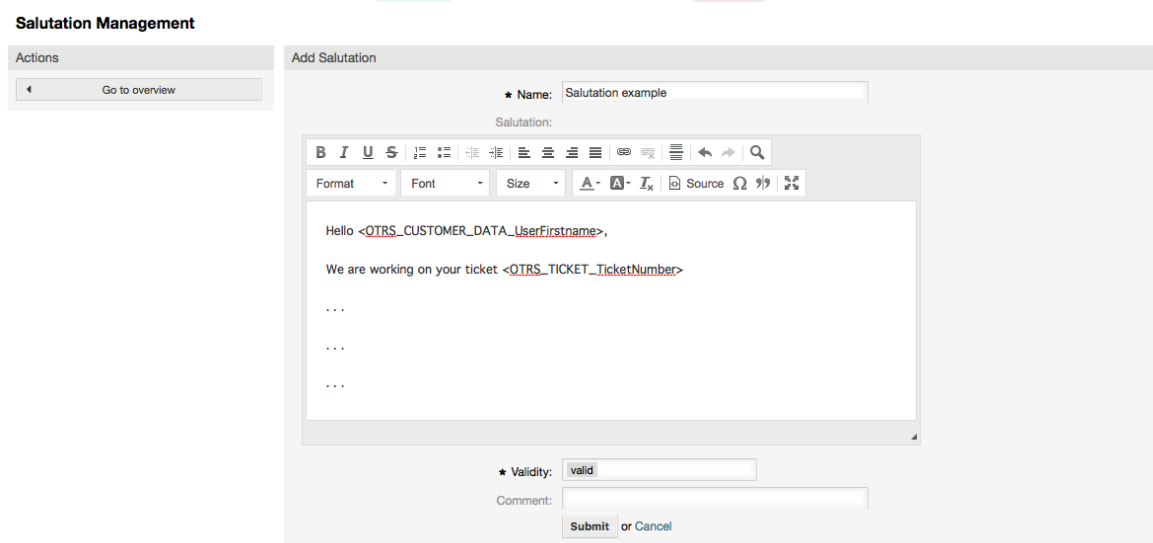
图 4.22. 问候语管理



在一个默认安装的OTRS中有一条可用的问候语“system standard salutation (en)-系统标准问候语（英语）”。

要创建一条新的问候语，点击按钮“添加问候语”，提供需要的数据并提交（见下图）。

图 4.23. 添加一条新的问候语



可以在问候语中使用变量。当你答复一个工单时，变量名将用它们的实际值替换。

可以在模板中使用的不同的变量名可以在问候语窗口的下方找到。例如，如果你使用了变量<OTRS\_LAST\_NAME>，则会在你的答复中包含工单发件人的姓。

## 注意

与OTRS其它条目一样，问候语不能删除，只能通过设置有效性标记为无效或临时无效来‘去激活’。

## 1.5.2. 签名

签名是一个模板的另一个文本模块。问签名可以链接到一个队列（参见 队列部分）。签名仅在一个模板链接到一个队列时才会追加为模板的文本。要管理系统中的签名，可以使用系统管理页面中的“签名”链接（见下图）。

图 4.24. 签名管理



在一个新安装的OTRS中有一个预定义的签名-“system standard signature (en)-系统标准签名（英语）”。

要创建一个新的签名，点击按钮“添加签名”，提供需要的数据并提交（见下图）。

图 4.25. 添加一个新签名

**Signature Management**

Actions

Add Signature

★ Name:

Signature:

**B I U S** [List of icons]  
 Format - Font - Size - [Color picker] [Background color] [Text color] [Source] [Link] [Image]

It is our pleasure to serve you.  
 Receive kind greetings,  
 <OTRS\_RESPONSIBLE\_UserFirstname> <OTRS\_RESPONSIBLE\_UserLastname>

★ Validity:

Comment:

or

Hint

像问候语一样，签名也可以包含动态内容，如答复这个工单的服务人员的姓、名。这也是一样，变量用来替换每个工单签名中的文本内容。查看签名窗口下方可以找到能够使用的变量名。例如：你在一个签名中包含了变量<OTRS\_LAST\_NAME>，则会用答复这个工单的服务人员的姓来替换它。

## 注意

与问候语一样，签名不能删除，只能通过设置有效性标记为无效或临时无效来‘去激活’。

### 1.5.3. 附件

你还可以添加一个或多个附件到一个模板中。如果（编写消息时）选择了一个模板，这个附件就会附加到消息编写窗口。如果有必要的话，服务人员可以在发送给客户之前移除这个附件（不会影响到模板的附件）。

通过系统管理页面的“附件”链接，你可以将附件载入到系统的数据库中（见下图）。

图 4.26. 附件管理

**Attachment Management**

Actions

List

NAME	FILENAME	COMMENT	VALIDITY	CHANGED	CREATED	DELETE
Sample One	mar.jpg		valid	12/03/2015 18:59	12/03/2015 18:59	<input type="button" value="🗑"/>
Sample Two	download.jpg		valid	12/03/2015 18:59	12/03/2015 18:59	<input type="button" value="🗑"/>

要创建一个新的附件，点击按钮“添加附件”，提供需要的数据并提交（见下图）。

图 4.27. 添加一个新附件

**Attachment Management**

Actions

Add Attachment

★ Name:

★ Attachment:  No file selected.

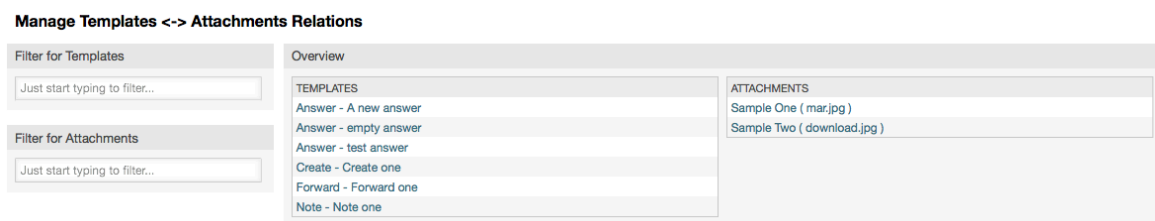
★ Validity:

Comment:

or

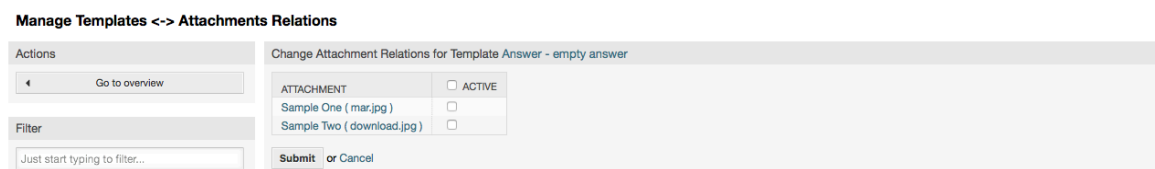
如果一个附件已经保存，它就可以链接到一个或多个模板中。点击系统管理页面的“附件<->模板”链接（见下图）。

**图 4.28. 连接附件到模板中**

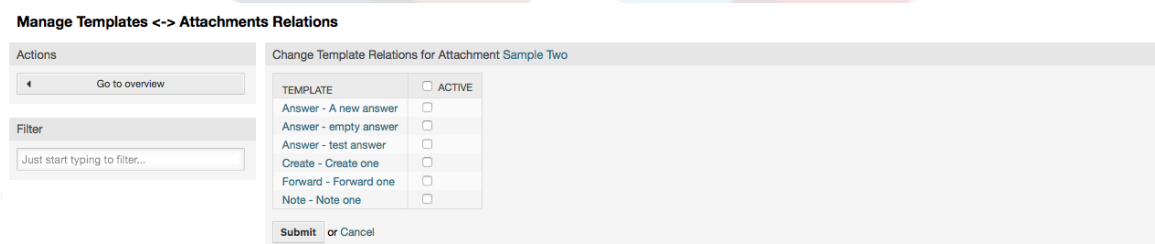


要给一个指定的模板关联不同的附件，或者将一个附件关联到不同的模板，点击相应的模板或附件名称（分别见下面图4.29和4.30）。

**图 4.29. 修改一个模板的附件**



**图 4.30. 修改一个附件关联的模板**



## 1.5.4. 模板

为提升工单处理效率并使用答复样式标准化，你可以在OTRS中定义模板。一个模板可以链接到一个或多个队列，反之亦然。

不同类型的模板用于OTRS的不同部分，它们都有自己的目的，下面是可用的模板类型列表：

- 答复：在答复或回复工单时使用
- 创建：在新的电话或邮件工单中使用
- 转发：在转发一个信件给其他人时使用
- 电话：在接入电话和拨出电话窗口中使用

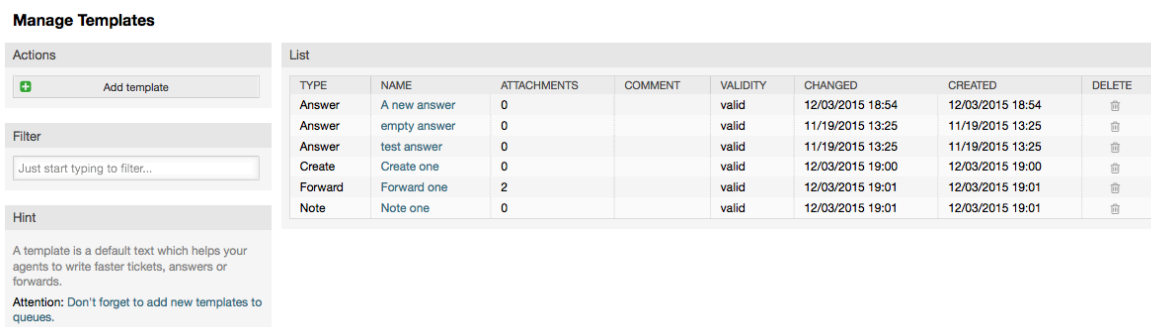
答复模板可以用两种方式访问，工单详情窗口的信件菜单或者一个更快的方式：从任何工单概览详细模式如状态视图或工单视图。在一个新安装的OTRS中，“empty answer-空白答复”模板（答复）设置为每个队列的默认模板。

一旦添加了转发模板并将它分配到队列，在工单详情窗口的‘转发’按钮就会变为转发模板下拉列表框，选项是添加的转发模板，通过选择一个转发模板，弹出的转发窗口就会显示模板中预先设置的文本和附件（类似于答复模板的下拉列表框）。

创建了‘创建’和‘电话’类型的模板后在各自窗口中会出现“文本模板”下拉列表框，从中选择一个模板后会用模板中的“文本”和“附件”重填当前窗口的字段。注意：选择模板后已经修改的文本和附件都会用模板中的内容覆盖。

点击系统管理页面的“模板”链接就能进入模板管理窗口（见下图）。

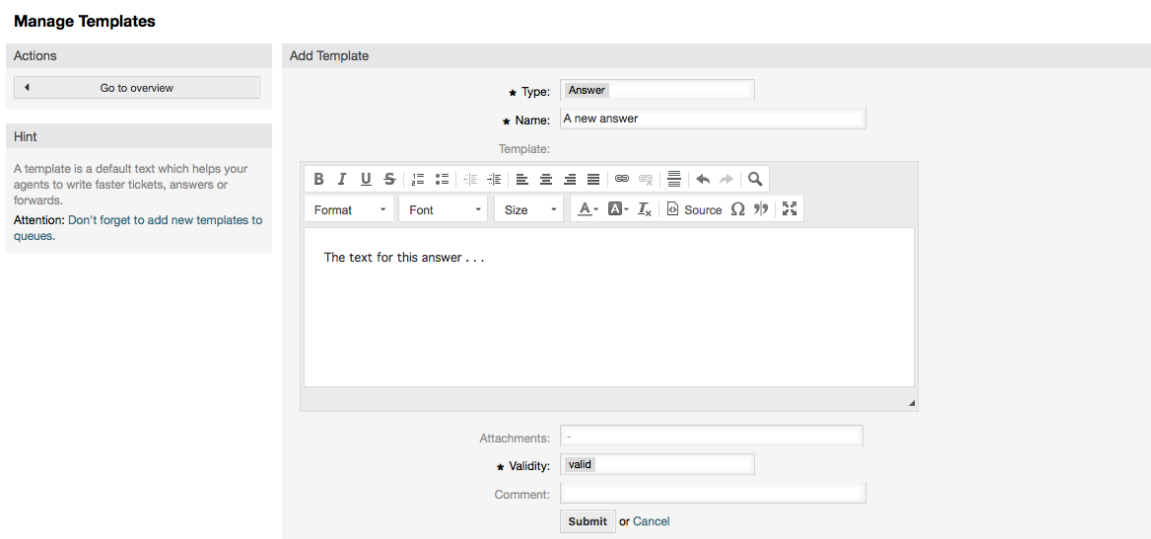
**图 4.31. 模板管理**



TYPE	NAME	ATTACHMENTS	COMMENT	VALIDITY	CHANGED	CREATED	DELETE
Answer	A new answer	0		valid	12/03/2015 18:54	12/03/2015 18:54	
Answer	empty answer	0		valid	11/19/2015 13:25	11/19/2015 13:25	
Answer	test answer	0		valid	11/19/2015 13:25	11/19/2015 13:25	
Create	Create one	0		valid	12/03/2015 19:00	12/03/2015 19:00	
Forward	Forward one	2		valid	12/03/2015 19:01	12/03/2015 19:01	
Note	Note one	0		valid	12/03/2015 19:01	12/03/2015 19:01	

要创建一个新模板，点击“添加模板”按钮，提供需要的数据（确保选择了适当的模板类型）并提交（见下图）。

**图 4.32. 添加一个模板**



要给一个或多个队列添加/移除模板，在系统管理页面点击“模板<->队列”链接（见下图）。你还可以使用过滤器来查找一个特定条目。

**图 4.33. 模板-队列关系管理**



可以给一个队列定义不同的模板，反之亦然，点击相应模板或队列名称（分别见下面的图5.32和5.33）。

图 4.34. 修改一个模板的队列关系

**Manage Template-Queue Relations**

Change Queue Relations for Template Answer - empty answer

Actions: [Go to overview](#)

Filter:

QUEUE	<input type="checkbox"/> ACTIVE
Junk	<input checked="" type="checkbox"/>
Misc	<input checked="" type="checkbox"/>
Postmaster	<input checked="" type="checkbox"/>
Raw	<input checked="" type="checkbox"/>

or

图 4.35. 修改一个队列的模板关系

**Manage Template-Queue Relations**

Change Template Relations for Queue Junk

Actions: [Go to overview](#)

Filter:

TEMPLATE	<input type="checkbox"/> ACTIVE
Answer - A new answer	<input type="checkbox"/>
Answer - empty answer	<input checked="" type="checkbox"/>
Answer - test answer	<input type="checkbox"/>
Create - Create one	<input type="checkbox"/>
Forward - Forward one	<input type="checkbox"/>
Note - Note one	<input type="checkbox"/>

or

选择一个模板时，可以添加附加的信息到模板正文中，这取决于模板类型：

‘电话’和‘创建’模板不会添加任何内容到模板正文中，然而新建邮件工单窗口会添加队列分配的签名到邮件正文中（这个窗口有一个单独的框来显示签名）。

选择答复模板时正文还包含队列关联的问候语，然后是模板正文，再然后是工单正文的引用，最后是队列关联的签名。

转发模板与答复模板类似，但不包含问候语部分。

## 1.6. 自动响应

OTRS允许在发生某些事件时发送自动响应给客户联系人，比如在一个特定队列中创建了一个工单、收到相关工单的一个跟进、关闭或拒绝一个工单等等。要管理这些响应，在系统管理页面点击“自动响应”链接（见下图）。

图 4.36. 自动响应管理

**Auto Response Management**

Actions:

NAME	TYPE	COMMENT	VALIDITY	CHANGED	CREATED
default follow-up (after a ticket follow-up has been added)	auto follow up		valid	11/19/2015 13:25	11/19/2015 13:25
default reject (after follow-up and rejected of a closed ticket)	auto reject		valid	11/19/2015 13:25	11/19/2015 13:25
default reject/new ticket created (after closed follow-up with new ticket creation)	auto reply/new ticket		valid	11/19/2015 13:25	11/19/2015 13:25
default reply (after new ticket has been created)	auto reply		valid	11/19/2015 13:25	11/19/2015 13:25

要创建一个自动响应，点击按钮“添加自动响应”，提交需要的数据并提交（见下图）。



图 4.37. 添加一个自动响应



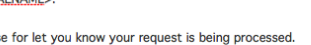
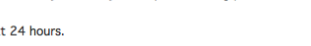

**Auto Response Management**



Actions

★ Name:

★ Subject:

Response:

**B I U S** |  |  |  |  | 

Format - Font - Size -  Source 

Hello <OTRS\_CUSTOMER\_REALNAME>:

This is an automatic response for let you know your request is being processed.

You will have news in the next 24 hours.

Thank you!

Your OTRS Team

★ Type:

★ Auto response from:

★ Validity:

Comment:

or

自动响应的主题和正文都可以通过变量生成，就如签名和问候语中一样。例如：如果你插入变量 <OTRS\_CUSTOMER\_EMAIL[5]>到自动答复的正文中，客户邮件正文的前5行内容就会插入到自动答复中。你可以在图中窗口下方找到有效变量的详细信息。

对每个自动响应，你都可以指定触发它的事件。默认安装后可用的系统事件在表4-4中描述。

表 4.4. 自动响应事件

名称	描述
auto reply-自动回复	在某个队列中创建一个工单时。
自动回复/新工单	重新处理一个已关闭的工单时，比如一个客户联系人回复了这个工单。
auto follow up-自动跟进	接到一个工单的跟进时。
auto reject 自动拒绝	系统自动拒绝一个工单时。
auto remove-自动移除	系统自动删除一个工单时。

### 注意

与其它OTRS条目一样，自动响应也不能被删除，只能通过设置有效性标记为无效或临时无效来‘去激活’。

要添加一个自动响应到一个队列，使用系统管理页面的“自动响应<->队列”链接（见下图）。每个队列会列出所有的系统事件，可以通过列表框选择或移除一个队列的自动响应。

图 4.38. 队列<->自动响应 关系管理

**Manage Queue-Auto Response Relations**

Filter for Queues

Overview

QUEUES	AUTO RESPONSES
Junk	default follow-up (after a ticket follow-up has been added) (auto follow up)
Misc	default reject (after follow-up and rejected of a closed ticket) (auto reject)
Postmaster	default reject/new ticket created (after closed follow-up with new ticket creation) (auto reply/new ticket)
Raw	default reply (after new ticket has been created) (auto reply)
Support	

Filter for Auto Responses

要定义一个队列不同的自动响应，点击相应的队列名称（见下图）。还可以编辑已有的自动响应，点击自动响应名称并像创建一个新的自动响应一样编辑相关设置。

图 4.39. 修改一个队列的自动响应关系

**Manage Queue-Auto Response Relations**

Filter for Queues

Overview

QUEUES	AUTO RESPONSES
Junk	default follow-up (after a ticket follow-up has been added) (auto follow up)
Misc	default reject (after follow-up and rejected of a closed ticket) (auto reject)
Postmaster	default reject/new ticket created (after closed follow-up with new ticket creation) (auto reply/new ticket)
Raw	default reply (after new ticket has been created) (auto reply)

Filter for Auto Responses

## 1.7. 系统邮件地址

要让OTRS发送邮件，你需要一个有效的系统邮件地址。因为许多支持系统需要使用不止一个地址，OTRS可以使用多个邮件地址。一个队列可以链接到多个邮件地址，反之亦然。一个队列外发邮件使用的邮件地址可以在创建队列时设置。使用系统管理页面的“系统邮件地址”链接来管理系统中的所有系统邮件地址（见下图）。

图 4.40. 系统邮件地址管理

**System Email Addresses Management**

Actions

---

Hint

All incoming email with this address in To or Cc will be dispatched to the selected queue.

List

EMAIL ADDRESS	DISPLAY NAME	QUEUE	VALIDITY	CHANGED	CREATED
otrs@localhost	OTRS System	Postmaster	valid	11/19/2015 13:25	11/19/2015 13:25
postmaster@mycompany.com	Postmaster	Junk	valid	01/03/2016 19:16	01/03/2016 19:16
support@mycompany.com	Support Team	Junk	valid	01/03/2016 19:15	01/03/2016 19:15

如果你创建一个新的系统邮件地址（见下图），你可以选择它链接的队列或子队列。这个链接允许系统对进入的消息根据邮件的收件人字段邮件地址分类到正确的队列。

图 4.41. 添加一个系统邮件地址

**System Email Addresses Management**

Actions

---

Hint

All incoming email with this address in To or Cc will be dispatched to the selected queue.

Add System Email Address

★ Email address:

★ Display name:

The display name and email address will be shown on mail you send.

★ Queue:

★ Validity:

Comment:

or

### 注意

与其它的OTRS条目一样，系统邮件地址也不能被删除，只能通过设置有效性标记为无效或临时无效来‘去激活’。

## 1.8. 工单通知

OTRS允许在发生某些事件时发送工单通知给服务人员和客户联系人。服务人员可以通过偏好设置定制他们的工单通知设置。

通过系统管理页面的“工单通知”链接，你可以管理系统的工单通知（如下图）。OTRS自带了一系列的预定义通知，涵盖范围广泛的用例。

图 4.42. 工单通知管理

**Ticket Notification Management**

**Actions**

- [Add notification](#)
- [Export Notifications](#)

**Configuration Import**

Here you can upload a configuration file to import Ticket Notifications to your system. The file needs to be in .yml format as exported by the Ticket Notification module.

No file selected.

Overwrite existing notifications?

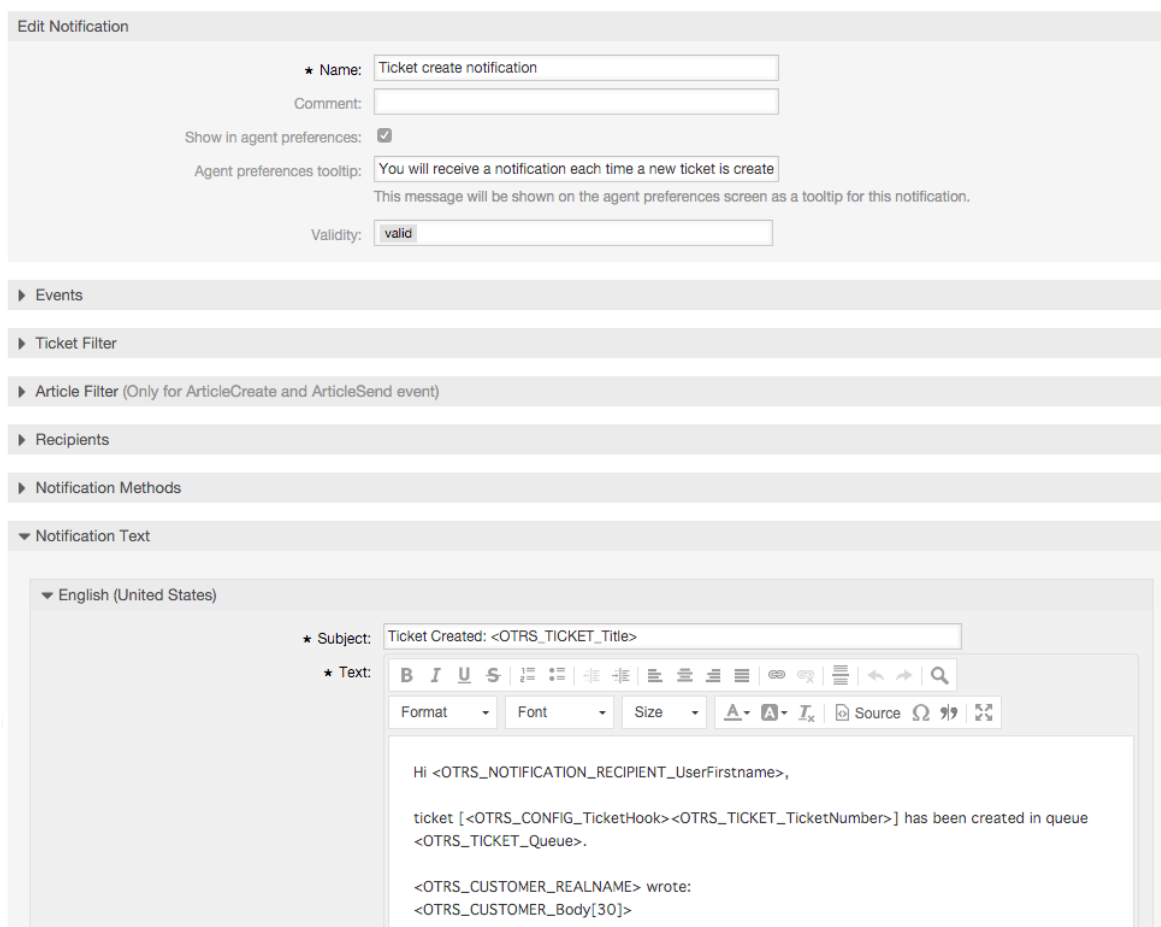
[Import Notification configuration](#)

**List**

NAME	COMMENT	VALIDITY	CHANGED	CREATED	EXPORT	COPY	DELETE
Ticket create notification		valid	11/19/2015 13:25	11/19/2015 13:25			
Ticket escalation notification		valid	11/19/2015 13:25	11/19/2015 13:25			
Ticket escalation warning notification		valid	11/19/2015 13:25	11/19/2015 13:25			
Ticket follow-up notification (locked)		valid	11/19/2015 13:25	11/19/2015 13:25			
Ticket follow-up notification (unlocked)		valid	11/19/2015 13:25	11/19/2015 13:25			
Ticket lock timeout notification		valid	11/19/2015 13:25	11/19/2015 13:25			
Ticket new note notification		valid	11/19/2015 13:25	11/19/2015 13:25			
Ticket owner update notification		valid	11/19/2015 13:25	11/19/2015 13:25			
Ticket pending reminder notification (locked)		valid	11/19/2015 13:25	11/19/2015 13:25			
Ticket pending reminder notification (unlocked)		valid	11/19/2015 13:25	11/19/2015 13:25			
Ticket queue update notification		valid	11/19/2015 13:25	11/19/2015 13:25			
Ticket responsible update notification		valid	11/19/2015 13:25	11/19/2015 13:25			
Ticket service update notification		valid	11/19/2015 13:25	11/19/2015 13:25			

你可以定制通知的许多方面。点击你想修改的通知名称，就会进入编辑通知内容的窗口（见下图）。

图 4.43. 定制一个通知



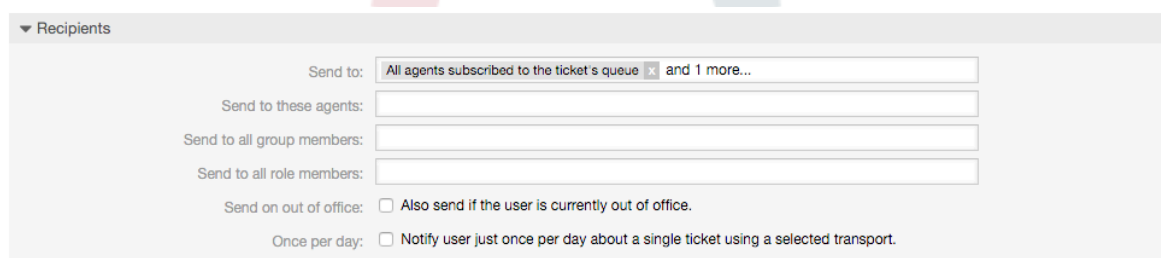
你可以编辑通知的基本信息如通知名称和注释，以及控制服务人员能否接收这个通知（按传输方法）。可为每种语言添加/编辑实际发送的通知的主题和正文。

就如签名和问候语一样，也可以用特殊变量创建通知的动态内容。你可以在通知窗口的底部找到可用的变量清单。

你可以选择触发这个通知的事件，并限制匹配某些条件（工单和/或信件过滤器）的工单使用这个通知。这可以让我们为不同的队列、优先级或其它与系统相关的条件创建不同的通知。

可以按不同的条件（组、角色、个别服务人员等）定义通知的收件人。所有配置的收件人都能收到这个通知。

图 4.44. 定制一个通知的收件人



此外，你还可以指定是否给不在办公室的服务人员发送通知，并限制每个工单每天仅发送一次（例如挂起提醒通知）。

可以用不同的发送方法来发送通知。“邮件”发送方法是OTRS中免费的，使用**OTRS Business Solution™** (OTRS商业版)你还可以在数据库中存储和查看通知（这样使用OTRS就不需要邮件客户端了），并通知短信发送通知（例如：对每个重要通知）。

图 4.45. 定制通知发送方法

▼ Notification Methods

These are the possible methods that can be used to send this notification to each of the recipients. Please select at least one method below.

---

**Email**

Enable this notification method:

Active by default in agent preferences:   
This is the default value for assigned recipient agents who didn't make a choice for this notification in their preferences yet. If the box is enabled, the notification will be sent to such agents.

Additional recipient email addresses:   
You can use OTRS-tags like <OTRS\_TICKET\_DynamicField\_...> to insert values from the current ticket.

Notification article type:   
An article will be created if the notification is sent to the customer or an additional email address.

Email template:   
Use this template to generate the complete email (only for HTML emails).

Enable email security:

Email security level:

If signing key/certificate is missing:

If encryption key/certificate is missing:

---

**Web View**

Enable this notification method:

Active by default in agent preferences:   
This is the default value for assigned recipient agents who didn't make a choice for this notification in their preferences yet. If the box is enabled, the notification will be sent to such agents.

---

**SMS (Short Message Service)**

Enable this notification method:

Active by default in agent preferences:   
This is the default value for assigned recipient agents who didn't make a choice for this notification in their preferences yet. If the box is enabled, the notification will be sent to such agents.

Recipient SMS numbers:   
You can use OTRS-tags like <OTRS\_TICKET\_DynamicField\_...> to insert values from the current ticket.

The field 'Additional recipient email addresses' provides the possibility to use OTRS-tags like <OTRS\_TICKET\_DynamicField\_...> to insert values from the current ticket. The according SMS field (only available in **OTRS Business Solution™**) also has this feature.

Since OTRS 5s Email transport contains security options for each notification, that includes signing and encrypting possibilities with PGP and S/MIME and the opportunity to decide what to do in case of missing key or certificate.

## 1.9. S/MIME

OTRS可以处理进入的S/MIME编码的消息并签名外发邮件。在使用这个功能之前，你需要激活S/MIME并在系统配置中修改一些Crypt::SMIME配置参数。

在系统管理页面的“S/MIME证书”链接允许你管理你的S/MIME证书（见下图）。你可以添加或移除证书，还可以搜索SMIME数据。

**图 4.46. S/MIME管理**

**S/MIME Management:**

**Actions**

+ Add certificate

+ Add private key

**Filter for certificates**

Just start typing to filter...

**Hint**

To show certificate details click on a certificate icon.  
 To manage private certificate relations click on a private key icon.

**Results**

	TYPE	SUBJECT	HASH	FINGERPRINT	CREATE	EXPIRES	DELETE
No data found.							

## 1.10. PGP

OTRS允许使用PGP密钥来加密/解密消息并给外发邮件签名。在使用这个功能之前，你需要激活它并在系统配置中修改一些Crypt::PGP配置参数。

通过系统管理页面的“PGP密钥”链接，可以管理OTRS中使用PGP的用户的密钥环（见下图），例如，本地的OTRS用户或WEB服务器用户。可以添加和移除密钥和签名，以及在密钥环中搜索所有数据。

**图 4.47. PGP管理**

**PGP Management**

**Actions**

+ Add PGP key

**Hint**

In this way you can directly edit the keyring configured in SysConfig.  
 Description: Introduction to PGP

**Result**

TYPE	STATUS	IDENTIFIER	BIT	KEY	FINGERPRINT	CREATED	EXPIRES	DELETE
No data found.								

## 1.11. 状态

通过系统管理页面的“状态”链接，你可以管理在系统中使用的不同的工单状态（见下图）。

**图 4.48. 状态管理**

**State Management**

**Actions**

+ Add state

**Hint**

Attention: Please also update the states in SysConfig where needed.  
 See also: <http://otrs.github.io/doc>

**List**

NAME	TYPE	COMMENT	VALIDITY	CHANGED	CREATED
closed successful	closed	Ticket is closed ...	valid	11/19/2015 13:25	11/19/2015 13:25
closed unsuccessful	closed	Ticket is closed ...	valid	11/19/2015 13:25	11/19/2015 13:25
merged	merged	State for merged ...	valid	11/19/2015 13:25	11/19/2015 13:25
new	new	New ticket create...	valid	11/19/2015 13:25	11/19/2015 13:25
open	open	Open tickets.	valid	11/19/2015 13:25	11/19/2015 13:25
pending auto close+	pending auto	Ticket is pending...	valid	11/19/2015 13:25	11/19/2015 13:25
pending auto close-	pending auto	Ticket is pending...	valid	11/19/2015 13:25	11/19/2015 13:25
pending reminder	pending reminder	Ticket is pending...	valid	11/19/2015 13:25	11/19/2015 13:25
removed	removed	Customer removed ...	valid	11/19/2015 13:25	11/19/2015 13:25

在默认安装后，有一些预定义的状态：

- closed successful-成功关闭
- closed unsuccessful-失败关闭
- merged-已合并
- new-新建

- open-处理中
- 挂起等待成功关闭
- 挂起等待失败关闭
- 挂起提醒
- removed-已删除

每个状态都链接到一个状态类型，需要在创建新状态时指定状态类型。默认的状态类型有：

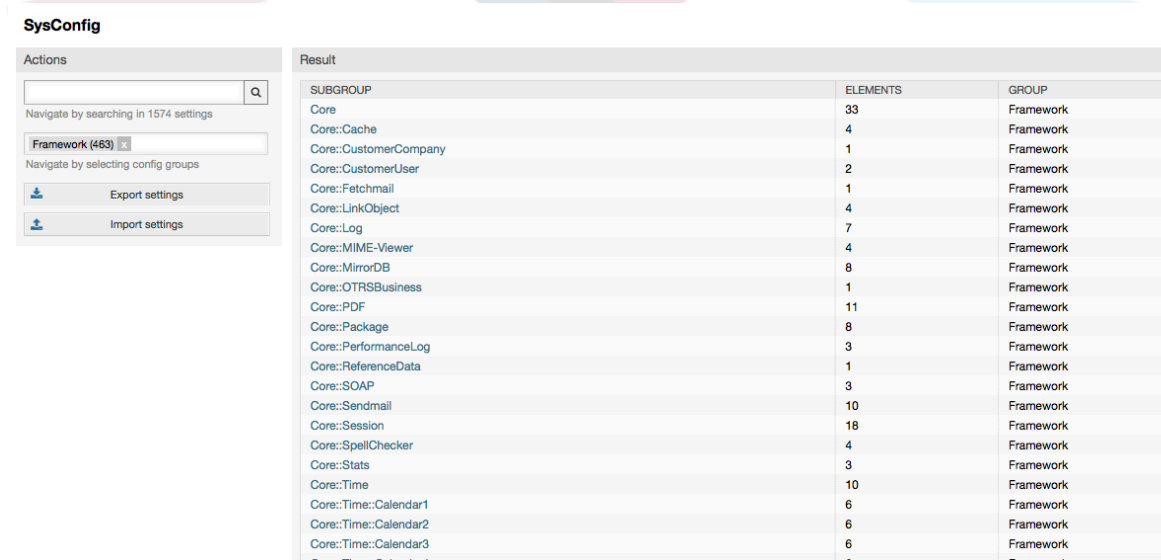
- closed-已关闭
- merged-已合并
- new-新建
- open-处理中
- 自动挂起
- 挂起提醒
- removed-已删除

## 1.12. 系统配置

系统配置就是维护大部分OTRS配置选项的地方。

在系统管理页面的‘系统配置’链接会载入系统配置的图形界面（见下图）。你可以上传你自己的配置文件，也可以备份你当前的设置到一个文件中。通过图形界面可以配置和查看几乎所有的OTRS框架和已安装的应用模块的配置参数。因为所有的配置参数按组和子组分类，这样就可以对大量的现有参数进行快速导航。还可以对所有配置参数执行全文搜索。

图 4.49. 系统配置的图形界面



The screenshot shows the SysConfig web interface. On the left, there is an 'Actions' panel with a search bar containing 'Framework (463)' and buttons for 'Export settings' and 'Import settings'. The main area displays a table with the following columns: SUBGROUP, ELEMENTS, and GROUP.

SUBGROUP	ELEMENTS	GROUP
Core	33	Framework
Core::Cache	4	Framework
Core::CustomerCompany	1	Framework
Core::CustomerUser	2	Framework
Core::Fetchmail	1	Framework
Core::LinkObject	4	Framework
Core::Log	7	Framework
Core::MIME-Viewer	4	Framework
Core::MirrorDB	8	Framework
Core::OTRSBusiness	1	Framework
Core::PDF	11	Framework
Core::Package	8	Framework
Core::PerformanceLog	3	Framework
Core::ReferenceData	1	Framework
Core::SOAP	3	Framework
Core::Sendmail	10	Framework
Core::Session	18	Framework
Core::SpellChecker	4	Framework
Core::Stats	3	Framework
Core::Time	10	Framework
Core::Time::Calendar1	6	Framework
Core::Time::Calendar2	6	Framework
Core::Time::Calendar3	6	Framework
Core::Time::Calendar4	6	Framework

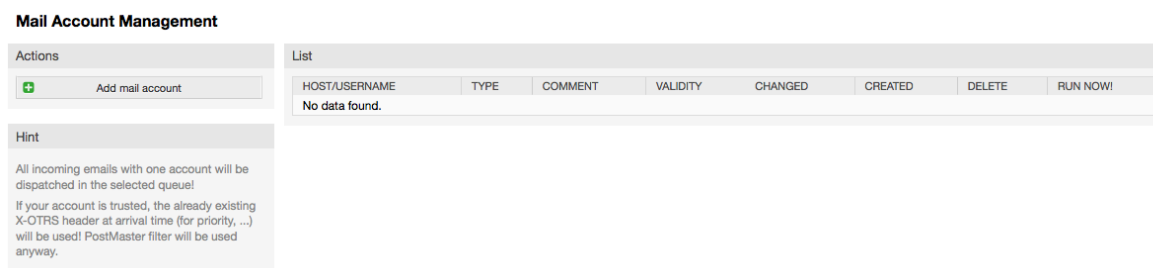
系统配置的图形界面在“通过WEB界面配置系统”部分有详细描述。

## 1.13. 使用邮件帐户

要将一封邮件传输到工单系统中有几种方法。一个方法是使用一个本地的MTA（邮件传输代理），并通过otrs.PostMaster.pl脚本将邮件直接管道传输进工单系统。另一个方法是使用邮件帐户（可以通

过WEB界面管理这些帐户)。在系统管理页面的“邮箱管理员邮件帐户”链接可以管理这些邮件帐户（见下图）。OTRS支持以下邮件传输协议：POP3、POP3S, IMAP和 IMAPS。

图 4.50. 邮件账号管理

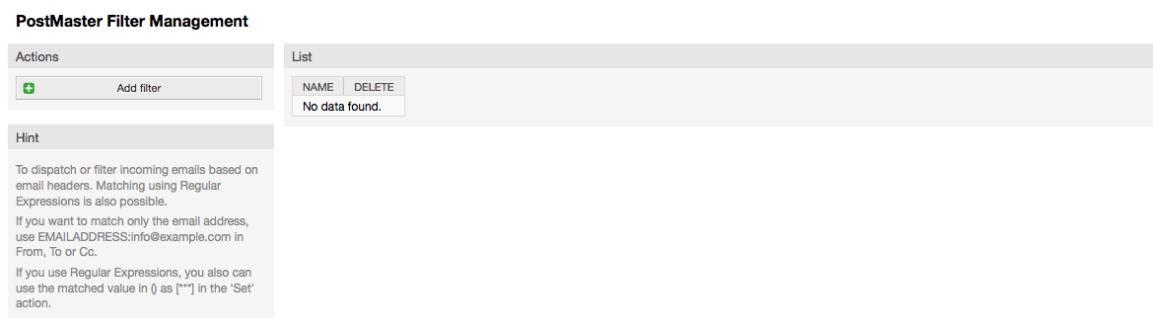


查看邮箱管理员邮件帐户以获得更多详细信息。

## 1.14. 过滤进入的邮件消息

OTRS可以过滤即将进入的邮件消息。例如，可以将某些邮件自动放入特定的队列，或给某些邮件设置特定状态或工单类型。这些过滤器适用于所有进入的邮件。你可以通过系统管理页面的“邮箱管理员过滤器”链接来管理这些过滤器（见下图）。

图 4.51. 邮箱管理员过滤器管理



一个过滤器由一个或多个需要按顺序逐一满足的条件组成，以对邮件执行定义的操作。过滤器条件可以使用邮件的头部或正文，例如搜索指定的邮件头条目如发件人地址或正文中的字符串。甚至可以使用正则表达式来扩展匹配模式。如果你的过滤器匹配了，你可以在图形界面中使用X-OTRS头来设置字段的值。这些值将在OTRS中用于创建工单或跟进消息。表4-5列出了不同的X-OTRS头信息和它们的含义。

### 注意

你还可以使用X-OTRS-FollowUp-\*头信息来设置跟进邮件的值。

表 4.5. 不同X-OTRS-headers的功能

名称	可能值	描述
X-OTRS-Priority:	1-非常低、2-低、3-正常、4-高、5-非常高	设置一个工单的优先级。
X-OTRS-Queue:	系统中一个队列的名称。	设置工单排序的队列。如果在X-OTRS头信息中设置了，将忽略其它所有试图分类工单到一个指定队列的过滤器。如果要使用子队列，需要按“父队列::子队列”格式指定。
X-OTRS-Lock:	锁定、解锁	设置一个工单的锁定状态。

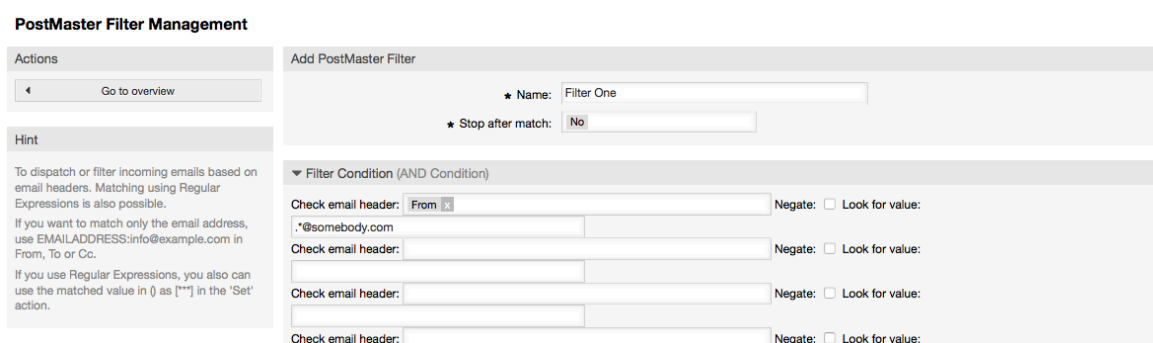


名称	可能值	描述
X-OTRS-Ignore:	Yes or True-是或真	如果这个X-OTRS值设置为“ <b>Yes</b> ”，即将进入的消息会完全忽略，永远不会传递到系统中。
X-OTRS-State:	新建、处理中、成功关闭、失败关闭等等	设置工单下一状态。
X-OTRS-State-PendingTime:	例如：2010-11-20 00:00:00	设置一个工单的挂起时间（你还应该通过X-OTRS-State设置一个挂起状态）。你可以指定绝对时间如“2010-11-20 00:00:00”或基于邮件到达时间的相对时间，相对时间的使用格式为“+ \$Number \$Unit”，\$Unit可以为‘s’（秒）、‘m’（分）、‘h’（小时）或‘d’（天），只能使用一个时间单位，例如有效设置为：“+50s”（挂起50秒）、“+30m”（30分钟）、“+12d”（12天）。注意：设置为“+1d 12h”是不行的，你可以使用“+36h”来替代。
X-OTRS-Type:	默认（取决于你的设置）	设置一个工单的类型（需要激活Ticket::Type）。
X-OTRS-Service:	（取决于你的设置）	设置一个工单的服务（需要激活Ticket::Service）。如果你要设置为一个子服务，应该按“父服务::子服务”格式指定。
X-OTRS-SLA:	（取决于你的设置）	设置一个工单的SLA（需要激活Ticket::Service支持）。
X-OTRS-CustomerUser:	客户联系人	设置工单的客户联系人。
X-OTRS-CustomerNo:	CustomerNo（客户联系人编号）	设置工单客户联系人的ID。
X-OTRS-SenderType:	agent, system, customer（服务人员、系统、客户联系人）	设置工单发件人的类型。
X-OTRS-ArticleType:	email-external（邮件-外部）、email-internal（邮件-内部）、email-notification-ext（邮件通知-外部）、email-notification-int（邮件通知-内部）、phone（电话）、fax（传真）、sms（短信）、webrequest（WEB请求）、note-internal（备注-内部）、note-external（备注-外部）、note-report（备注-报告）	设置即将进入工单的信件类型。
X-OTRS-DynamicField-<DynamicFieldName>:	取决于动态字段配置（文本：笔记本、日期：2010-11-20 00:00:00、整数：1）	为工单的<DynamicFieldName>动态字段保存一个额外的信息值。
X-OTRS-Loop:	True（真）	如果设置了这个X-OTRS头，就不会给消息的发件人发送自动答复（邮件循环保护）。

你应该给每个过滤器规则指定一个名称。过滤器条件可以在“过滤器条件”部分指定。从第一行、第二行等左边的下拉列表框中选择需要从中搜索的消息属性，然后在右边指定要过滤的值。在“设置邮件头”部分，你可以选择过滤器匹配后触发的操作。你可以为第一行、第二行等选择合适的X-OTRS-Header，并设置关联的值（见下图）。

过滤器规则按字母顺序评估，除非设置了“在匹配后停止”为“是”（在这种情况下匹配一条规则后就忽略其它的规则），否则会逐一评估所有的规则。

**图 4.52. 添加一个邮箱管理员过滤器**



**PostMaster Filter Management**

**Actions**

Go to overview

**Hint**

To dispatch or filter incoming emails based on email headers. Matching using Regular Expressions is also possible.  
 If you want to match only the email address, use EMAILADDRESS:info@example.com in From, To or Cc.  
 If you use Regular Expressions, you also can use the matched value in () as [""] in the 'Set' action.

**Add PostMaster Filter**

Name: Filter One

Stop after match: No

**Filter Condition (AND Condition)**

Check email header: From x Negate:  Look for value: \*

\*@somebody.com

Check email header: Negate:  Look for value:

Check email header: Negate:  Look for value:

Check email header: Negate:  Look for value:

Check email header: Negate:  Look for value:

### 例 4.1. 将垃圾邮件分类到一个指定队列

一个有用的过滤器规则可以让OTRS自动将标记为垃圾的邮件（通过垃圾邮件检测工具如SpamAssassin）转移到“Junk”队列。SpamAssassin会给检查过的所有邮件添加“X-Spam-Flag”头，当邮件被标记为垃圾邮件时，这个头的值被设置为“Yes”。所以这个过滤器条件就应该是：“X-Spam-Flag: Yes”。要使用这个条件创建一个过滤器，你可以将这个过滤器取名为诸如“spam-mails垃圾邮件”之类，在“过滤器条件”部分，从第一行列表框中选择“X-Spam-Flag”，并在右边的框中输入“Yes”作为搜索的值。现在就指定了过滤器的条件。要确保所有的垃圾邮件被放入“Junk”队列，在“设置邮件头”部分，为第一行选择“X-OTRS-Queue”，并指定“Junk”作为这个头的值。最后保存这个新的过滤器规则并用在系统新消息上。

还有一些附加的模块可以用来更有效地过滤即将进入的消息。这些模块在处理更大型、更复杂的系统时可能会很有用。

## 1.15. 使用GenericAgent（自动任务）执行自动任务

GenericAgent（自动任务）是一个自动执行任务的工具。自动任务的例子有关闭或转移工单、发送升级工单的通知等等。

系统管理页面点击“自动任务”链接（见下图），会用表格显示系统中所有的自动任务。这些任务可以被编辑、手动运行或完全删除。

**图 4.53. GenericAgent（自动任务）任务列表**



**Generic Agent**

**Actions**

Add job

**List**

NAME	LAST RUN	VALIDITY	DELETE	RUN NOW!
Job One		valid	Delete	Run this task
Job Three		valid	Delete	Run this task
Job Two		valid	Delete	Run this task

点击“添加任务”按钮来创建一个新的任务。你首先需要提供一个名称，然后可以指定这个任务如何执行：在固定的时间自动执行（像一个cron任务，这个模式会对工单过滤器匹配到的所有工单起作用）或基于工单事件执行（在一个匹配工单过滤器的特定工单被修改后立即执行）。注意：如果你从概览窗口手动运行基于事件的任务，则会对所有匹配工单过滤器的工单起作用。

图 4.54. 创建一个新的GenericAgent ( 自动任务 ) 任务

**Generic Agent**

Actions

[Go to overview](#)

Job Settings

Job name:

Validity:

Automatic execution (multiple tickets)

SCHEDULE MINUTES	SCHEDULE HOURS	SCHEDULE DAYS
<input type="text"/>	<input type="text"/>	<input type="text"/>

Currently this generic agent job will not run automatically.  
To enable automatic execution select at least one value from minutes, hours and days!

Event based execution (single ticket)

Event Triggers:

TYPE	EVENT	DELETE
<input type="text"/>	<input type="text"/>	<input type="text"/>

Additionally or alternatively to a periodic execution, you can define ticket events that will trigger this job. If a ticket event is fired, the ticket filter will be applied to check if the ticket matches. Only then the job is run on that ticket.

Add Event Trigger:

To add a new event select the event object and event name and click on the "+" button.

你可以给每个任务指定一个工单过滤器，例如只对某个队列里的工单起作用。一个工单上要运行的任务必须满足所有过滤条件。

最后，可以通过设置不同的工单字段如一个新的队列或状态来修改工单；可以附加一个备注到工单或运行一个定制模块；你还能从数据库中删除这个工单，这对清除系统中过期的或无效的数据有用。

## 警告

如果你使用了工单删除功能，所有影响的工单和它们的附件都将从数据库中删除并不可恢复！

编辑完成一个自动任务后，OTRS返回到任务概览窗口，你可以手动运行任何任务。如果运行一个任务，你首先会看到任务实际运行时影响到的所有工单。这个清单有助于验证任务是否按预期工作。在这个时候还没有对工单有任何修改，只有你在窗口中确认后任务才会执行。

## 1.16. 管理员消息

OTRS系统管理员可以给特定的用户或组发送消息。在系统管理页面的“管理员通知”链接打开一个发送管理员通知的窗口，可以选择要通知的服务人员和组（见下图）。

图 4.55. 管理员通知窗口

**Admin Notification**

Hint  
With this module, administrators can send messages to agents, group or role members.

Create Administrative Message

★ From: admin@mycompany.com

Send message to users: marc.bonsels x martin.gruner x and 1 more...

Send message to group members:

Group members need to have permission:  ro  rw

Send message to role members: Service Desk x

Also send to customers in groups:

★ Subject: A complete subject for notification

★ Body: **B I U S** [Rich Text Editor]  
Text example ...

Send

可以指定通知的发件人、主题和正文内容，还可以指定能收到这个消息的服务人员、组和角色。

## 1.17. 会话管理

系统管理页面的“会话管理”链接查看所有登录的用户及其会话详细信息（见下图）。

图 4.56. 会话管理

**Session Management**

SESSION	TYPE	USER	KILL
3sydOnqpwHQLUxa4083rVcTmBC1wPFB	Agent	Carlos Garcia	Kill this session
jzBgJlquUNb6950CaGHAN0tsME7eA5Fn	Customer	Han Solo	Kill this session

Actions

All sessions	2
Agent sessions	1
Customer sessions	1
Unique agents	1
Unique customers	1

Kill all sessions

在左侧会显示有关活动会话的统计信息，如登录的服务人员数、客户联系人数及活动的会话数。可以通过点击列表右侧的 终止该会话移除单个会话。还能够终止所有会话，当你要让系统离线（如系统维护等）时这个选项很有用。每个会话的详细信息也能看到（见下图）。

图 4.57. 会话详细信息

**Session Management**

Actions

Go to overview

Kill this session

Detail View for SessionID : 3sydOnqpwHQLUxa4083rVcTrmBC1wP18 - Carlos Garcia

KEY	VALUE
AdminDynamicFieldsOverviewPageShown	25
ChangeTime	2015-11-25 13:25:36
CreateTime	2015-11-25 13:25:36
ExternalChannels	[]
NotificationTransport	{&quot;Notification-3-Email&quot;;0,&quot;Notification-1-Email&quot;;0,&quot;Notification-4-Email&quot;;0,&quot;Notification-8-Email&quot;;0,&quot;Notification-13-Email&quot;;0,&quot;Notification-2-Email&quot;;0}
OutOfOfficeEndDay	12
OutOfOfficeEndMonth	12
OutOfOfficeEndYear	2015
OutOfOfficeStartDay	11
OutOfOfficeStartMonth	12
OutOfOfficeStartYear	2015
SessionID	3sydOnqpwHQLUxa4083rVcTrmBC1wP18
UserChallengeToken	4K1P2yUGJdCMXetMeSFIZom4zCqjRRfh
UserEmail	carlos.garcia@mycompany.com
UserFirstname	Carlos
UserFullname	Carlos Garcia
UserID	2
UsersGroupRo[admin]	Yes
UsersGroupRo[stats]	Yes
UsersGroupRo[users]	Yes
UsersGroupRo[admin]	Yes
UsersGroupRo[stats]	Yes

## 1.18. 系统维护

系统维护可用来计划OTRS系统的一个或多个维护周期。在计划周期内服务人员或客户联系人（除了“admin”组中的服务人员）都不能登录系统。当前登录的用户会收到一个关于系统维护的通知消息（在维护周期开始前和持续期间）。系统管理员可以选择终止登录的服务人员和客户联系人会话，所有这些准备工作使得我们可以在一个“安全”的环境中对系统进行修改（如系统更新）。

图 4.58. 有一些计划周期的系统维护概览窗口

**System Maintenance Management**

Actions

Schedule New System Maintenance

Hint

Schedule a system maintenance period for announcing the Agents and Customers the system is down for a time period.  
Some time before this system maintenance starts the users will receive a notification on each screen announcing about this fact.

List

START DATE	STOP DATE	COMMENT	VALIDITY	DELETE
2015-12-03 19:57:00	2015-12-03 22:57:00	A comment about this maintenance period	valid	

开始日期和停止日期是必填字段，唯一的组合规则是开始日期不能在停止日期之后。

图 4.59. 系统维护编辑窗口

**Edit System Maintenance**

Actions

Go to overview

▼ Edit System Maintenance information

Start date: 12 / 03 / 2015 - 19 : 57

Stop date: 12 / 03 / 2015 - 22 : 57

★ Comment:

Login message:

Show login message:

Notify message:

★ Validity:

or

---

▼ Manage Sessions

All Sessions 2  
Unique agents 1  
Unique customers 1

**Agent Sessions**

SESSION	TYPE	USER	KILL
3sydOnqpwHQLUxa4083rVcTrmBC1wP18	Agent	Carlos Garcia	Kill this session

**Customer Sessions**

SESSION	TYPE	USER	KILL
jzBgJlquUNb6950CaGHAN0tsME7eA5Fn	Customer	Han Solo	Kill this session

在定义了新的维护周期后会显示当前活动会话的概览和详细信息，系统管理员在这里能够根据需要逐个终止会话或终止所有会话（除了当前会话）。

## 1.19. 系统日志

系统管理页面的“系统日志”链接显示系统中的日志条目，按时间倒序将最近的显示在最上面（见下图）。

图 4.60. 系统日志

**System Log**

Hint

Here you will find log information about your system.

Recent Log Entries

TIME	PRIORITY	FACILITY	MESSAGE
Tue Dec 2 02:00:12 2014	notice	OTRS-otrs.GenericAgent.pl-2664	Use module (Kernel::System::GenericAgent::TriggerAdvancedEscalationStartEvents) for Ticket (20141112266400012/3).
Tue Dec 2 02:00:11 2014	notice	OTRS-otrs.GenericAgent.pl-2664	Added scheduler job 'EscalationHistory' by escalation event 'EscalationBreachd_2' for ticket '2'!
Tue Dec 2 02:00:11 2014	notice	OTRS-CGI-3051	CustomerUser: 'bruce.banner' changed password successfully!
Tue Dec 2 02:00:11 2014	notice	OTRS-otrs.GenericAgent.pl-1092	Use module (Kernel::System::GenericAgent::TriggerEscalationStartEvents) for Ticket (109200664/990).

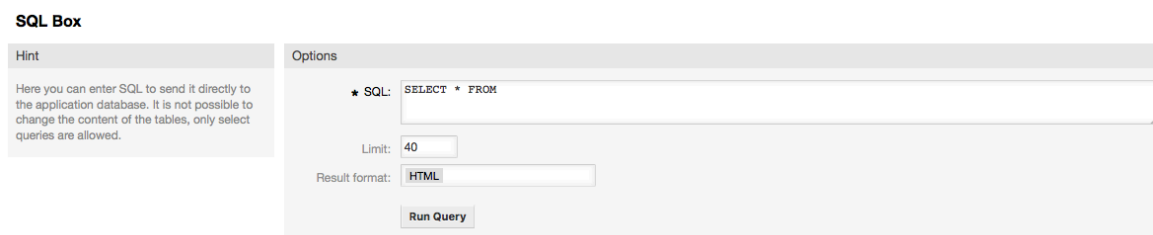
日志中每一行都包含一个时间戳、日志优先级、系统组件和日志内容。

### 注意

WEB界面中的系统日志只在Linux/Unix系统上可用。

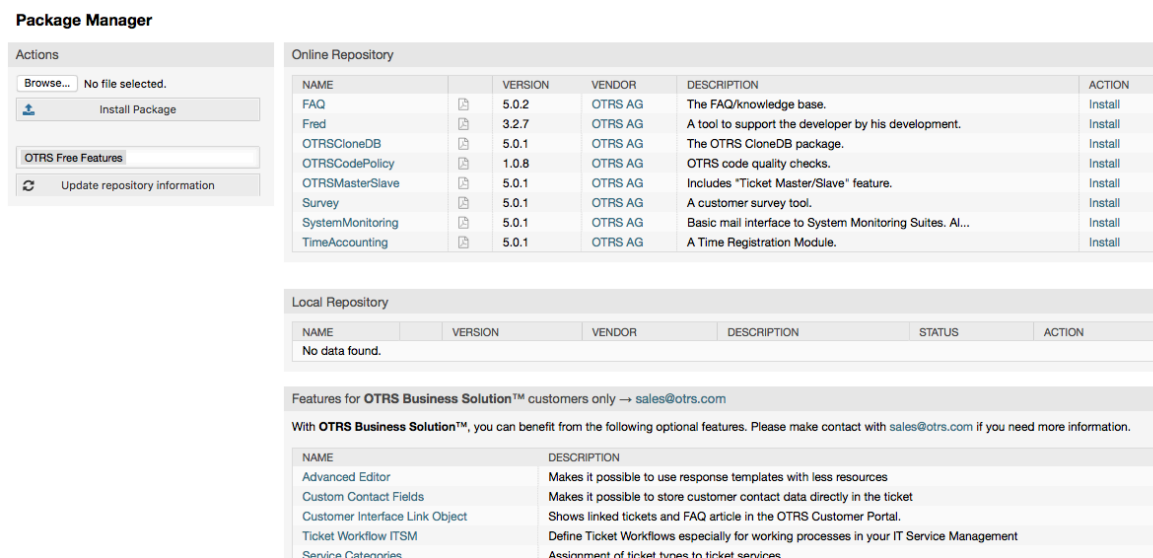
## 1.20. 通过SQL查询窗口执行SQL查询

系统管理页面的“SQL查询窗口”让你可以查询OTRS数据库中表的数据（见下图）。不能修改表的内容，只允许执行“select”查询语句。

**图 4.61. SQL查询窗口**


## 1.21. 软件包管理器

使用系统管理页面的“软件包管理器”链接，可以安装和管理软件包，以扩展OTRS的功能（见下图）。查看附加的应用程序部分可以找到OTRS可用的扩展应用程序。

**图 4.62. 软件包管理器**


软件包管理器显示服务器上当前已经安装了的OTRS插件软件包以及它们的版本号。

你可以从远程主机安装软件包，选择在线软件仓库，点击更新软件仓库信息，则会显示可用的软件。窗口右侧显示可用的软件包。要安装一个软件包，点击安装即可。安装完成后，这个软件包就会在本地软件仓库部分显示。

要更新已安装的软件包，在线软件仓库中会在操作列显示有高版本可用的软件包为更新。只需点击“更新”即可将新版本的软件包安装到系统中。

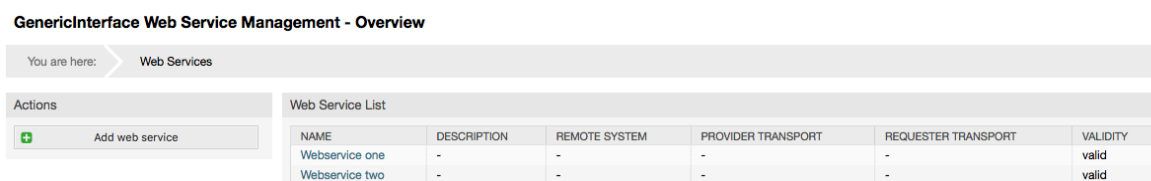
在某些情况下，比如你的OTRS系统无法连接到Internet，你还可以安装下载到本地磁盘的软件包。点击左侧的浏览按钮，并选择磁盘上软件包的.opm文件，点击打开，再点击安装软件包按钮。在安装完成后，这个软件包就会在本地软件仓库部分显示。同样可以用这个方法更新一个已经安装的软件包。

在特殊情况下，你可能想配置软件包管理器，例如：使用代理或使用本地软件仓库，可以看看系统配置Framework:Core::Package中可用的选项。

## 1.22. Web服务

系统管理页面的WEB服务链接引出创建和维护WEB服务（用于OTRS通用接口）的图形界面（见下图）。

图 4.63. WEB服务的图形界面

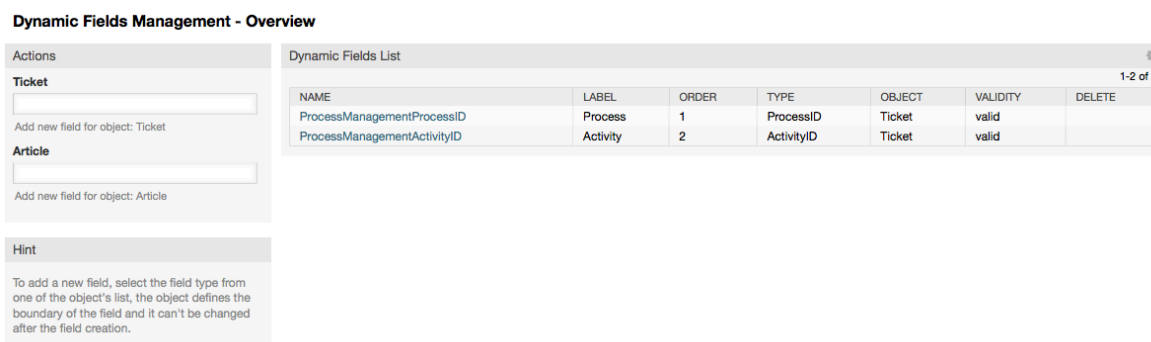


WEB服务配置的图形界面在“WEB服务图形界面”部分有详细描述。

## 1.23. 动态字段

动态字段是你设置和管理工单和信件的定制字段的地方（见下图）。

图 4.64. 有一些动态字段的动态字段概览窗口



动态字段配置的详细描述在“动态字段”部分。

每个动态字段类型都有它自己的配置设置，因而有它自己的配置窗口。

### 注意

在OTRS框架中，动态字段默认只能链接到工单和信件，但它们实际上同样可以扩展到其它的对象。

## 2. 系统配置

### 2.1. OTRS 配置文件

OTRS的所有配置文件都保存在目录Kernel及其子目录中。除了Kernel/Config.pm外无需手动修改任何其它文件，因为其它的文件在系统升级时会被覆盖。只需从其它文件将配置参数复制到Kernel/Config.pm并按你的需求修改即可，这个文件在升级过程中永远不会被改动，所以你的手动配置是安全的。

在目录Kernel/Config/Files中有一些其它文件，在访问OTRS登录页面时解析。如果安装了附加的应用模块如FAQ知识库或文件管理器，这些模块的配置文件也放在这个目录中。

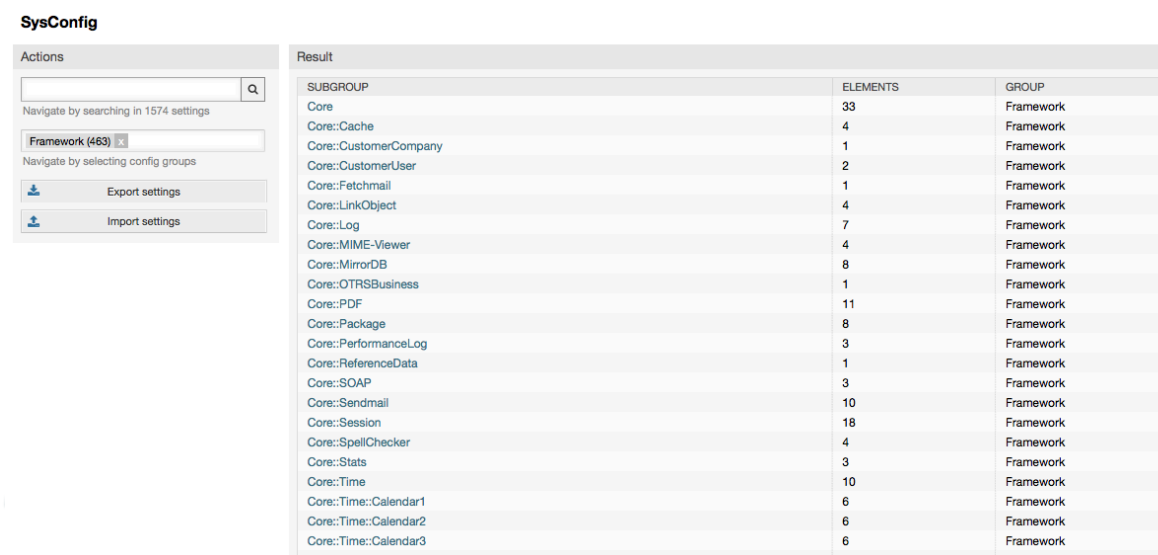
如果访问OTRS的WEB界面，目录Kernel/Config/Files下的所有.xml文件按字母顺序被解析，并载入中央框架和附加应用模块的设置。然后评估Kernel/Config/Files/ZZZAAuto.pm、Kernel/Config/Files/ZZZAuto.pm和Kernel/Config/Files/ZZZProcessManagement.pm（如果有）中的设置。这些文件用于图形界面的系统配置缓存，永远不要手动修改。最后解析Kernel/Config.pm，它包含了你的独特设置和手动修改的配置参数。



## 2.2. 通过WEB界面配置系统

从OTRS 2.0开始，近乎所有的中央框架或附加应用模块的配置参数都可以通过系统配置图形界面轻松修改。以OTRS系统管理员登录，并在系统管理页面点击系统配置链接以执行新的配置工具（见下图）。

图 4.65. 系统配置的图形界面



OTRS当前已经有超过600个配置参数，可以通过不同的方法来快速访问一个特定的配置参数。使用全文搜索，可以使用一个或多个关键词扫描所有的配置参数，全文搜索不光搜索配置参数名称，还会搜索参数的描述文字。这样即使不知道名称也能轻松查找一个配置参数。

此外，所有的配置参数存储在主组和子组中。主组代表配置参数所属的应用模块如“框架”就是OTRS核心框架、“Ticket”就是工单系统、“FAQ”就是FAQ知识库系统等等。从主组列表中选择了应用模块后，点击右侧的子组名即可访问子组中的所有配置参数。

每个配置参数都可以通过一个复选框来打开或关闭。如果该参数被关闭，系统就会忽略这个参数或使用一个默认值。还可以通过重置链接将修改过的配置参数改回系统默认值。‘更新’按钮将所有更改内容提交到系统配置参数。

如果你要保存所有你修改过的系统配置，比如要快速设置一个新安装的系统，你可以使用“导入配置”按钮，它会创建一个.pm文件。要恢复你自己的设置，只需点击“导入配置”并选择之前创建的.pm文件即可。

### 注意

出于安全原因，数据库连接的配置参数不能在系统配置图形界面修改。它们必须在文件Kernel/Config.pm中手动设置。

## 3. 备份系统

本章描述了ORTS数据的备份和恢复。

### 3.1. 备份

有两种类型的数据需要备份：应用程序文件（如/opt/otrs目录下的文件）和存储在数据库中的数据。

为了简化备份，在每个OTRS安装中已经包括了脚本scripts/backup.pl。运行它可以备份所有重要的数据（见下面的脚本）。

```
linux:/opt/otrs# cd scripts/
linux:/opt/otrs/scripts# ./backup.pl --help
backup.pl - backup script
Copyright (C) 2001-2020 OTRS AG, https://otrs.com/
usage: backup.pl -d /data_backup_dir/ [-c gzip|bzip2] [-r 30] [-t fullbackup|nofullbackup|
dbonly]
linux:/opt/otrs/scripts#
```

脚本：获取关于OTRS备份机制的帮助。

执行下面脚本中指定的命令来创建一个备份：

```
linux:/opt/otrs/scripts# ./backup.pl -d /backup/
Backup /backup//2010-09-07_14-28/Config.tar.gz ... done
Backup /backup//2010-09-07_14-28/Application.tar.gz ... done
Dump MySQL rdbms ... done
Compress SQL-file... done
linux:/opt/otrs/scripts#
```

脚本：创建一个备份。

所有数据都保存在目录/backup/2010-09-07\_14-28/下（见下面的脚本）。另外，这些数据被保存到一个.tar.gz文件。

```
linux:/opt/otrs/scripts# ls /backup/2010-09-07_14-28/
Application.tar.gz Config.tar.gz DatabaseBackup.sql.gz
linux:/opt/otrs/scripts#
```

脚本：检查备份文件。

## 3.2. 恢复

要恢复一个备份，保存的应用程序数据必须被写回到安装目录，如/opt/otrs。还必须要恢复数据库。

每个OTRS安装都自带了一个脚本文件scripts/restore.pl（见下面的脚本），它简化了恢复过程，支持MySQL和PostgreSQL。

```
linux:/opt/otrs/scripts# ./restore.pl --help
restore.pl - restore script
Copyright (C) 2001-2020 OTRS AG, https://otrs.com/
usage: restore.pl -b /data_backup/<TIME>/ -d /opt/otrs/
linux:/opt/otrs/scripts#
```

脚本：获取关于OTRS恢复机制的帮助。

例如：数据保存在目录/backup/2010-09-07\_14-28/下，可以使用下面脚本中指定的命令来恢复，假定OTRS安装在/opt/otrs目录。

```
linux:/opt/otrs/scripts# ./restore.pl -b /backup/2010-09-07_14-28 -d /opt/otrs/
Restore /backup/2010-09-07_14-28//Config.tar.gz ...
Restore /backup/2010-09-07_14-28//Application.tar.gz ...
create MySQL
decompresses SQL-file ...
cat SQL-file into MySQL database
compress SQL-file...
linux:/opt/otrs/scripts#
```

脚本：恢复OTRS数据。

## 4. 邮件设置

### 4.1. 发送/接收邮件

#### 4.1.1. 发送邮件

##### 4.1.1.1. 通过Sendmail (默认)

OTRS可以通过Sendmail、Postfix、Qmail 或 Exim发送邮件。默认配置是使用Sendmail，开箱即用。

你可以通过图形配置界面配置sendmail设置 ( Framework::Core::Sendmail )

##### 4.1.1.2. 通过SMTP或SMTPS服务器

OTRS可以通过SMTP ( Simple Mail Transfer Protocol /RFC 821 ) 或Secure SMTP发送邮件。

可以通过系统配置Framework::Core::Sendmail设置SMTP服务器参数。如果你没有看到SMTPS选项，说明缺失需要的Perl模块。在这种情况下，请参考为OTRS安装额外的Perl模块的说明。

#### 4.1.2. 接收邮件

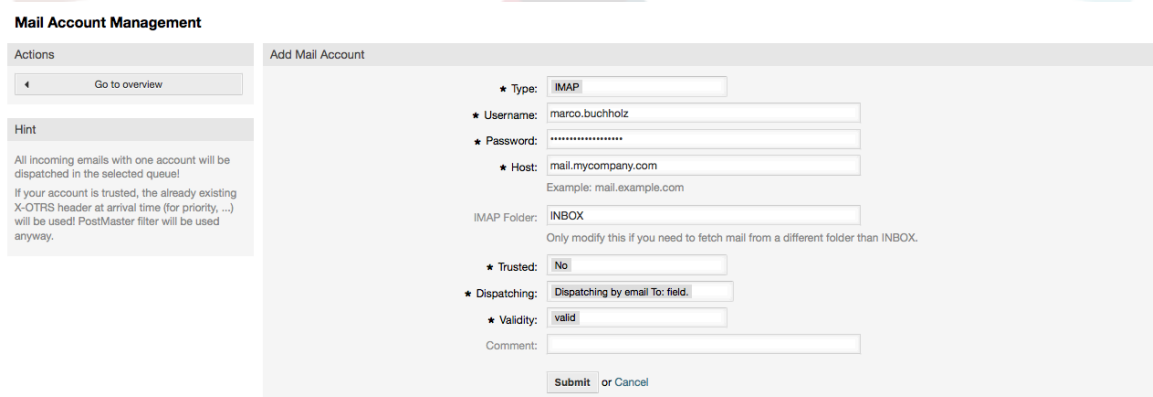
##### 4.1.2.1. 通过OTRS图形界面配置邮件帐户

OTRS能够从POP3、POP3S、IMAP和IMAPS的邮件帐户接收邮件。

通过系统管理页面的“邮箱管理员邮件帐户”链接配置你的邮件帐户。

如果要创建一个新的邮件帐户（见下图），必须指定邮件服务器名称、登录用户名和密码。另外，你要选择邮件服务器的类型（可以是POP3、POP3S、IMAP或IMAPS）。如果你没有看到需要的类型，说明系统中缺失需要的Perl模块。在这种情况下，请参考为OTRS安装额外的Perl模块的说明。

图 4.66. 新增一个邮件账号



**Mail Account Management**

Actions: [Go to overview](#)

Hint: All incoming emails with one account will be dispatched in the selected queue! If your account is trusted, the already existing X-OTRS header at arrival time (for priority, ...) will be used! PostMaster filter will be used anyway.

**Add Mail Account**

★ Type:

★ Username:

★ Password:

★ Host:   
Example: mail.example.com

IMAP Folder:   
Only modify this if you need to fetch mail from a different folder than INBOX.

★ Trusted:

★ Dispatching:

★ Validity:

Comment:

or

如果“是否信任”选择为“是”，OTRS将对邮件信头中现有的X-OTRS标记进行过滤评估并执行相应的处理操作。因为在工单系统中可以根据X-OTRS标记执行一些操作，你应该将“是否信任”选项设置为“是”仅已知的发件人。OTRS中过滤器模块使用X-OTRS-Headers，X-OTRS头标识在这个表中有详细解释。而创建的邮箱管理员过滤器总是要执行的，与“是否信任”选项的设置无关。

如果进入消息需要按队列或按收件人内容分类，就可以控制它们的分配。对于Dispatching字段，如果选择了“按指定的队列”，所有进入的消息会按指定的队列分类，这种情况下邮件的收件人地址就会被忽略。如果选择了“按邮件收件人字段”，系统检查一个队列是否链接到进入邮件的收件人字段。可以在系统管理页面的邮件地址管理中将一个地址链接到一个队列。如果收件人字段中的地址已经链接

到一个队列，新消息就会分类到这个队列中。如果收件人字段中的地址没有链接到任何队列，则消息就会流入系统的“RAW（原始）”队列（这是默认安装后PostmasterDefaultQueue参数设置）。

邮件帐户的所有数据都保存在OTRS数据库中。bin/otrs.Console.pl  
Maint::PostMaster::MailAccountFetch命令使用数据库中的设置来收取邮件。你可以手动执行它来检查你的邮件设置是否能正常工作。

在一个普通安装的OTRS中，由OTRS守护进程每10分钟收取一次邮件。

## 注意

收取邮件时OTRS会从POP或IMAP服务器上删除邮件。没有选项来让服务器仍保留一份副本。如果你想在服务器保留一份副本，你应当在邮件服务器上创建转发规则，请查阅你的邮件服务器文档以获取详细信息。

### 4.1.2.2. 通过命令程序和Procmail (otrs.Console.pl Maint::PostMaster::Read)

如果你不能使用邮件帐户来收取邮件到OTRS，命令程序 bin/otrs.Console.pl  
Maint::PostMaster::Read可能是一个解决这个问题的办法。它通过STDIN（标准输入）取走邮件并通过管道直接输入到OTRS中，这意味着如果MDA（邮件投递代理，如procmail）执行这个程序就能在OTRS中使用邮件。

为了测试不用MDA时bin/otrs.Console.pl Maint::PostMaster::Read的效果，执行下面的脚本。

```
linux:/opt/otrs# cd bin
linux:/opt/otrs/bin# cat ../doc/sample_mails/test-email-1.box | ./otrs.Console.pl
Maint::PostMaster::Read
linux:/opt/otrs/bin#
```

脚本：测试没有MDA的邮箱管理员。

如果这封邮件显示在队列视图中，则你的设置能工作了。

### 例 4.2. 使用otrs.Console.pl通过Procmail路由

为了在一个特定队列中路由邮件，按下面的例子使用otrs.Console.pl。

```
| $SYS_HOME/bin/otrs.Console.pl Maint::PostMaster::Read --target-queue=QUEUENAME
```

当分类到一个子队列时，你必须使用双冒号::来分隔父队列和子队列。

```
| $SYS_HOME/bin/otrs.Console.pl Maint::PostMaster::Read --target-queue=QUEUENAME::SUBQUEUE
```

Procmail是Linux平台非常通用的一个邮件过滤器。它已大部分系统中安装，如果没有安装，看看[procmail主页](#)。

要配置procmail用于OTRS（基于procmail配置的MTA如sendmail、postfix、exim或qmail），使用~otrs/.procmailrc.dist文件，将它复制为.procmailrc，并添加下面脚本中的行到文件中。

```
SYS_HOME=$HOME
PATH=/bin:/usr/bin:/usr/local/bin
# --
# Pipe all email into the PostMaster process.
# --
:0 :
| $SYS_HOME/bin/otrs.Console.pl Maint::PostMaster::Read
```

脚本：配置*procmail*用于OTRS。

所有发送给本地OTRS用户的邮件将通过管道传递给bin/otrs.Console.pl  
Maint::PostMaster::Read，然后显示在你的队列视图中。

#### 4.1.2.3. 通过POP3或IMAP收取邮件并将fetchmail用于otrs.Console.pl Maint::PostMaster::Read

为了通过POP3或IMAP邮件中收取邮件到OTRS主机/本地OTRS帐户和procmail，使用*fetchmail*。

### 注意

OTRS主机上需要一个可以工作的SMTP配置。

你可以使用OTRS的home目录中的文件*.fetchmailrc.dist*，将它得到为*.fetchmailrc*。修改为你需要的内容（见下面的例子）。

#### 例 4.3. .fetchmailrc

```
#poll (mailserver) protocol POP3 user (user) password (password) is (localuser)
poll mail.example.com protocol POP3 user joe password mama is otrs
```

不要忘记设置文件*.fetchmailrc*的权限为710（**chmod 710 .fetchmailrc**）！

使用上面例子中的文件*.fetchmailrc*，如果执行了命令**fetchmail -a**，会转发所有的邮件到本地OTRS帐户。如果你要定期收取邮件，用这个命令设置一个cron任务即可。

#### 4.1.2.4. 通过OTRS 邮箱管理员模块过滤/分派（用于更复杂的分派）

如果你使用bin/otrs.Console.pl Maint::PostMaster::Read或bin/otrs.Console.pl  
Maint::PostMaster::MailAccountFetch方法，你可以使用邮箱管理员过滤器模块插入或修改X-OTRS头信息。通过X-OTRS头信息，工单系统能够对进入的邮件执行一些操作，例如：分类到一个特定的队列、修改优先级或修改客户联系人ID。关于X-OTRS头的更多信息请参阅系统管理页面的添加邮件帐户部分。

有一些默认的过滤器模块：

### 注意

任务名称（如\$Self->{'PostMaster::PreFilterModule'}->{'JobName'}）需要是唯一的。

Kernel::System::PostMaster::Filter::Match是一个匹配一些邮件头（如发件人、收件人、主题等）的默认模块。如果匹配了一个过滤器规则，它可以设置新的邮件头（例如X-OTRS-Ignore: yes或X-OTRS-Queue: spam）。下面例子中的任务可以添加到文件Kernel/Config.pm。

#### 例 4.4. 过滤器模块Kernel::System::PostMaster::Filter::Match的任务示例

```
# Job Name: 1-Match
# (block/ignore all spam email with From: noreply@)
$Self->{'PostMaster::PreFilterModule'}->{'1-Match'} = {
  Module => 'Kernel::System::PostMaster::Filter::Match',
  Match => {
    From => 'noreply@',
  },
  Set => {
    'X-OTRS-Ignore' => 'yes',
  },
}
```

```

};
# Job Name: 2-Match
# (sort emails with From: sales@example.com and Subject: **ORDER**
# into queue 'Order')
$self->{'PostMaster::PreFilterModule'}->{'2-Match'} = {
  Module => 'Kernel::System::PostMaster::Filter::Match',
  Match => {
    To => 'sales@example.com',
    Subject => '**ORDER**',
  },
  Set => {
    'X-OTRS-Queue' => 'Order',
  },
};

```

Kernel::System::PostMaster::Filter::CMD是一个将邮件通过管道传递到外部命令的默认模块。输出为STDOUT（标准输出），如果结果为真，则设置新的邮件头（如X-OTRS-Ignore: yes或X-OTRS-Queue: spam）。下面例子中的内容可以添加到文件Kernel/Config.pm。

#### 例 4.5. 过滤器模块Kernel::System::PostMaster::Filter::CMD的任务示例

```

# Job Name: 5-SpamAssassin
# (SpamAssassin example setup, ignore spam emails)
$self->{'PostMaster::PreFilterModule'}->{'5-SpamAssassin'} = {
  Module => 'Kernel::System::PostMaster::Filter::CMD',
  CMD => '/usr/bin/spamassassin | grep -i "X-Spam-Status: yes"',
  Set => {
    'X-OTRS-Ignore' => 'yes',
  },
};

```

Kernel::System::PostMaster::Filter::ExternalTicketNumberRecognition是一个在邮件主题、正文或这两者中使用正则表达式解析外部标识的默认模块。它在一个定义的动态字段中存储这个值。当一封邮件进入时，OTRS会首先搜索一个外部标识，当它找到一个外部标识，查询OTRS中预定义的动态字段，如果它找到了一个存在的工单，它就会更新这个工单，否则它会创建一个新的工单并将这个外部引用号放入单独的字段。

OTRS系统配置中已经提供了4种不同的设置来设置不同的外部工单编号。如果需要更多的设置，就得手动添加。下面的例子可以用在文件Kernel/Config.pm中来扩展系统配置的设置。

#### 例 4.6. 过滤器模块Kernel::System::PostMaster::Filter::ExternalTicketNumberRecognition的任务示例

```

# Job Name: ExternalTicketNumberRecognition
# External Ticket Number Reconition, check for Incident-<number> in incoming mails
subject and
# body from the addeesses <sender>@externalticket.com, if number is found it will be
stored in
# the dynamic field 'ExternalNumber' (that need to be setup in the Admin Panel).
$self->{'PostMaster::PreFilterModule'}->{'000-ExternalTicketNumberRecognition'} = {
  'FromAddressRegExp' => '\\s*@externalticket.com',
  'NumberRegExp' => 'Incident-(\\d.*)',
  'SearchInSubject' => '1',
  'SearchInBody' => '1',
  'TicketStateTypes' => 'new;open'
  'DynamicFieldName' => 'ExternalNumber',
  'Module' =>
'Kernel::System::PostMaster::Filter::ExternalTicketNumberRecognition',
  'Name' => 'Test External Ticket Number',
  'SenderType' => 'system',
};

```

配置选项

- FromAddressRegExp (发件人地址正则表达式)

这是一个可选的设置。只有匹配这个“收件人”地址的邮件才会考虑使用这个过滤器。你可以调整这个设置为外部系统用来发出邮件的发件人地址。在这种情况下这个地址可以不同，你可以设置这个选项为空，这样OTRS不会检查发件人地址。

- NumberRegExp (数字正则表达式)

这是一个强制设置。这个设置包含了OTRS用来从主题或工单正文中提取工单编号的正则表达式。默认的正则表达式会匹配出现的如‘Incident-12354’，并把括号之间的部分放入动态字段，此处是‘12354’。

- SearchInSubject (搜索主题)

如果这个设置为‘1’，会在邮件主题中搜索工单编号。

- SearchInBody (搜索正文)

如果这个设置为‘1’，会在邮件正文中搜索工单编号。

- TicketStateTypes (工单状态类型)

这是一个可选的设置。如果指定了,它将只搜索OTRS处理指定状态类型的外部工单。状态类型用分号分隔。

- DynamicField (动态字段)

这是一个必填设置。它定义了用来存储外部编号的动态字段(字段名必须存在于系统并处于有效状态)。

- SenderType (发件人类型)

这个参数定义OTRS中创建的用于信件的发件人类型。

Kernel::System::PostMaster::Filter::Decrypt是用来解密传入的(S/MIME或PGP)加密过的电子邮件消息的默认模块，它将未加密的消息正文放置在邮件头X-OTRS-BodyDecrypted中供以后处理。此外它还可以更新电子邮件正文为未加密版本。

要对电子邮件进行解密，系统需要正确地配置S/MIME和/或PGP，并且有解密信息所需的私钥。

本模块默认是禁用的，可以直接在系统管理面板的系统配置中进行配置。

#### 配置选项

- StoreDecryptedBody

设置本选项为“1”来更新邮件正文为未加密的版本(如果成功解密)。注意使用此选项后电子邮件将存储为未加密格式且无法还原此操作。

当然你也可以开发你自己的邮箱管理员过滤器模块。

#### 4.1.2.5. 邮件过滤器故障排除

这部分显示了在排查邮箱管理员过滤器的故障时一些共同问题及需要考虑的事情。

- 过滤器按它们名称的字母顺序按序匹配。如果某个条件匹配了两次，则最后匹配的过滤器生效。
- “匹配后停止”可以防止二次匹配。
- 确保正则表达式是有效的。
- 可以设置头信息来控制OTRS，但不会写入邮件中。

- 当匹配一个发件人、抄送、收件人时，使用EMAILADDRESS: <your@address>
- 邮箱必须是可信的。
- 匹配条件是“与”条件。
- 邮箱管理员过滤器不能匹配工单属性。

## 4.2. 用PGP加密邮件

OTRS能够使用PGP来签名或加密外发消息。此外，进入的加密消息能被解密。加密和解密都通过GPL工具GnuPG完成。要为OTRS设置GnuPG，需要执行以下步骤：

1. 通过操作系统的软件包管理器安装GnuPG。
2. 配置GnuPG用于OTRS。必须创建GnuPG需要的目录和私有密钥。下面脚本中显示的命令必须以‘otrs’用户执行。

```
linux:~# su otrs
linux:/root$ cd
linux:~$ pwd
/opt/otrs
linux:~$ gpg --gen-key
gpg (GnuPG) 1.4.2; Copyright (C) 2005 Free Software Foundation, Inc.
This program comes with ABSOLUTELY NO WARRANTY.
This is free software, and you are welcome to redistribute it
under certain conditions. See the file COPYING for details.

gpg: directory `/opt/otrs/.gnupg' created
gpg: new configuration file `/opt/otrs/.gnupg/gpg.conf' created
gpg: WARNING: options in `/opt/otrs/.gnupg/gpg.conf' are not yet active during t
his run
gpg: keyring `/opt/otrs/.gnupg/secring.gpg' created
gpg: keyring `/opt/otrs/.gnupg/pubring.gpg' created
Please select what kind of key you want:
  (1) DSA and Elgamal (default)
  (2) DSA (sign only)
  (5) RSA (sign only)
Your selection? 1
DSA keypair will have 1024 bits.
ELG-E keys may be between 1024 and 4096 bits long.
What keysize do you want? (2048)
Requested keysize is 2048 bits
Please specify how long the key should be valid.
  0 = key does not expire
  <n> = key expires in n days
  <n>w = key expires in n weeks
  <n>m = key expires in n months
  <n>y = key expires in n years
Key is valid for? (0)
Key does not expire at all
Is this correct? (y/N) y

You need a user ID to identify your key; the software constructs the user ID
from the Real Name, Comment and Email Address in this form:
  "Heinrich Heine (Der Dichter) <heinrichh@duesseldorf.de>"

Real name: Ticket System
Email address: support@example.com
Comment: Private PGP Key for the ticket system with address support@example.com
You selected this USER-ID:
"Ticket System (Private PGP Key for the ticket system with address support@examp
le.com) <support@example.com>"

Change (N)ame, (C)omment, (E)mail or (O)kay/(Q)uit? 0
You need a Passphrase to protect your secret key.

Passphrase: secret
```



```

Repeat passphrase: secret

We need to generate a lot of random bytes. It is a good idea to perform
some other action (type on the keyboard, move the mouse, utilize the
disks) during the prime generation; this gives the random number
generator a better chance to gain enough entropy.
+++++++.....+++++.++++.....+++++.....+++++
+++++++.....+++++.++++.++++.+++++.....+++++>+++++>+++++
.....>+++++<+++++.....+++++

Not enough random bytes available. Please do some other work to give
the OS a chance to collect more entropy! (Need 280 more bytes)

+++++++.....+++++.++++.....+++++.....+++++.....+++++.....+++++
+++++++.....+++++.++++.++++.+++++.....+++++.....+++++.....+++++
..+++++>.....>+++++.....>+++++<+++++.....>+++++<+++++.....
.....>+++++<+++++.....
.....+++++^^^
gpg: /opt/otrs/.gnupg/trustdb.gpg: trustdb created
gpg: key 7245A970 marked as ultimately trusted
public and secret key created and signed.

gpg: checking the trustdb
gpg: 3 marginal(s) needed, 1 complete(s) needed, PGP trust model
gpg: depth: 0 valid: 1 signed: 0 trust: 0-, 0q, 0n, 0m, 0f, 1u
pub 1024D/7245A970 2006-02-03
    Key fingerprint = 2ED5 BC36 D2B6 B055 7EE1 5833 1D7B F967 7245 A970
    uid          Ticket System (Private gpg key for ticket system with address support@example.com) <support@example.com>
sub 2048g/52B97069 2006-02-03

linux:~$

```

脚本：配置GnuPG。

正如脚本中显示那样，大部分需要的参数使用默认设置即可。密钥所有者仅需正确输入这个密钥合适的密码就行。

### 3. 在 PGP设置窗口，PGP应该已激活（第一个选项）。另外还应该设置并检查gpg程序路径。

下一个配置设置（PGP::Options）可能也需要修改。通过这个配置设置，可以指定‘otrs’用户每次gpg执行时的参数。尤其是‘otrs’用户的GnuPG配置文件目录很重要。在示例中使用的是/opt/otrs/.gnupg目录，这个目录要在PGP配置过程之前创建。

通过下一个配置选项（PGP::Key::Password）可以指定密钥对的ID和它们私有密钥的密码。因为外部的通信伙伴用你的公钥加密写给工单系统的消息，OTRS就能使用在这里指定的ID/密码来解密这些消息。

如何取得你自己的私钥ID？你自己的私钥ID已经在生成密钥过程中显示过了（见前面的步骤1）。可以通过‘otrs’用户执行下面脚本中的命令来取得这个ID：

```

linux:~# su otrs
linux:/root$ cd
linux:~$ pwd
/opt/otrs
linux:~$ gpg --list-keys
/opt/otrs/.gnupg/pubring.gpg
-----
pub 1024D/7245A970 2006-02-03
uid          Ticket System (Private gpg key for ticket system with address support@example.com) <support@example.com>
sub 2048g/52B97069 2006-02-03

linux:~$

```

脚本：获取你自己的私钥ID。

私钥ID可以在以“sub”开头的行中找到，它是8个字符长度的十六进制字符串，在这个示例中它是“52B97069”。在OTRS中需要指定的密码就是在生成密钥时输入的密码。

在输入这些数据后，点击“更新”按钮来保存设置。OTRS现在就准备好接收和解密加密过的消息了。

4. 最后，导入一个客户联系人的公钥。这确保可以给这个客户联系人发送加密消息。有两个方法来导入一个客户联系人的公钥。

第一个方法是在客户管理界面指定一个客户联系人的公钥。

第二个方法是从系统管理页面通过PGP设置指定公钥。在窗口右侧会显示所有已导入的客户联系人公钥。在激活了PGP且配置了OTRS的PGP后，你自己的公钥也应该在这里列出。在PGP设置窗口的左侧可以搜索密钥。此外，可以通过一个文件上传一个新的公钥。

需要导入到OTRS中公钥文件必须是GnuPG兼容的密钥文件。大部分情况下，存储在一个文件中的密钥是“ASCII码封装的密钥”，OTRS能够处理这个格式。

### 4.3. 用S/MIME加密邮件

乍一看，使用S/MIME加密似乎要比PGP复杂一点。首先，你必须为OTRS系统建立一个认证机构（CA）。随后的步骤与PGP非常相似：配置OTRS、安装你自己的证书、导入其它需要的公共证书等等。

S/MIME配置主要是在OTRS的WEB界面之外进行，需要用‘otrs’用户在shell中执行。Linux下MIME配置基于SSL（OpenSSL）。所以首先要检查你的系统中是否安装了OpenSSL软件包。OpenSSL中包括了一个叫CA.pl的脚本文件，可以用它来执行创建证书的大部分重要步骤。为了简化过程，在文件系统查找CA.pl的路径，并将这个路径临时加入当前shell的PATH变量（见下面的脚本）。

```
otrs@linux:~> rpm -ql openssl | grep CA
/usr/share/ssl/misc/CA.pl
otrs@linux:~> export PATH=$PATH:/usr/share/ssl/misc
otrs@linux:~> which CA.pl
/usr/share/ssl/misc/CA.pl
otrs@linux:~> mkdir tmp; cd tmp
otrs@linux:~/tmp>
```

脚本：配置S/MIME。

上面的脚本显示创建了一个新的临时目录~/tmp，这是生成证书的位置。

要创建一个证书，在命令行中执行下面的操作（我们假定OTRS系统管理员已经为测试和学习目的创建了一个SSL证书。如果你已经有一个验证过的用来加密的SSL证书，使用它就就行了，可以跳过这些步骤）：

1. 建立你自己的SSL认证机构。你需要它来证明你自己的SSL证书请求（见下面的脚本）。

```
otrs@linux:~/tmp> CA.pl -newca
CA certificate filename (or enter to create)

Making CA certificate ...
Generating a 1024 bit RSA private key
...+++++
.....+++++
writing new private key to './demoCA/private/cakey.pem'
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
```

```

-----
Country Name (2 letter code) [AU]:DE
State or Province Name (full name) [Some-State]:OTRS-state
Locality Name (eg, city) []:OTRS-town
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Your company
Organizational Unit Name (eg, section) []:
Common Name (eg, YOUR name) []:OTRS Admin
Email Address []:otrs@your-domain.tld
otrs@linux:~/tmp> ls -la demoCA/
total 8
-rw-r--r--  1 otrs otrs 1330 2006-01-08 17:54 cacert.pem
drwxr-xr-x  2 otrs otrs  48 2006-01-08 17:53 certs
drwxr-xr-x  2 otrs otrs  48 2006-01-08 17:53 crl
-rw-r--r--  1 otrs otrs   0 2006-01-08 17:53 index.txt
drwxr-xr-x  2 otrs otrs  48 2006-01-08 17:53 newcerts
drwxr-xr-x  2 otrs otrs  80 2006-01-08 17:54 private
-rw-r--r--  1 otrs otrs  17 2006-01-08 17:54 serial
otrs@linux:~/tmp>

```

脚本：建立一个SSL认证机构。

## 2. 生成一个证书请求（见下面的脚本）。

```

otrs@linux:~/tmp> CA.pl -newreq
Generating a 1024 bit RSA private key
.....+++++
....+++++
writing new private key to 'newreq.pem'
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:DE\keyreturn
State or Province Name (full name) [Some-State]:OTRS-state
Locality Name (eg, city) []:OTRS-town
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Your company
Organizational Unit Name (eg, section) []:
Common Name (eg, YOUR name) []:OTRS admin
Email Address []:otrs@your-domain.tld

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
Request (and private key) is in newreq.pem
otrs@linux:~/tmp> ls -la
total 4
drwxr-xr-x  6 otrs otrs  232 2006-01-08 17:54 demoCA
-rw-r--r--  1 otrs otrs 1708 2006-01-08 18:04 newreq.pem
otrs@linux:~/tmp>

```

脚本：创建一个证书请求。

## 3. 证书请求的签名。证书请求要么用你自己的CA证明签名，要么使用另外的外部认证过的CA以获取更可信的签名（见下面的脚本）。

```

otrs@linux:~/tmp> CA.pl -signreq
Using configuration from /etc/ssl/openssl.cnf
Enter pass phrase for ./demoCA/private/cakey.pem:
Check that the request matches the signature
Signature ok

```

```

Certificate Details:
  Serial Number:
    fd:85:f6:9f:14:07:16:c8
  Validity
    Not Before: Jan  8 17:04:37 2006 GMT
    Not After : Jan  8 17:04:37 2007 GMT
  Subject:
    countryName           = DE
    stateOrProvinceName  = OTRS-state
    localityName         = OTRS-town
    organizationName     = Your Company
    commonName           = OTRS administrator
    emailAddress         = otrs@your-domain.tld
  X509v3 extensions:
    X509v3 Basic Constraints:
      CA:FALSE
    Netscape Comment:
      OpenSSL Generated Certificate
  X509v3 Subject Key Identifier:
    01:D9:1E:58:C0:6D:BF:27:ED:37:34:14:D6:04:AC:C4:64:98:7A:22
  X509v3 Authority Key Identifier:
    keyid:10:4D:8D:4C:93:FD:2C:AA:9A:B3:26:80:6B:F5:D5:31:E2:8E:DB:A8
    DirName:/C=DE/ST=OTRS-state/L=OTRS-town/O=Your Company/
    CN=OTRS admin/emailAddress=otrs@your-domain.tld
    serial:FD:85:F6:9F:14:07:16:C7

Certificate is to be certified until Jan  8 17:04:37 2007 GMT (365 days)
Sign the certificate? [y/n]:y

1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated
Signed certificate is in newcert.pem
otrs@linux:~/tmp>
  
```

脚本：证书请求的签名。

#### 4. 使用已经签名的证书请求生成你自己的证书和相关的的所有数据。

```

otrs@linux:~/tmp> CA.pl -pkcs12 "OTRS Certificate"
Enter pass phrase for newreq.pem:
Enter Export Password:
Verifying - Enter Export Password:
otrs@linux:~/tmp> ls -la
total 12
drwxr-xr-x  6 otrs otrs  328 2006-01-08 18:04 demoCA
-rw-r--r--  1 otrs otrs 3090 2006-01-08 18:13 newcert.p12
-rw-r--r--  1 otrs otrs 3791 2006-01-08 18:04 newcert.pem
-rw-r--r--  1 otrs otrs 1708 2006-01-08 18:04 newreq.pem
otrs@linux:~/tmp>
  
```

脚本：生成一个新证书。

现在已经执行了这些操作，还必须在OTRS中完成S/MIME设置。

这部分的设置在系统管理页面中进行，选择“S/MIME证书”链接。如果在OTRS中还没有启用S/MIME支持，会弹出遮罩窗口提示系统管理员并提供了启用它的链接。

通过系统配置组“Crypt::SMIME”，你还可以启用并配置通用S/MIME支持。

在这里你可以激活S/MIME支持，并定义OpenSSL命令的路径和证书目录。前面创建的密钥文件必须保存在这里指定的目录中，否则OpenSSL无法使用证书。

下一步是在系统管理页面的S/MIME证书链接中执行。在这里，你可以导入OTRS系统的私钥和其它通信伙伴的公钥。输入本节开始部分创建的公钥并添加到OTRS中。

显然，所有通信伙伴的S/MIME公钥也可以通过客户联系人管理工具导入。

### 4.3.1. 在客户用户后端获取S/MIME证书

可以使用客户联系人后端（例如 LDAP）作为公共S/MIME 证书的来源。这些证书可以导入到系统，并在 OTRS系统管理页面上的S/MIME 配置 显示。它们可用于从 OTRS 向客户发送加密的电子邮件。

为了启用此特性需要：

1. 在系统配置中启用SMIME
2. 在系统配置中启用 SMIME::FetchFromCustomer
3. 配置客户联系人后端以提供属性 UserSMIMECertificate的客户联系人的S/MIME 证书（在 \$OTRS\_HOME/Kernel/Config/Defaults.pm文件中有一个 LDAP 客户联系人映射的例子）。

该功能可通过三种不同的方式使用：

1. 收到的邮件：

一个专用的邮箱管理员过滤器(在系统配置PostMaster::PreFilterModule###000-SMIMEFetchFromCustomer 中)将提取每个传入的电子邮件的地址，并试图从客户清单中找到匹配的电子邮件地址。如果发现匹配，它将尝试从客户联系人属性获取S/MIME 证书，如果找到了证书它会尝试将其导入（除非它已经导入）。

2. 特定的邮件地址或所有客户：

控制台命令Maint::SMIME::CustomerCertificate::Fetch可用来导入一个客户邮件地址的S/MIME证书：

```
shell> perl /opt/otrs/bin/otrs.Console.pl Maint::SMIME::CustomerCertificate::Fetch --email customer@example.com
```

在这种情况下，控制台命令将尝试用提供的电子邮件地址与一个客户联系人的邮件地址进行匹配。如果发现匹配，它将尝试向系统中添加客户联系人属性中找到的S/MIME 证书（如果证书还没有添加）。

同样的控制台命令可用于导入所有客户联系人（受限于客户联系人后端的CustomerUserSearchListLimit 属性）的 S/MIME 证书。此选项是专为具有大量客户联系人的系统设计的，因为需要太多的时间来执行，根据限制它并不一定会获取所有的客户证书。在这种模式下按如下执行控制台命令：

```
shell> perl /opt/otrs/bin/otrs.Console.pl Maint::SMIME::CustomerCertificate::Fetch --add-all
```

使用此选项后控制台命令将客户联系人后端以获取所有可能的客户，检查找到的每个客户是否有S/MIME 证书。如果找到了证书，它将向系统中添加此S/MIME 证书（如果此证书还没有添加到系统）。

3. 更新已有的证书：

可以用另一个控制台命令 Maint::SMIME::CustomerCertificate::Renew 来检查系统中所有现有的证书。这将验证客户联系人现有的证书是否匹配通过客户联系人属性查到的证书。并将在客户联系人后端中找到的任何新证书添加到系统（在这一过程中不会删除证书）。

这个控制台命令由 OTRS 守护进程通过任务Daemon::SchedulerCronTaskManager::Task###RenewCustomerSMIMECertificates（如 系统配置所示）每天自动执行一次，但它也可以根据需求手动执行：

```
shell> perl /opt/otrs/bin/otrs.Console.pl Maint::SMIME::CustomerCertificate::Renew
```

## 5. 使用外部后端

### 5.1. 客户数据

OTRS会使用许多客户联系人数据属性如用户名、邮件地址、电话号码等等，这些属性在服务人员和客户联系人前端界面都会显示，还用于客户联系人的身份验证。

在OTRS中使用或显示的客户联系人数据是高度可定制的，但下面的信息始终需要用于客户联系人身份验证。

- 用户登录名
- 电子邮件地址
- 客户联系人ID

如果你要在服务人员界面显示客户联系人信息，使用下面的系统配置参数。

```
# Ticket::Frontend::CustomerInfo*
# (在编写电话/邮件工单、工单详情和队列视图窗口显示客户联系人信息)
$self->{'Ticket::Frontend::CustomerInfoCompose'} = 1;
$self->{'Ticket::Frontend::CustomerInfoZoom'} = 1;
```

脚本：系统配置参数

### 5.2. 客户联系人后端

你可以使用两种类型的客户联系人后端：DB和LDAP。如果你已经有另外的客户联系人后端（如SAP），当然可用写一个模块来使用它。

#### 5.2.1. 数据库(默认)

下面的示例显示一个客户联系人数据库后端的配置，使用了存储在OTRS数据库中的客户联系人数据。

#### 例 4.7. 配置客户联系人的数据库后端

```
# CustomerUser (customer database backend and settings)
$self->{CustomerUser} = {
  Name => 'Database Datasource',
  Module => 'Kernel::System::CustomerUser::DB',
  Params => {
    # if you want to use an external database, add the required settings
    #   DSN => 'DBI:odbc:yourdsn',
    #   Type => 'mssql', # only for ODBC connections
    #   DSN => 'DBI:mysql:database=customerdb;host=customerdbhost',
    #   User => '',
    #   Password => '',
    Table => 'customer_user',

    # CaseSensitive will control if the SQL statements need LOWER()
    #   function calls to work case insensitively. Setting this to
    #   1 will improve performance dramatically on large databases.
    CaseSensitive => 0,
  },
# customer unique id
CustomerKey => 'login',

# customer #
CustomerID => 'customer_id',
```

```

CustomerValid => 'valid_id',
CustomerUserListFields => ['first_name', 'last_name', 'email'],
CustomerUserSearchFields => ['login', 'last_name', 'customer_id'],
CustomerUserSearchPrefix => '',
CustomerUserSearchSuffix => '*',
CustomerUserSearchListLimit => 250,
CustomerUserPostMasterSearchFields => ['email'],
CustomerUserNameFields => ['title', 'first_name', 'last_name'],
CustomerUserEmailUniqCheck => 1,
# # show not own tickets in customer panel, CompanyTickets
# CustomerUserExcludePrimaryCustomerID => 0,
# # generate auto logins
# AutoLoginCreation => 0,
# AutoLoginCreationPrefix => 'auto',
# # admin can change customer preferences
# AdminSetPreferences => 1,
# # cache time to live in sec. - cache any database queries
# CacheTTL => 0,
# # just a read only source
# ReadOnly => 1,
Map => [
  # note: Login, Email and CustomerID needed!
  # var, frontend, storage, shown (1=always,2=lite), required, storage-type, http-
link, readonly, http-link-target, link class(es)
  [ 'UserTitle', 'Title', 'title', 1, 0, 'var', '', 0 ],
  [ 'UserFirstname', 'Firstname', 'first_name', 1, 1, 'var', '', 0 ],
  [ 'UserLastname', 'Lastname', 'last_name', 1, 1, 'var', '', 0 ],
  [ 'UserLogin', 'Username', 'login', 1, 1, 'var', '', 0 ],
  [ 'UserPassword', 'Password', 'pw', 0, 0, 'var', '', 0 ],
  [ 'UserEmail', 'Email', 'email', 1, 1, 'var', '', 0 ],
# [ 'UserEmail', Translatable('Email'), 'email', 1, 1, 'var', "[%
Env("CGIHandle") %]?Action=AgentTicketCompose;ResponseID=1;TicketID=[% Data.TicketID | uri
%];ArticleID=[% Data.ArticleID | uri %]", 0, '', 'AsPopup OTRSPopup_TicketAction' ],
  [ 'UserCustomerID', 'CustomerID', 'customer_id', 0, 1, 'var', '', 0 ],
# [ 'UserCustomerIDs', 'CustomerIDs', 'customer_ids', 1, 0, 'var', '', 0 ],
  [ 'UserPhone', 'Phone', 'phone', 1, 0, 'var', '', 0 ],
  [ 'UserFax', 'Fax', 'fax', 1, 0, 'var', '', 0 ],
  [ 'UserMobile', 'Mobile', 'mobile', 1, 0, 'var', '', 0 ],
  [ 'UserStreet', 'Street', 'street', 1, 0, 'var', '', 0 ],
  [ 'UserZip', 'Zip', 'zip', 1, 0, 'var', '', 0 ],
  [ 'UserCity', 'City', 'city', 1, 0, 'var', '', 0 ],
  [ 'UserCountry', 'Country', 'country', 1, 0, 'var', '', 0 ],
  [ 'UserComment', 'Comment', 'comments', 1, 0, 'var', '', 0 ],
  [ 'ValidID', 'Valid', 'valid_id', 0, 1, 'int', '', 0 ],

  # Dynamic field example
# [ 'DynamicField_Name_X', undef, 'Name_X', 0, 0, 'dynamic_field', undef, 0, undef,
undef, ],
],
# default selections
Selections => {
  UserTitle => {
    'Mr.' => 'Mr.',
    'Mrs.' => 'Mrs.',
  },
},
};

```

如果你要定制客户联系人数据，修改OTRS数据库中customer\_user表的列标题或添加一个新列。作为一个示例，下面的脚本显示了如果添加一个房间号的新列。

```

linux:~# mysql -p
Enter password:
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 116 to server version: 5.0.18-Debian_7-log

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> use otrs;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

```

```
Database changed
mysql> ALTER TABLE customer_user ADD room VARCHAR (250);
Query OK, 1 rows affected (0.01 sec)
Records: 1 Duplicates: 0 Warnings: 0

mysql> quit
Bye
linux:~#
```

脚本：添加一个房间字段到`customer_user`表。

现在添加新列到文件`Kernel/Config.pm`中的MAP数组，如下面的脚本所示。

```
# var, frontend, storage, shown (1=always,2=lite), required, storage-type, http-link,
readonly, http-link-target, link class(es)
[...]
[ 'UserRoom',      'Room',      'room',      0, 1, 'var', '', 0 ],
```

脚本：添加一个房间列到`Kernel/Config.pm`文件。

还可以在服务人员界面通过客户链接编辑所有的客户联系人信息。

## 注意

Please note that you may omit `http-link-target` and `link class` keys in map array elements, if they are not to be used. These keys add `target=""` and `class=""` attributes to the HTTP link element, respectively. They are ignored if `http-link` is not set.

### 5.2.1.1. 多ID的客户联系人（公司工单）

给一个客户联系人分配多个ID是可能的。这在一个客户联系人必须访问另一个客户联系人的工单时很有用，例如：一名主管想关注他的助手的工单。如果一个客户联系人能够访问另一个客户联系人的工单，需要使用OTRS的公司工单功能。可通过客户界面的“公司工单”链接来访问公司工单。

要使用公司工单，必须给OTRS数据库中的`customer_user`表添加一个新列，以保存一个客户联系人的ID。

```
linux:~# mysql -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 124 to server version: 5.0.18-Debian_7-log

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> use otrs;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> ALTER TABLE customer_user ADD customer_ids VARCHAR (250);
Query OK, 1 rows affected (0.02 sec)
Records: 1 Duplicates: 0 Warnings: 0

mysql> quit
Bye
linux:~#
```

脚本：给`customer_user`表添加一个`customer_ids`列。

现在必须将新列添加到文件`Kernel/Config.pm`中的MAP数组，如下面的脚本所示。

```
# var, frontend, storage, shown (1=always,2=lite), required, storage-type, http-link,
readonly, http-link-target, link class(es)
[...]
[ 'UserRoom',      'Room',      'room',      0, 1, 'var', '', 0 ],
```



```
[ 'UserCustomerIDs', 'CustomerIDs', 'customer_ids', 1, 0, 'var', '', 0 ],
```

脚本：在文件Kernel/Config.pm中添加UserCustomerIDs列。

现在，这个用于多ID的新列能够通过服务人员界面客户联系人管理部分进行编辑。

要确保一个客户联系人能够访问其它客户联系人的工单，将其它客户联系人的ID添加到这个新列，每个ID要用分号分隔（见下面的示例）。

#### 例 4.8. 通过数据库后端使用公司工单

系统中有客户联系人A、B和C，A想要通过客户界面访问B和C的工单，B和C不能访问其他人的工单。

要实现这个设置，修改表customer\_user并按以上描述修改文件Kernel/Config.pm中的映射。然后在服务人员界面中或者通过系统管理页面载入客户联系人A的设置，设置显示出来后，将CustomerIDs字段输入值“B;C;”。

## 5.2.2. LDAP

如果你有一个客户联系人的LDAP目录，你可以用它作为OTRS的客户后端，如下面的例子所示。

#### 例 4.9. 配置一个LDAP客户后端

```
# CustomerUser
# (customer ldap backend and settings)
$self->{CustomerUser} = {
  Name => 'LDAP Data Source',
  Module => 'Kernel::System::CustomerUser::LDAP',
  Params => {
    # ldap host
    Host => 'bay.csuhayward.edu',
    # ldap base dn
    BaseDN => 'ou=seas,o=csuh',
    # search scope (one|sub)
    SSCOPE => 'sub',
    # The following is valid but would only be necessary if the
    # anonymous user does NOT have permission to read from the LDAP tree
    UserDN => '',
    UserPw => '',
    # in case you want to add always one filter to each ldap query, use
    # this option. e. g. AlwaysFilter => '(mail=*)' or AlwaysFilter =>
    '(objectclass=user)'
    AlwaysFilter => '',
    # if the charset of your ldap server is iso-8859-1, use this:
    SourceCharset => 'iso-8859-1',

    # Net::LDAP new params (if needed - for more info see perldoc Net::LDAP)
    Params => {
      port => 389,
      timeout => 120,
      async => 0,
      version => 3,
    },
  },
  # customer unique id
  CustomerKey => 'uid',
  # customer #
  CustomerID => 'mail',
  CustomerUserListFields => ['cn', 'mail'],
  CustomerUserSearchFields => ['uid', 'cn', 'mail'],
  CustomerUserSearchPrefix => '',
  CustomerUserSearchSuffix => '*',
  CustomerUserSearchListLimit => 250,
  CustomerUserPostMasterSearchFields => ['mail'],
  CustomerUserNameFields => ['givenname', 'sn'],
  # show not own tickets in customer panel, CompanyTickets
  CustomerUserExcludePrimaryCustomerID => 0,
  # add an ldap filter for valid users (expert setting)
```

```
# CustomerUserValidFilter => '(!description=locked)',
# administrator can't change customer preferences
AdminSetPreferences => 0,
# cache time to live in sec. - cache any database queries
# CacheTTL => 0,
Map => [
  # note: Login, Email and CustomerID are mandatory!
  # var, frontend, storage, shown (1=always,2=lite), required, storage-type, http-
link, readonly, http-link-target, link class(es)
  [ 'UserTitle',      'Title',      'title',          1, 0, 'var', '', 0 ],
  [ 'UserFirstname', 'Firstname', 'givenname',     1, 1, 'var', '', 0 ],
  [ 'UserLastname',  'Lastname',  'sn',           1, 1, 'var', '', 0 ],
  [ 'UserLogin',     'Username',  'uid',          1, 1, 'var', '', 0 ],
  [ 'UserEmail',    'Email',    'mail',         1, 1, 'var', '', 0 ],
  [ 'UserCustomerID', 'CustomerID', 'mail',        0, 1, 'var', '', 0 ],
#   [ 'UserCustomerIDs', 'CustomerIDs', 'second_customer_ids', 1, 0, 'var', '', 0 ],
  [ 'UserPhone',    'Phone',    'telephonenumber', 1, 0, 'var', '', 0 ],
  [ 'UserAddress',  'Address',  'postaladdress',  1, 0, 'var', '', 0 ],
  [ 'UserComment',  'Comment',  'description',    1, 0, 'var', '', 0 ],
],
};
```

如果你的LDAP目录存储有附加的客户信息如经理的姓名、手机号或部分，并且你想在OTRS中显示这些信息，只需用这些属性条目扩展文件Kernel/Config.pm中的MAP数组即可，如下面的脚本所示。

```
# var, frontend, storage, shown (1=always,2=lite), required, storage-type, http-link,
readonly, http-link-target, link class(es)
[... ]
[ 'UserPhone',      'Phone',      'telephonenumber', 1, 0, 'var', '', 0 ],
```

脚本：添加新的字段到 *Kernel/Config.pm* 文件。

### 5.2.2.1. 多ID的客户联系人（公司工单）

使用LDAP后端时同样可以给一个客户联系人分配多个ID。要使用公司工单，需要在LDAP中添加一个新的字段以包含其它客户联系人的ID。

如果这个新的字段已经在LDAP目录中创建成功，必须将新的条目添加到文件Kernel/Config.pm的MAP数组中，如下面的脚本所示。

```
# var, frontend, storage, shown (1=always,2=lite), required, storage-type, http-link,
readonly
[... ]
[ 'UserCustomerIDs', 'CustomerIDs', 'customer_ids', 1, 0, 'var', '', 0 ],
```

脚本：映射新的字段到 *Kernel/Config.pm* 文件中。

用于多ID的字段必须在LDAP目录中直接编辑。OTRS只能从LDAP读取数据，而不能写入数据。

要确保一个客户联系人能够访问其它客户联系人的工单，将其它客户联系人的ID添加到这个在LDAP中的新字段，每个ID要用分号分隔（见下面的示例）。

#### 例 4.10. 通过LDAP后端使用公司工单

系统中有客户联系人A、B和C，A想要通过客户界面访问B和C的工单，B和C不能访问其他人的工单。

要实现这个设置，按上述方法修改LDAP目录和文件Kernel/Config.pm中的映射。然后在LDAP目录中为客户联系人A的新字段输入值“B;C;”。

### 5.2.3. 在OTRS中使用多个客户后端

如果你要在OTRS中利用多个客户数据源（例如：一个LDAP和一个数据库后端），配置参数CustomerUser的后面要加个数字后缀，如“CustomerUser1”、“CustomerUser2”（见下面的示例）。

## 例 4.11. 在OTRS中利用多个客户数据后端

下面的配置示例显示了OTRS中使用一个LDAP和一个数据库后端的用法。

```
# 1. Customer user backend: DB
# (customer database backend and settings)
$Self->{CustomerUser1} = {
  Name => 'Customer Database',
  Module => 'Kernel::System::CustomerUser::DB',
  Params => {
    # if you want to use an external database, add the
    # required settings
    DSN => 'DBI:odbc:yourdsn',
    Type => 'mssql', # only for ODBC connections
    DSN => 'DBI:mysql:database=customerdb;host=customerdbhost',
    User => '',
    Password => '',
    Table => 'customer_user',
  },
  # customer unique id
  CustomerKey => 'login',
  # customer #
  CustomerID => 'customer_id',
  CustomerValid => 'valid_id',
  CustomerUserListFields => ['first_name', 'last_name', 'email'],
  CustomerUserSearchFields => ['login', 'last_name', 'customer_id'],
  CustomerUserSearchPrefix => '',
  CustomerUserSearchSuffix => '*',
  CustomerUserSearchListLimit => 250,
  CustomerUserPostMasterSearchFields => ['email'],
  CustomerUserNameFields => ['title', 'first_name', 'last_name'],
  CustomerUserEmailUniqCheck => 1,
  # # show not own tickets in customer panel, CompanyTickets
  # CustomerUserExcludePrimaryCustomerID => 0,
  # # generate auto logins
  # AutoLoginCreation => 0,
  # AutoLoginCreationPrefix => 'auto',
  # # admin can change customer preferences
  # AdminSetPreferences => 1,
  # # cache time to live in sec. - cache any database queries
  # CacheTTL => 0,
  # # just a read only source
  # ReadOnly => 1,
  Map => [

    # note: Login, Email and CustomerID needed!
    # var, frontend, storage, shown (1=always,2=lite), required, storage-type, http-
link, readonly, http-link-target
    [ 'UserTitle',      'Title',      'title',      1, 0, 'var', '', 0 ],
    [ 'UserFirstname', 'Firstname',  'first_name', 1, 1, 'var', '', 0 ],
    [ 'UserLastname',  'Lastname',   'last_name',  1, 1, 'var', '', 0 ],
    [ 'UserLogin',     'Username',   'login',      1, 1, 'var', '', 0 ],
    [ 'UserPassword',  'Password',   'pw',         0, 0, 'var', '', 0 ],
    [ 'UserEmail',     'Email',      'email',      1, 1, 'var', '', 0 ],
    [ 'UserCustomerID', 'CustomerID', 'customer_id', 0, 1, 'var', '', 0 ],
    [ 'UserPhone',     'Phone',      'phone',      1, 0, 'var', '', 0 ],
    [ 'UserFax',       'Fax',        'fax',        1, 0, 'var', '', 0 ],
    [ 'UserMobile',    'Mobile',     'mobile',     1, 0, 'var', '', 0 ],
    [ 'UserStreet',    'Street',     'street',     1, 0, 'var', '', 0 ],
    [ 'UserZip',       'Zip',        'zip',        1, 0, 'var', '', 0 ],
    [ 'UserCity',      'City',       'city',       1, 0, 'var', '', 0 ],
    [ 'UserCountry',   'Country',    'country',    1, 0, 'var', '', 0 ],
    [ 'UserComment',   'Comment',    'comments',   1, 0, 'var', '', 0 ],
    [ 'ValidID',       'Valid',      'valid_id',   0, 1, 'int', '', 0 ],
  ],
  # default selections
  Selections => {
    UserTitle => {
      'Mr.' => 'Mr.',
      'Mrs.' => 'Mrs.',
    },
  },
}
```

```

    },
  };

# 2. Customer user backend: LDAP
# (customer ldap backend and settings)
$self->{CustomerUser2} = {
  Name => 'LDAP Datasource',
  Module => 'Kernel::System::CustomerUser::LDAP',
  Params => {
    # ldap host
    Host => 'bay.csuhayward.edu',
    # ldap base dn
    BaseDN => 'ou=seas,o=csuh',
    # search scope (one|sub)
    SSCOPE => 'sub',
    # The following is valid but would only be necessary if the
    # anonymous user does NOT have permission to read from the LDAP tree
    UserDN => '',
    UserPw => '',
    # in case you want to add always one filter to each ldap query, use
    # this option. e. g. AlwaysFilter => '(mail=*)' or AlwaysFilter =>
    '(objectclass=user)'
    AlwaysFilter => '',
    # if the charset of your ldap server is iso-8859-1, use this:
    # SourceCharset => 'iso-8859-1',

    # Net::LDAP new params (if needed - for more info see perldoc Net::LDAP)
    Params => {
      port => 389,
      timeout => 120,
      async => 0,
      version => 3,
    },
  },
  # customer unique id
  CustomerKey => 'uid',
  # customer #
  CustomerID => 'mail',
  CustomerUserListFields => ['cn', 'mail'],
  CustomerUserSearchFields => ['uid', 'cn', 'mail'],
  CustomerUserSearchPrefix => '',
  CustomerUserSearchSuffix => '*',
  CustomerUserSearchListLimit => 250,
  CustomerUserPostMasterSearchFields => ['mail'],
  CustomerUserNameFields => ['givenname', 'sn'],
  # show not own tickets in customer panel, CompanyTickets
  CustomerUserExcludePrimaryCustomerID => 0,
  # add a ldap filter for valid users (expert setting)
  # CustomerUserValidFilter => '(!description=locked)',
  # admin can't change customer preferences
  AdminSetPreferences => 0,
  Map => [
    # note: Login, Email and CustomerID needed!
    # var, frontend, storage, shown (1=always,2=lite), required, storage-type, http-
    link, readonly
    [ 'UserTitle',      'Title',      'title',          1, 0, 'var', '', 0 ],
    [ 'UserFirstname', 'Firstname', 'givenname',     1, 1, 'var', '', 0 ],
    [ 'UserLastname',  'Lastname',  'sn',           1, 1, 'var', '', 0 ],
    [ 'UserLogin',     'Username',  'uid',          1, 1, 'var', '', 0 ],
    [ 'UserEmail',     'Email',     'mail',         1, 1, 'var', '', 0 ],
    [ 'UserCustomerID', 'CustomerID', 'mail',         0, 1, 'var', '', 0 ],
    #
    [ 'UserCustomerIDs', 'CustomerIDs', 'second_customer_ids', 1, 0, 'var', '', 0 ],
    [ 'UserPhone',     'Phone',     'telephonenumber', 1, 0, 'var', '', 0 ],
    [ 'UserAddress',   'Address',   'postaladdress',  1, 0, 'var', '', 0 ],
    [ 'UserComment',   'Comment',   'description',    1, 0, 'var', '', 0 ],
  ],
};

```

最多可以集成10个不同的客户数据后端，在OTRS中使用客户管理界面来查看或编辑（需要写入权限）所有的客户联系人数据。

## 5.2.4. 在动态字段中存储客户联系人数据

有些时候在工单中直接存储客户联系人数据也是有用的，例如在这些数据上创建一些特殊的统计。

当创建一个工单或修改了工单的客户联系人时设置动态字段的值。动态字段的值来源于客户联系人数据，这适用于所有的客户数据后端，但对LDAP后端特别有用。

To activate this optional feature of OTRS, please activate the settings "Ticket::EventModulePost###950-DynamicFieldFromCustomerUser" and "DynamicFieldFromCustomerUser::Mapping". The latter setting contains the configuration of which CustomerUser field entry should be stored in which ticket dynamic field. The fields must be present in the system and should be enabled for AgentTicketFreeText, so that they can be set manually. They mustn't be enabled for AgentTicketPhone, AgentTicketEmail and AgentTicketCustomer. If they were, they would have precedence over the automatically set values.

## 5.3. 服务人员和客户联系人身份认证后端

OTRS提供了不同的服务人员和客户联系人身份认证后端。

### 5.3.1. 服务人员身份认证后端

#### 5.3.1.1. 数据库认证(默认)

服务人员身份认证后端默认是OTRS数据库，可以通过系统管理页面中的服务人员管理界面来添加和编辑服务人员（见下面的例子）。

#### 例 4.12. 使用数据库后端认证服务人员

```
$Self->{'AuthModule'} = 'Kernel::System::Auth::DB';
```

#### 5.3.1.2. LDAP

如果你所有的服务人员数据存储存储在LDAP目录中，你可以使用LDAP模块来对OTRS中的用户进行身份认证（见下面例子）。这个模块只对LDAP树有读取权限，这意味着你不能通过服务人员管理界面来编辑用户数据。

#### 例 4.13. 使用LDAP后端认证服务人员

```
# This is an example configuration for an LDAP auth. backend.
# (Make sure Net::LDAP is installed!)
$Self->{'AuthModule'} = 'Kernel::System::Auth::LDAP';
$Self->{'AuthModule::LDAP::Host'} = 'ldap.example.com';
$Self->{'AuthModule::LDAP::BaseDN'} = 'dc=example,dc=com';
$Self->{'AuthModule::LDAP::UID'} = 'uid';

# Check if the user is allowed to auth in a posixGroup
# (e. g. user needs to be in a group xyz to use otrs)
$Self->{'AuthModule::LDAP::GroupDN'} = 'cn=otrsallow,ou=posixGroups,dc=example,dc=com';
$Self->{'AuthModule::LDAP::AccessAttr'} = 'memberUid';
# for ldap posixGroups objectclass (just uid)
# $Self->{'AuthModule::LDAP::UserAttr'} = 'UID';
# for non ldap posixGroups objectclass (with full user dn)
# $Self->{'AuthModule::LDAP::UserAttr'} = 'DN';

# The following is valid but would only be necessary if the
# anonymous user do NOT have permission to read from the LDAP tree
$Self->{'AuthModule::LDAP::SearchUserDN'} = '';
$Self->{'AuthModule::LDAP::SearchUserPw'} = '';
```

```
# in case you want to add always one filter to each ldap query, use
# this option. e. g. AlwaysFilter => '(mail=*)' or AlwaysFilter => '(objectclass=user)'
$self->{'AuthModule::LDAP::AlwaysFilter'} = '';

# in case you want to add a suffix to each login name, then
# you can use this option. e. g. user just want to use user but
# in your ldap directory exists user@domain.com
#   $self->{'AuthModule::LDAP::UserSuffix'} = '@domain.com';

# Net::LDAP new params (if needed - for more info see perldoc Net::LDAP)
$self->{'AuthModule::LDAP::Params'} = {
    port => 389,
    timeout => 120,
    async => 0,
    version => 3,
};
```

下面脚本中显示的配置参数可用来从LDAP目录同步用户数据到本地OTRS数据库。这减少了到LDAP的请求时间，加快了OTRS身份认证的速度。当服务人员首次认证后就会完成数据同步。虽然数据已经同步到本地的OTRS数据库，但LDAP目录是身份认证的最后实例，所以一个在LDAP中非活动的用户是不能通过OTRS认证的，即使帐户数据已经存储到OTRS数据库中。不能通过OTRS的WEB界面编辑LDAP中的服务人员数据，所以需要直接在LDAP目录中管理这些数据。

```
# defines AuthSyncBackend (AuthSyncModule) for AuthModule
# if this key exists and is empty, there won't be a sync.
# example values: AuthSyncBackend, AuthSyncBackend2
$self->{'AuthModule::UseSyncBackend'} = 'AuthSyncBackend';

# agent data sync against ldap
$self->{'AuthSyncModule'} = 'Kernel::System::Auth::Sync::LDAP';
$self->{'AuthSyncModule::LDAP::Host'} = 'ldap://ldap.example.com/';
$self->{'AuthSyncModule::LDAP::BaseDN'} = 'dc=otrs, dc=org';
$self->{'AuthSyncModule::LDAP::UID'} = 'uid';
$self->{'AuthSyncModule::LDAP::SearchUserDN'} = 'uid=sys, ou=user, dc=otrs, dc=org';
$self->{'AuthSyncModule::LDAP::SearchUserPw'} = 'some_pass';
$self->{'AuthSyncModule::LDAP::UserSyncMap'} = {
    # DB -> LDAP
    UserFirstname => 'givenName',
    UserLastname  => 'sn',
    UserEmail     => 'mail',
};
[...]

# AuthSyncModule::LDAP::UserSyncInitialGroups
# (sync following group with rw permission after initial create of first agent
# login)
$self->{'AuthSyncModule::LDAP::UserSyncInitialGroups'} = [
    'users',
];
```

脚本：同步LDAP目录中的用户数据到OTRS数据库。

做为选择，你可以使用LDAP组信息来确定OTRS中的组成员或角色。更多的信息和示例，请查看Kernel/Config/Defaults.pm，这里有一个同步LDAP组信息到OTRS组的例子。

```
# Attributes needed for group syncs
# (attribute name for group value key)
$self->{'AuthSyncModule::LDAP::AccessAttr'} = 'memberUid';
# (select the attribute for type of group content UID/DN for full ldap name)
# $self->{'AuthSyncModule::LDAP::UserAttr'} = 'UID';
# $self->{'AuthSyncModule::LDAP::UserAttr'} = 'DN';

AuthSyncModule::LDAP::UserSyncGroupsDefinition
# (If "LDAP" was selected for AuthModule and you want to sync LDAP
# groups to otrs groups, define the following.)
$self->{'AuthSyncModule::LDAP::UserSyncGroupsDefinition'} = {
    # your ldap group
```

```
'cn=agent,o=otrs' => {
  # otrs group(s)
  'admin' => {
    # permission
    rw => 1,
    ro => 1,
  },
  'faq' => {
    rw => 0,
    ro => 1,
  },
},
'cn=agent2,o=otrs' => {
  'users' => {
    rw => 1,
    ro => 1,
  },
}
};
```

### 5.3.1.3. 用于服务人员的HTTPBasicAuth模块

如果你要为所有的服务人员部署一套“单点登录”解决方案，可以使用HTTP基本认证（对所有需要单点登录的系统）和OTRS的HTTPBasicAuth模块（见下面的例子）。

#### 例 4.14. 使用HTTPBasic认证服务人员

```
# This is an example configuration for an apache ($ENV{REMOTE_USER})
# auth. backend. Use it if you want to have a single login through
# apache http-basic-auth
$self->{'AuthModule'} = 'Kernel::System::Auth::HTTPBasicAuth';

# Note:
#
# If you use this module, you should use as fallback
# the following configuration settings if the user is not authorized
# apache ($ENV{REMOTE_USER})
$self->{'LoginURL'} = 'http://host.example.com/not-authorized-for-otrs.html';
$self->{'LogoutURL'} = 'http://host.example.com/thanks-for-using-otrs.html';
```

### 5.3.1.4. Radius

下面例子中显示的配置参数使用一个Radius服务器来认证服务人员。

#### 例 4.15. 用Radius后端认证服务人员

```
# This is example configuration to auth. agents against a radius server
$self->{'AuthModule'} = 'Kernel::System::Auth::Radius';
$self->{'AuthModule::Radius::Host'} = 'radiushost';
$self->{'AuthModule::Radius::Password'} = 'radiussecret';
```

## 5.3.2. 客户联系人身份认证后端

### 5.3.2.1. 数据库(默认)

OTRS中客户联系人身份认证后端默认是OTRS数据库，可以通过OTRS的WEB界面编辑所有的客户联系人数据（见下面的例子）。

#### 例 4.16. 用数据库后端认证客户联系人

```
# This is the auth. module against the otrs db
```

```

$Self->{'Customer::AuthModule'} = 'Kernel::System::CustomerAuth::DB';
$Self->{'Customer::AuthModule::DB::Table'} = 'customer_user';
$Self->{'Customer::AuthModule::DB::CustomerKey'} = 'login';
$Self->{'Customer::AuthModule::DB::CustomerPassword'} = 'pw';
#$Self->{'Customer::AuthModule::DB::DSN'} =
  "DBI:mysql:database=customerdb;host=customerdbhost";
#$Self->{'Customer::AuthModule::DB::User'} = "some_user";
#$Self->{'Customer::AuthModule::DB::Password'} = "some_password";

```

### 5.3.2.2. LDAP

如果你所有的客户联系人数据在LDAP目录中，可以使用OTRS的LDAP模块来认证你的客户联系人（见下面的例子）。因为这个模块只有LDAP后端的只读权限，所以不能通过OTRS的WEB界面编辑客户联系人数据。

#### 例 4.17. 用LDAP后端认证客户联系人

```

# This is an example configuration for an LDAP auth. backend.
# (make sure Net::LDAP is installed!)
$Self->{'Customer::AuthModule'} = 'Kernel::System::CustomerAuth::LDAP';
$Self->{'Customer::AuthModule::LDAP::Host'} = 'ldap.example.com';
$Self->{'Customer::AuthModule::LDAP::BaseDN'} = 'dc=example,dc=com';
$Self->{'Customer::AuthModule::LDAP::UID'} = 'uid';

# Check if the user is allowed to auth in a posixGroup
# (e. g. user needs to be in a group xyz to use otrs)
$Self->{'Customer::AuthModule::LDAP::GroupDN'} =
  'cn=otrsallow,ou=posixGroups,dc=example,dc=com';
$Self->{'Customer::AuthModule::LDAP::AccessAttr'} = 'memberUid';
# for ldap posixGroups objectclass (just uid)
$Self->{'Customer::AuthModule::LDAP::UserAttr'} = 'UID';
# for non ldap posixGroups objectclass (full user dn)
#$Self->{'Customer::AuthModule::LDAP::UserAttr'} = 'DN';

# The following is valid but would only be necessary if the
# anonymous user does NOT have permission to read from the LDAP tree
$Self->{'Customer::AuthModule::LDAP::SearchUserDN'} = '';
$Self->{'Customer::AuthModule::LDAP::SearchUserPw'} = '';

# in case you want to add always one filter to each ldap query, use
# this option. e. g. AlwaysFilter => '(mail=*)' or AlwaysFilter => '(objectclass=user)'
$Self->{'Customer::AuthModule::LDAP::AlwaysFilter'} = '';

# in case you want to add a suffix to each customer login name, then
# you can use this option. e. g. user just want to use user but
# in your ldap directory exists user@domain.com
#$Self->{'Customer::AuthModule::LDAP::UserSuffix'} = '@domain.com';

# Net::LDAP new params (if needed - for more info see perldoc Net::LDAP)
$Self->{'Customer::AuthModule::LDAP::Params'} = {
  port => 389,
  timeout => 120,
  async => 0,
  version => 3,
};

```

### 5.3.2.3. 用于客户联系人的HTTPBasicAuth模块

如果你要为所有的客户联系人部署一套“单点登录”解决方案，可以使用HTTP基本认证（对所有需要单点登录的系统）和OTRS的HTTPBasicAuth模块（客户联系人使用OTRS不再需要登录），见下面的例子。

#### 例 4.18. 为客户联系人配置HTTP基本认证

```

# This is an example configuration for an apache (${ENV{REMOTE_USER}})
# auth. backend. Use it if you want to have a single login through

```



```
# apache http-basic-auth
$self->{'Customer::AuthModule'} = 'Kernel::System::CustomerAuth::HTTPBasicAuth';

# Note:
# If you use this module, you should use the following
# config settings as fallback, if user isn't login through
# apache ($ENV{REMOTE_USER})
$self->{'CustomerPanelLoginURL'} = 'http://host.example.com/not-authorized-for-otrs.html';
$self->{'CustomerPanelLogoutURL'} = 'http://host.example.com/thanks-for-using-otrs.html';
```

### 5.3.2.4. Radius

下面例子中显示的配置参数使用一个Radius服务器来认证客户联系人。

#### 例 4.19. 使用一个Radius后端的客户联系人身份认证

```
# This is a example configuration to auth. customer against a radius server
$self->{'Customer::AuthModule'} = 'Kernel::System::Auth::Radius';
$self->{'Customer::AuthModule::Radius::Host'} = 'radiushost';
$self->{'Customer::AuthModule::Radius::Password'} = 'radiussecret';
```

## 5.4. 定制客户联系人自助注册

可以定制新的客户联系人通过customer.pl面板访问的自助注册。可以添加新的可选或必填字段如房间号、地址或状态等。

下面的例子显示了如何在客户联系人数据库中指定一个必填字段，在这个例子中是存储客户联系人的房间号。

### 5.4.1. 定制WEB界面

To display the new field for the room number in the customer.pl web interface, the .tt file responsible for the layout in this interface has to be modified. Edit the Kernel/Output/HTML/Templates/Standard/CustomerLogin.tt file, adding the new field around line 130 (see Script below).

```
[...]
<div class="NewLine">
  <label for="Room">[% Translate("Room{CustomerUser}") | html %]</label>
  <input title="[% Translate("Room Number") | html %]" name="Room" type="text"
  id="UserRoom" maxlength="50" />
</div>
[...]
```

脚本：在WEB界面显示一个新字段。

### 5.4.2. 客户联系人映射

下一步用新的房间号条目扩展客户联系人映射。为确保OTRS更新后不会丢失自定义的修改内容，将“CustomerUser”设置从文件Kernel/Config/Defaults.pm放到文件Kernel/Config.pm中。现在修改MAP数组并添加新的房间号字段，如下面脚本所示。

```
# CustomerUser
# (customer database backend and settings)
$self->{'CustomerUser'} = {
  Name => 'Database Backend',
  Module => 'Kernel::System::CustomerUser::DB',
  Params => {
    # if you want to use an external database, add the
    # required settings
```

```

#      DSN => 'DBI:odbc:yourdsn',
#      Type => 'mssql', # only for ODBC connections
#      DSN => 'DBI:mysql:database=customerdb;host=customerdbhost',
#      User => '',
#      Password => '',
#      Table => 'customer_user',
},
# customer unique id
CustomerKey => 'login',
# customer #
CustomerID => 'customer_id',
CustomerValid => 'valid_id',
CustomerUserListFields => ['first_name', 'last_name', 'email'],
# CustomerUserListFields => ['login', 'first_name', 'last_name', 'customer_id', 'email'],
CustomerUserSearchFields => ['login', 'last_name', 'customer_id'],
CustomerUserSearchPrefix => '',
CustomerUserSearchSuffix => '*',
CustomerUserSearchListLimit => 250,
CustomerUserPostMasterSearchFields => ['email'],
CustomerUserNameFields => ['title', 'first_name', 'last_name'],
CustomerUserEmailUniqCheck => 1,
# # show not own tickets in customer panel, CompanyTickets
# CustomerUserExcludePrimaryCustomerID => 0,
# # generate auto logins
# AutoLoginCreation => 0,
# AutoLoginCreationPrefix => 'auto',
# # admin can change customer preferences
# AdminSetPreferences => 1,
# # cache time to live in sec. - cache database queries
# CacheTTL => 0,
# # just a read only source
# ReadOnly => 1,
Map => [

    # note: Login, Email and CustomerID needed!
    # var, frontend, storage, shown (1=always,2=lite), required, storage-type, http-
link, readonly, http-link-target
    [ 'UserTitle',      'Title',      'title',      1, 0, 'var', '', 0 ],
    [ 'UserFirstname', 'Firstname',  'first_name', 1, 1, 'var', '', 0 ],
    [ 'UserLastname',  'Lastname',  'last_name',  1, 1, 'var', '', 0 ],
    [ 'UserLogin',     'Username',  'login',      1, 1, 'var', '', 0 ],
    [ 'UserPassword',  'Password',  'pw',         0, 0, 'var', '', 0 ],
    [ 'UserEmail',     'Email',     'email',      1, 1, 'var', '', 0 ],
    [ 'UserCustomerID', 'CustomerID', 'customer_id', 0, 1, 'var', '', 0 ],
    [ 'UserPhone',     'Phone',     'phone',      1, 0, 'var', '', 0 ],
    [ 'UserFax',       'Fax',       'fax',        1, 0, 'var', '', 0 ],
    [ 'UserMobile',    'Mobile',    'mobile',     1, 0, 'var', '', 0 ],
    [ 'UserRoom',      'Room',      'room',       1, 0, 'var', '', 0 ],
    [ 'UserStreet',    'Street',    'street',     1, 0, 'var', '', 0 ],
    [ 'UserZip',       'Zip',       'zip',        1, 0, 'var', '', 0 ],
    [ 'UserCity',      'City',      'city',       1, 0, 'var', '', 0 ],
    [ 'UserCountry',   'Country',   'country',    1, 0, 'var', '', 0 ],
    [ 'UserComment',   'Comment',   'comments',   1, 0, 'var', '', 0 ],
    [ 'ValidID',       'Valid',     'valid_id',   0, 1, 'int', '', 0 ],
],
# default selections
Selections => {
    UserTitle => {
        'Mr.' => 'Mr.',
        'Mrs.' => 'Mrs.',
    },
},
};

```

脚本：修改映射数组。

### 5.4.3. 定制OTRS数据库中的customer\_user表

最后一步是将新的房间号列添加到OTRS数据库（见下面的脚本）。这个列会存储房间号信息。

```
linux:~# mysql -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 6 to server version: 5.0.18-Debian_7-log

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> use otrs;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> ALTER TABLE customer_user ADD room VARCHAR (200);
Query OK, 3 rows affected (0.01 sec)
Records: 3  Duplicates: 0  Warnings: 0

mysql> quit
Bye
linux:~#
```

脚本: 添加一个新列到`customer_user`表。

现在新的房间字段（若填写了）会在客户信息面板中显示，在客户联系人管理窗口也会显示。另外，新的客户联系人在注册新帐户时应该输入他们的房间号。

## 6. 工单设置

### 6.1. 工单状态

#### 6.1.1. 预定义状态

OTRS允许你修改预定义的工单状态和它们的类型，甚至添加新的状态。一个状态有两个重要属性：状态名称和状态类型。

OTRS中默认状态有：‘成功关闭’、‘失败关闭’、‘已合并’、‘新建’、‘处理中’、‘挂起等待成功关闭’、‘挂起等待失败关闭’、‘挂起提醒’和‘已删除’。

##### 6.1.1.1. New（新建）

从进入邮件创建的工单通常是这个状态。

##### 6.1.1.2. Open（处理中）

工单分配到队列和服务人员后的默认状态。

##### 6.1.1.3. Pending reminder（挂起提醒）

在挂起时间过期后，工单所有者会收到一个关于该工单的提醒邮件。如果工单没有锁定，这个提醒会发送给队列中的所有服务人员。提醒工单只在上班时间发送，并每24小时重复发送直到服务人员修改工单状态为止。工单在这个状态期间消耗的时间会计入升级时间。

##### 6.1.1.4. Pending auto close-（挂起等待失败关闭）

这个状态的工单如果挂起时间过期会被设置为“失败关闭”。工单在这个状态期间消耗的时间会计入升级时间。

##### 6.1.1.5. Pending auto close+（挂起等待成功关闭）

这个状态的工单如果挂起时间过期会被设置为“成功关闭”。工单在这个状态期间消耗的时间会计入升级时间。

### 6.1.1.6. Merged (已合并)

这个状态表明该工单已与其它工单合并。

### 6.1.1.7. 已删除

这是工单已被客户删除的状态。工单不会真的被删除，它们只是不能再显示为处理。为了在客户界面启用这个状态，你需要添加“已删除”的状态类型到系统配置设置"Ticket::Frontend::CustomerTicketZoom ###StateType"中。

### 6.1.1.8. Closed Successful (成功关闭)

这是已经成功解决了的工单的最终状态。取决于你的配置，可以或不可以重新处理已关闭的工单。

### 6.1.1.9. Closed Unsuccessful (失败关闭)

这是没有成功解决的工单的最终状态。取决于你的配置，可以或不可以重新处理已关闭的工单。

## 6.1.2. 定制状态

每个状态有一个名称（状态名称）和一个类型（状态类型）。在系统管理页面点击状态链接，并点击按钮“添加状态”来创建一个新的状态。你可以随意选择一个新状态的名称。不能通过WEB界面修改状态类型。如果你要添加新的状态类型或修改已有类型的名称必须通过数据库直接修改。通常不要修改默认的状态类型，因为这可能产生不可预知的结果。比如说，升级计算和解锁功能都是基于特定的状态类型。

通过这个窗口可以修改已经存在的状态的名称或者添加新的状态。如果通过WEB界面修改了状态“新建”，这个修改还必须通过配置文件Kernel/Config.pm或通过系统配置界面进行配置。必须修改下面脚本中指定的设置以确保OTRS能使用“新建”状态的修改内容正常工作。

```
[...]
# PostmasterDefaultState
# (The default state of new tickets.) [default: new]
$self->{PostmasterDefaultState} = 'new';

# CustomerDefaultState
# (default state of new customer tickets)
$self->{CustomerDefaultState} = 'new';
[...]
```

脚本：修改文件Kernel/Config.pm中的设置。

如果要添加一个新的状态，需要使用一个数据库客户端修改OTRS数据库中的ticket\_state\_type表，如下面脚本所示。

```
linux:~# mysql -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 23 to server version: 5.0.16-Debian_1-log

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> use otrs;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> insert into ticket_state_type (name,comments) values ('own','Own
state type');
Query OK, 1 row affected (0.00 sec)

mysql> quit
```

```
Bye  
linux:~#
```

脚本：修改OTRS数据库。

现在就可以使用你刚刚创建的新状态类型了。在一个状态已经链接到这个新状态类型后，OTRS配置还必须修改以确保这个新状态可用。只需通过系统配置修改下面的选项即可：

Ticket -> Frontend::Agent::Ticket::ViewPhoneNew > AgentTicketPhone###StateDefault - 定义新建电话工单默认的下一状态。

Ticket -> Frontend::Agent::Ticket::ViewPhoneNew > AgentTicketPhone###StateType - 定义新建电话工单可用的下一状态。

Ticket -> Frontend::Agent::Ticket::ViewEmailNew > AgentTicketEmail###StateDefault - 定义新建邮件工单默认的下一状态。

Ticket -> Frontend::Agent::Ticket::ViewEmailNew > AgentTicketEmail###StateType - 定义新建邮件工单可用的下一状态。

Ticket -> Frontend::Agent::Ticket::ViewPhoneOutbound > AgentTicketPhoneOutbound###State - 定义新建电话信件默认的下一状态。

Ticket -> Frontend::Agent::Ticket::ViewPhoneOutbound > AgentTicketPhoneOutbound###StateType - 定义新建电话信件可用的下一状态。

Ticket -> Frontend::Agent::Ticket::ViewMove > AgentTicketMove###State - 定义转移一个工单默认的下一状态。

Ticket -> Frontend::Agent::Ticket::ViewMove > AgentTicketMove###StateType - 定义转移一个工单可用的下一状态。

Ticket -> Frontend::Agent::Ticket::ViewBounce > StateDefault - 定义退回一个工单默认的下一状态。

Ticket -> Frontend::Agent::Ticket::ViewBounce > StateType - 定义退回一个工单可用的下一状态。

Ticket -> Frontend::Agent::Ticket::ViewBulk > StateDefault - 定义批量操作默认的下一状态。

Ticket -> Frontend::Agent::Ticket::ViewBulk > StateType - 定义批量操作窗口可用的下一状态。

Ticket -> Frontend::Agent::Ticket::ViewClose > StateDefault - 定义关闭一个工单时默认的下一状态。

Ticket -> Frontend::Agent::Ticket::ViewClose > StateType - 定义关闭工单窗口可用的下一状态。

Ticket -> Frontend::Agent::Ticket::ViewCompose > StateDefault - 定义编写（答复）窗口默认的下一状态。

Ticket -> Frontend::Agent::Ticket::ViewCompose > StateType - 定义编写（答复）窗口可用的下一状态。

Ticket -> Frontend::Agent::Ticket::ViewForward > StateDefault - 定义工单转发窗口默认的下一状态。

Ticket -> Frontend::Agent::Ticket::ViewForward > StateType - 定义工单转发窗口可用的下一状态。

Ticket -> Frontend::Agent::Ticket::ViewForward > StateDefault - 定义工单自定义窗口默认的下一状态。

Ticket -> Frontend::Agent::Ticket::ViewForward > StateType - 定义工单自定义窗口可用的下一状态。

Ticket -> Core::PostMaster > PostmasterDefaultState - 定义从邮件创建的工单的状态。

Ticket -> Core::PostMaster > PostmasterFollowUpState - 定义收到一个跟进后的工单状态。

Ticket -> Core::PostMaster > PostmasterFollowUpStateClosed - 定义一个已关闭工单收到一个跟进后的工单状态。

Ticket -> Core::Ticket > ViewableStateType - 定义在系统中各处显示的状态类型，例如在队列视图中。

Ticket -> Core::Ticket > UnlockStateType - 定义解锁工单的状态类型。

Ticket -> Core::Ticket > PendingReminderStateType - 定义提醒工单的状态类型。

Ticket -> Core::Ticket > PendingAutoStateType - 定义挂起等待自动（成功/失败关闭）工单的状态类型。

Ticket -> Core::Ticket > StateAfterPending - 定义一个工单在挂起等待自动（成功/失败关闭）计时器过期后设置的状态。

## 6.2. 工单优先级

OTRS自带5个默认的优先级级别，可以通过系统管理页面的“优先级”链接修改。当创建优先级的定制列表时，请牢记它们在用户界面的优先级选择框中是按字母顺序排序的。此外，OTRS在队列视图中是通过优先级的数据库内部ID对工单排序的。

### 注意

与OTRS其它条目一样，优先级不能删除，只能通过设置有效性标记为无效或临时无效来去激活。

### 重要

如果添加了一个新的优先级或修改了一个已有的优先级，你还要修改系统配置中的一些值：

- Ticket:Core::Postmaster::PostmasterDefaultPriority - 定义所有进入邮件的默认优先级。
- Ticket:Frontend::Agent:Ticket::ViewPhoneNew:Priority - 定义服务人员新建电话工单窗口的默认优先级。
- Ticket:Frontend::Agent:Ticket::ViewEmailNew:Priority - 定义服务人员新建邮件工单窗口的默认优先级。
- Ticket:Frontend::Customer:Ticket::ViewNew:PriorityDefault - 定义客户联系人界面新建工单窗口的默认优先级。

## 6.3. 工单负责人 & 工单关注

从OTRS 2.1开始，可以指定一个人来负责一个工单，除了它的所有者之外。另外，所有与工单相关的活动都能被除了工单所有者之外的人关注。这两个功能被部署为TicketResponsible（工单负责人）和TicketWatcher（工单关注人），它们使任务分配更容易且更适合分层的团队架构。

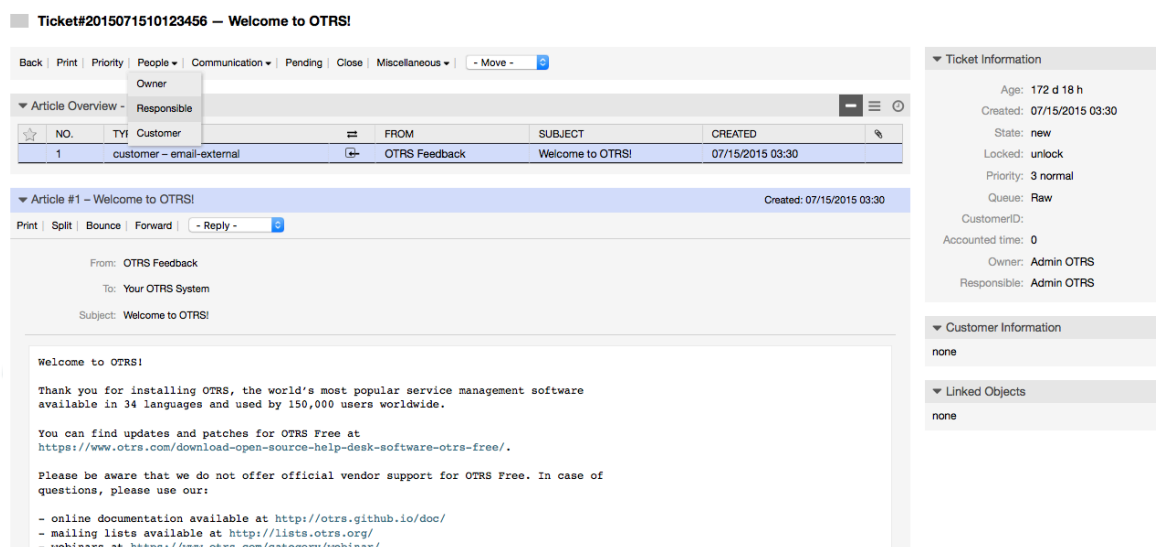
### 6.3.1. 工单负责人

工单负责人功能使一个工单所有者之外的服务人员容易完成一个工单处理。这样一个锁定了一个工单的服务人员可以将工单传递给另一个不是工单所有者的服务人员，以便作为第二个响应客户请求的人员。在请求被处理完成后，第一个服务人员可以从第二个服务人员收回工单负责人。

通过配置参数Ticket::Responsible可以激活工单负责人功能。这会产生3个新的链接显示在服务人员界面工单详情窗口的菜单中。

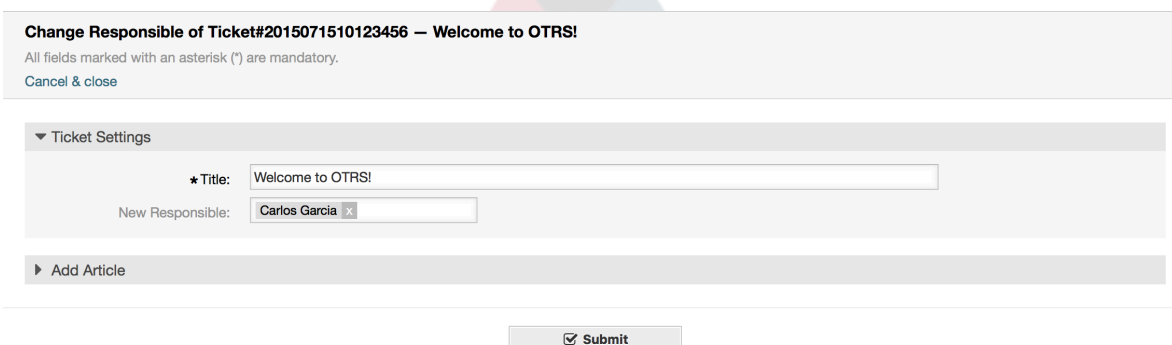
在服务人员界面工单详情窗口的菜单中点击“人员”-“负责人”链接可以指定工单负责人（见下图）。

图 4.67. 在工单详情窗口变更一个工单的负责人



点击“负责人”后，会打开一个弹出窗口以变更处理该工单的负责人（见下图）。这个对话框还可以用来给新的负责人发送一个消息。

图 4.68. 变更一个工单的负责人的弹出对话框



一旦激活了工单负责人功能，就可以通过服务人员界面仪表盘上各个小部件的“我负责的工单”视图查看一个服务人员负责的工单列表。

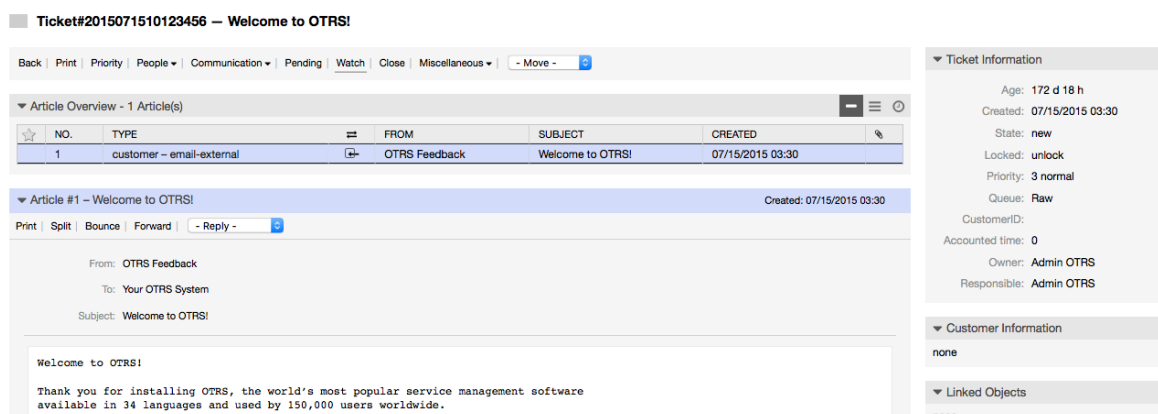
### 6.3.2. 工单关注

从OTRS 2.1开始，通过工单关注人功能，服务人员如主管可以关注系统中某些不是由他们处理的工单。

工单关注人功能通过配置参数Ticket::Watcher激活，激活后会在操作菜单中添加一个新的链接。通过配置参数Ticket::WatcherGroup还可以定义一个或多个有权关注工单的用户组。

为了关注一个工单，进入该工单的详情窗口并点击菜单中的“关注”链接（见下图）。

**图 4.69.** 在一个工单的详情窗口添加关注



**Ticket#2015071510123456 – Welcome to OTRS!**

Back | Print | Priority | People | Communication | Pending | **Watch** | Close | Miscellaneous | - Move -

Article Overview - 1 Article(s)

NO.	TYPE	FROM	SUBJECT	CREATED
1	customer – email-external	OTRS Feedback	Welcome to OTRS!	07/15/2015 03:30

Article #1 – Welcome to OTRS! Created: 07/15/2015 03:30

Print | Split | Bounce | Forward | - Reply -

From: OTRS Feedback  
 To: Your OTRS System  
 Subject: Welcome to OTRS!

Welcome to OTRS!  
 Thank you for installing OTRS, the world's most popular service management software available in 34 languages and used by 150,000 users worldwide.

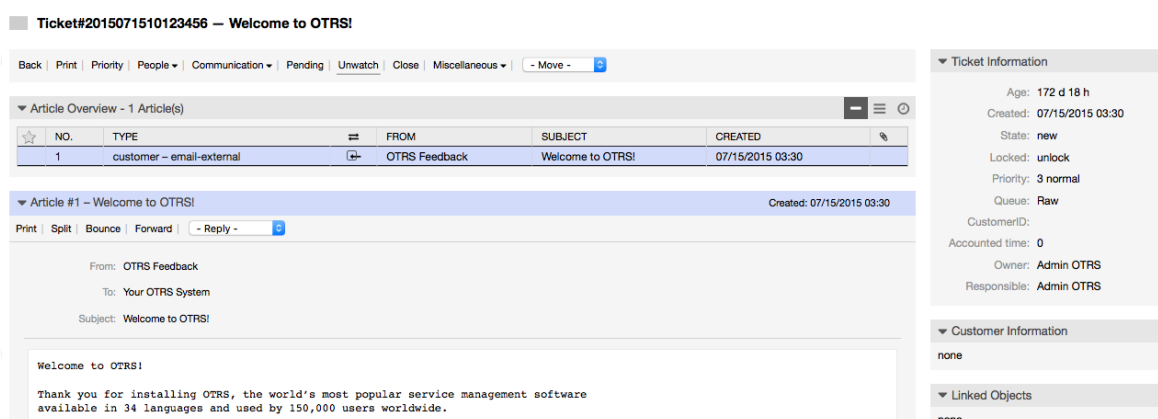
**Ticket Information**  
 Age: 172 d 18 h  
 Created: 07/15/2015 03:30  
 State: new  
 Locked: unlock  
 Priority: 3 normal  
 Queue: Raw  
 CustomerID:  
 Accounted time: 0  
 Owner: Admin OTRS  
 Responsible: Admin OTRS

**Customer Information**  
 none

**Linked Objects**  
 none

如果你不再关注一个特定的工单，进入它的详情窗口并点击菜单中的“取消关注”链接（见下图）。

**图 4.70.** 在一个工单的详情窗口取消关注



**Ticket#2015071510123456 – Welcome to OTRS!**

Back | Print | Priority | People | Communication | Pending | **Unwatch** | Close | Miscellaneous | - Move -

Article Overview - 1 Article(s)

NO.	TYPE	FROM	SUBJECT	CREATED
1	customer – email-external	OTRS Feedback	Welcome to OTRS!	07/15/2015 03:30

Article #1 – Welcome to OTRS! Created: 07/15/2015 03:30

Print | Split | Bounce | Forward | - Reply -

From: OTRS Feedback  
 To: Your OTRS System  
 Subject: Welcome to OTRS!

Welcome to OTRS!  
 Thank you for installing OTRS, the world's most popular service management software available in 34 languages and used by 150,000 users worldwide.

**Ticket Information**  
 Age: 172 d 18 h  
 Created: 07/15/2015 03:30  
 State: new  
 Locked: unlock  
 Priority: 3 normal  
 Queue: Raw  
 CustomerID:  
 Accounted time: 0  
 Owner: Admin OTRS  
 Responsible: Admin OTRS

**Customer Information**  
 none

**Linked Objects**  
 none

一旦激活了工单关注人功能，就可以通过服务人员界面仪表盘上各个小部件的“我关注的工单”视图查看一个服务人员关注的工单列表。

**图 4.71.** “我关注的工单”视图



**My Watched Tickets: All**

All 1 | New Article 0 | Pending 0 | Reminder Reached 0

Bulk 1-1 of 1 [S] [M] [L]

TICKET#	AGE	FROM / SUBJECT	STATE	LOCK	QUEUE	OWNER	CUSTOMERID
2015071510123456	172 d 18 h	OTRS Feedback Welcome to OTRS!	new	unlock	Raw	Admin OTRS	

## 7. 时间相关的功能

### 7.1. 设置上班时间和、假期和时区

OTRS中的一些功能如升级和工单自动解锁，取决于适当的上班时间和、假期和时区配置。你可以通过系统配置界面的Framework > Core::Time来定义这些设置。还可以指定不同的上班时间和、假期和时区集合作为单独的‘日历’，从Framework > Core::Time::Calendar1到Framework > Core::Time::Calendar9。可以通过队列设置来选择日历，或根据SLA级别来选择日历。举例来



说，这意味着你可以为‘标准’SLA指定一个5X8上班时间的日历，为‘金牌’SLA创建一个7X24支持的单独日历；同样可以为‘支持-美国’队列设置一个日历，其时间窗口与‘支持-日本’队列的日历的时间窗口不同。OTRS可以处理多达99个不同的日历。

### 7.1.1. 上班时间

在系统配置Framework > Core::Time::TimeWorkingHours中设置上班时间，或在日历配置部分你特定的日历中设置。OTRS能处理的时间粒度是1小时，复选框上的数字标识 8、9、10.....18对应的上班时间是：上午8点-下午6点。

只有在上班时间内才会升级工单、发送升级和挂起工单的通知和解锁工单。

### 7.1.2. 固定日期的假日

每年固定日期的假日，如元旦节、国庆节等，可以在TimeVacationDays中或日历1-9中相应部分中指定。

在TimeVacationDays中定义的日期内不会升级或解锁工单。

#### 注意

OTRS默认安装使用的是德国假日。

### 7.1.3. 浮动假日

每年没有固定日期的假日如春节、重阳节等，可以在TimeVacationDaysOneTime中指定。

在TimeVacationDaysOneTime中定义的日期内不会升级或解锁工单。

#### 注意

OTRS没有预置一次性假日，这意味着你需要添加这些节日如春节、重阳节等到OTRS系统配置中。

## 7.2. 自动解锁

锁定的工单可以由系统自动解锁。这个功能可能很有用，举例来说，如果一个服务人员锁定了一些需要处理的工单，但由于某些原因而不能处理它们，比如因紧急情况不在办公室。自动解锁功能会在一个给定的时间内解锁工单，确保没有锁定的工单被遗忘，从而允许其它服务人员可以处理它们。

在队列设置中指定一个工单被解锁前的时间值。命令bin/otrs.Console.pl Maint::Ticket::Unlock作为一个cron任务定期执行，完成工单的自动解锁。

解锁工单的通知只发送给“我的队列”中包含了解锁工单所属队列且在个人偏好设置中激了解锁工单通知的服务人员。

如果满足了下列所有条件，就会解锁工单：

- 在工单所在队列定义有一个解锁超时。
- 工单被设置为锁定。
- 工单状态是处理中。

如果一个服务人员添加了一封新的外部信件到工单，解锁计时器会被重置。可以是下列任意类型的信件：邮件-外部、电话、传真、短信或备注-外部。

此外，如果工单中最近的信件由一个服务人员创建，并且一个客户联系人添加了另外一封信件（不管是通过WEB还是邮件响应），解锁计时器都将被重置。

最后一个会重置解锁计时器的事件是当工单被分配给另外的服务人员时。

## 8. 定制PDF输出

这部分处理OTRS中PDF输出相关的可配置选项。

如果你在OTRS界面的任何地方使用打印操作，都将生成一个格式化的PDF文件。

通过创建你自己的LOGO文件并修改PDF::LogoFile参数，你可以调节OTRS生成的PDF文件的外观。使用PDF::PageSize参数，可以定义PDF文件的标准页面尺寸（DIN-A4或信笺）。还可以通过PDF::MaxPage指定一个PDF文件的最大页数，这可以防止一个用户因误操作生成一个巨大的PDF文件。

## 9. 统计

OTRS统计模块支持跟踪操作的统计和生成定制的OTRS使用情况的报告的功能。OTRS系统使用“统计”术语一般指一个呈现各种指标的报表。

### 注意

对**OTRS Business Solution™**（OTRS商业版）的客户，还有一个专用的报表生成器可用。这里的“报表”指在一个PDF文档中数个统计的集合，可以轻松配置并自动生成和分发PDF文档。请在**OTRS Business Solution™**（OTRS商业版）手册中查找更详细的信息。

OTRS统计模块的合理配置与大量的需求和考量相关，包括评估多种OTRS模块、用户权限设置、计算指标及其复杂性级别、统计模块配置的易用性、计算的速度和效率以及支持一组丰富的输出变量。

统计的元素如文件（增补了统计模块特定需求的功能）可以集成来计算复杂的统计数据。

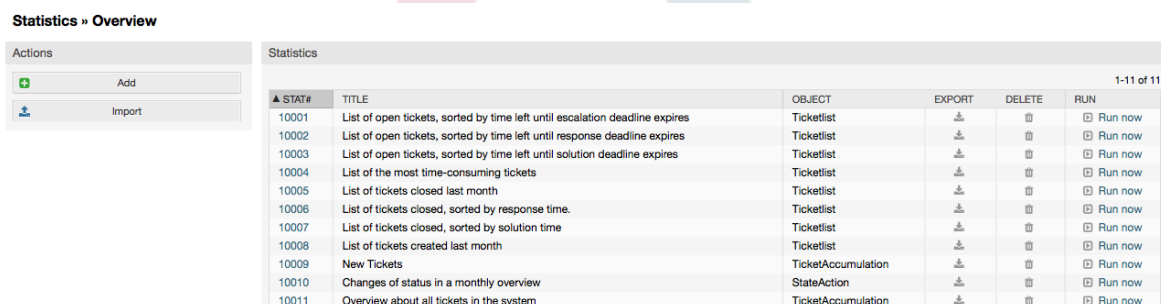
### 9.1. 统计配置和用法

作为服务人员登录时，可以通过选择主菜单中的“报表”和“统计”来打开统计模块。

#### 9.1.1. 概览

从导航栏选择“统计”然后点击子菜单“概览”可以调出概览窗口。概览窗口提供了该服务人员能够使用的预配置报表的列表（见下图）。

图 4.72. 标准统计概览



STAT#	TITLE	OBJECT	EXPORT	DELETE	RUN
10001	List of open tickets, sorted by time left until escalation deadline expires	Ticketlist	⬇	🗑	🏃 Run now
10002	List of open tickets, sorted by time left until response deadline expires	Ticketlist	⬇	🗑	🏃 Run now
10003	List of open tickets, sorted by time left until solution deadline expires	Ticketlist	⬇	🗑	🏃 Run now
10004	List of the most time-consuming tickets	Ticketlist	⬇	🗑	🏃 Run now
10005	List of tickets closed last month	Ticketlist	⬇	🗑	🏃 Run now
10006	List of tickets closed, sorted by response time.	Ticketlist	⬇	🗑	🏃 Run now
10007	List of tickets closed, sorted by solution time	Ticketlist	⬇	🗑	🏃 Run now
10008	List of tickets created last month	Ticketlist	⬇	🗑	🏃 Run now
10009	New Tickets	TicketAccumulation	⬇	🗑	🏃 Run now
10010	Changes of status in a monthly overview	StateAction	⬇	🗑	🏃 Run now
10011	Overview about all tickets in the system	TicketAccumulation	⬇	🗑	🏃 Run now

统计模块安装后，它预先加载了导入到系统中的一些示例统计，这些统计在概览窗口显示为一个列表。如果单个列表扩展为多面显示的列表，服务人员可以在不同的页面切换浏览。点击列表上期望排序的列标题，统计列表可以按期望排序。要生成一个特定的统计，在这个统计上点击“立即运行”链接即可。

## 9.1.2. 生成统计

查看统计界面提供了统计的配置设置（见下图）。

图 4.73. 查看一个特定的统计。

Statistics » View Stat#10001 – List of open tickets, sorted by time left until escalation deadline expires

Actions	Settings
<input type="button" value="Go to overview"/> <input type="button" value="Edit"/>	Object: Ticketlist Description: List of open tickets, sorted by time left until escalation deadline expires. NOTE: Please check the output and configuration of the statistics carefully to make sure that it produces the results you expect. If necessary, change the configuration before using the statistics in a production environment. Format: <input type="text" value="CSV"/> <b>X-axis</b> Attributes to be printed: Number, Ticket#, Age, Title, Created, Changed, Close Time, Queue, State, Priority, Customer User, CustomerID, Service... <b>Y-axis</b> Order by: EscalationTimeWorkingTime Sort sequence: ascending <b>Filter</b> State: new, open, pending auto close+, pending auto close-, pending reminder <input type="button" value="Run now"/> or <input type="button" value="Cancel"/>
<b>Statistic Information</b> Created: 12/14/2015 09:32:45 Created by: test11450081960626627533 test11450081960626627533 Changed: 12/14/2015 09:32:45 Changed by: test11450081960626627533 test11450081960626627533 Sum rows: No Sum columns: No Show as: No dashboard widget: Cache: No Validity: valid	

可以在编辑窗口设置一个特定统计有限的配置选项。统计创建者或其它有合适权限的人员可以编辑这些设置。

点击“立即运行”按钮（在窗口左侧）可以生成这个统计。




## 9.1.3. 配置

有写入权限的服务人员可以在统计模块的编辑界面编辑一个已有的报表配置。另外，他们也可以创建一个新的报表。

配置一个统计有4个可能的步骤：一般预设数据、配置X轴、配置Y轴和可能的报表数据过滤器（或限制）。让我们创建一个新的示例统计，在概览窗口点击“添加”按钮。我们的目标是获取每个队列（X轴）和每个状态（Y轴）有多少工单优先级是非常高的列表。

图 4.74. 添加一个新的统计，第一步。

Add New Statistic

 <p><b>Dynamic Matrix</b> Tabular reporting data where each cell contains a singular data point (e. g. the number of tickets).</p>	 <p><b>Dynamic List</b> Tabular reporting data where each row contains data of one entity (e. g. a ticket).</p>	 <p><b>Static</b> Complex statistics that cannot be configured and may return non-tabular data.</p>
---	--	--

在开始页面我们必须选择要添加的统计的类型。有3个可用的类型：

动态矩阵统计

这个类型的统计会生成一个计算值的矩阵（例如：月度和队列每天的新工单）。矩阵中所有的单元格值类型相同（数字、平均时间等等）。值通过系统中的条目（例如：工单）计算而来。一些矩阵统计支持一个汇总列和/或汇总行（只对某些数据有用）。

## 动态列表统计

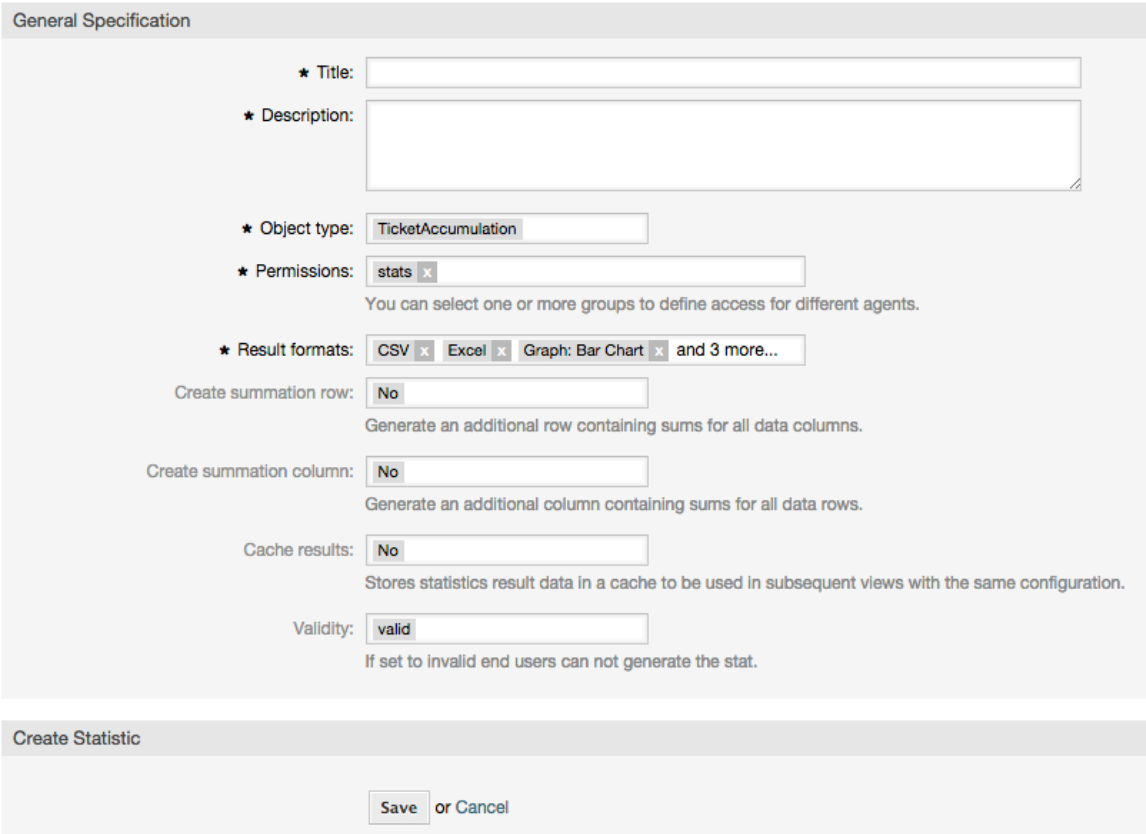
这种统计会生成一个表格，表格的每一行（不是单元格）代表系统中的一个实体（例如：一个工单），这一行的每列通常是可以配置的（X轴，见下）并包含这个对象的数据（例如：工单属性）。一列的所有单元格值的类型相同。

## 静态统计

这种统计很少配置，通常用于非常特殊和/或复杂的计算。

所以在本示例中让我们选择“动态矩阵”。然后在统计类型下会出现“一般设定”的配置。

**图 4.75.** 添加一个新的统计，第二步。



**General Specification**

★ Title:

★ Description:

★ Object type:

★ Permissions:

You can select one or more groups to define access for different agents.

★ Result formats:

Create summation row:

Generate an additional row containing sums for all data columns.

Create summation column:

Generate an additional column containing sums for all data rows.

Cache results:

Stores statistics result data in a cache to be used in subsequent views with the same configuration.

Validity:

If set to invalid end users can not generate the stat.

---

**Create Statistic**

or

提供新统计的标题和描述后，我们必须选择要使用的统计后端。这是负责收集和分析统计数据实际后端模块。在我们的例子中我们选择“TicketAccumulation（累积工单）”。

通过配置权限组，我们能够促进谁可以在后面查看和生成预配置的统计的组限制（也就是服务人员）。所以可以给需要统计的不同部门和工作组分配多个不同的统计。也可以将一个统计分配给多个组。

### 例 4.20. 默认统计权限组

选择“stats”组，对“stats”组至少有‘只读’权限的用户均可查看这个报表。这个访问权限是默认可用的。

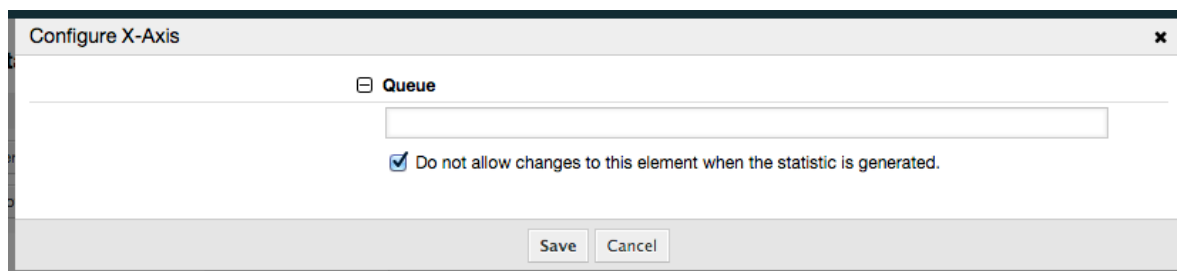
### 例 4.21. 定制统计权限组

选择一个名为“sales”的组，对“sales”组有‘只读’权限的用户均可查看并生成这个报表。但是其它用户无法查看这个报表。

此外，还可以选择输出格式。在这里我们只需保持所有输出格式并在实际生成统计时再选择一个输出格式。现在让我们保存这个统计。

“保存”统计后会出现下一个窗口：配置X轴。点击“X轴”按钮，出现一个对话框，我们可以选择用于X轴的元素。在我们的例子中就是‘队列’：

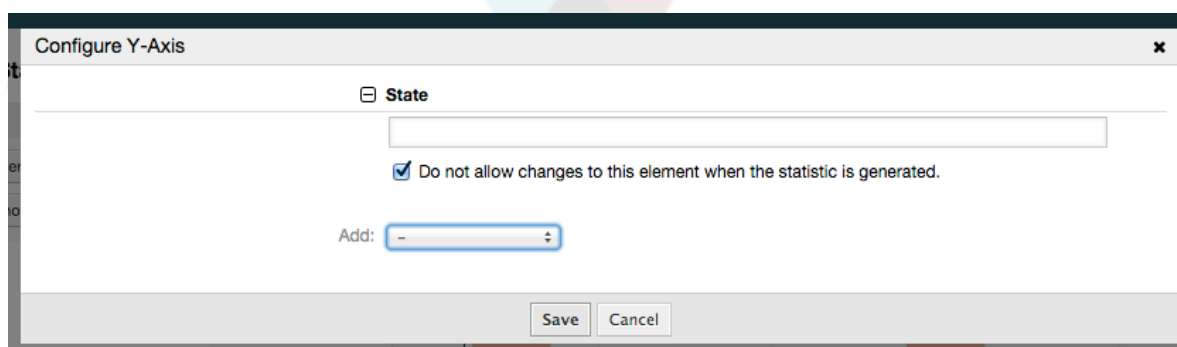
图 4.76. 配置一个统计的X轴。



通过在队列字段选择一些队列，我们可以随意地限制要显示的队列。通过复选框我们能够控制服务人员生成统计时是否可以修改选择的队列。我们保持默认并点击“保存”按钮。

现在我们可以通过相同的方法配置Y轴：选择‘状态’字段

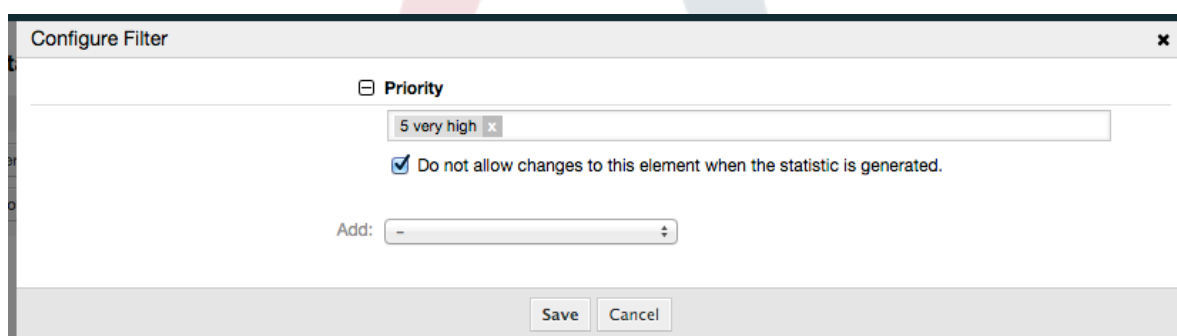
图 4.77. 配置一个统计的Y轴。



在这里可以选择一个或两个元素。一个元素时，元素的每个值就是Y轴的一个元素。如果选择了两个元素，它们的排列就是值系列（Y轴）的元素。例如你可以选择“状态”和“优先级”，结果元素就是“新建-1-非常低”、“新建-2-低” ... “处理中-1-非常低”等等。让我们仅使用“状态”并点击“保存”。

现在最后一步，我们可以添加数据过滤器以便只统计属于某个客户某个优先级等条件的工单。我们添加一个‘非常高’优先级的工单过滤器：

图 4.78. 配置一个统计的数据过滤器。



现在再次点击“保存”按钮，配置就完成了。

你可以已经注意到了，在配置对话框有一个统计预览区域，我们可以在这里检查配置设置的效果。

图 4.79. 配置一个统计的数据过滤器。



## 注意

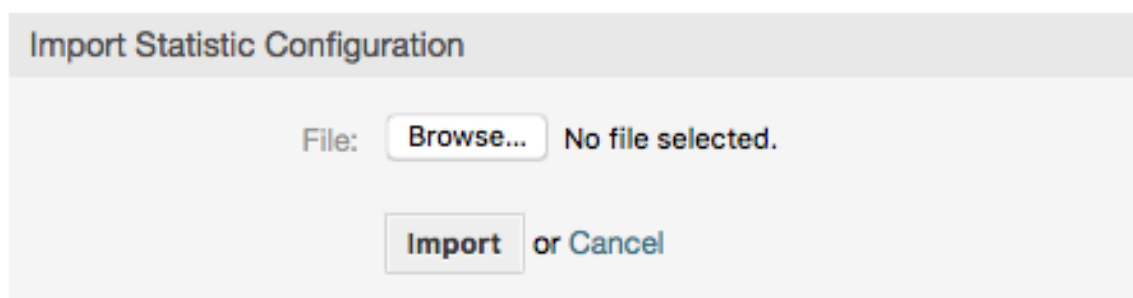
请注意预览使用的是随机数据且没有考虑数据过滤器。

统计已经配置好了，点击“立即运行”按钮，我们能够进入查看窗口选择输出格式，统计可以用不同格式生成。

### 9.1.4. 导入

可以通过点击统计概览窗口的“导入”按钮访问导入界面，需要统计模块的“读写”权限。

图 4.80. 导入统计



促进统计模块的报表导入与导出功能结合使用，是一个非常方便的功能。可以在测试系统中方便地创建和测试统计，然后导入到生产系统。

## 9.2. 统计系统管理员

本部分提供系统管理员处理统计模块的任务和职责的信息。

## 9.2.1. 权限设置、组和队列

统计模块注册的默认配置给所有服务人员有“stats”组权限来访问统计模块。

根据权限设置的访问：

- rw（读写），允许配置统计。
- ro（只读），允许生成预先配置的统计。

OTRS系统管理员决定是將有权限生成预先配置的报表服务人员分配“stats”组的‘只读’权限，还是將他们各自的组加入系统配置中的模块注册信息。

## 9.2.2. 系统配置设置

系统配置组Framework:Core::Stats和Framework:Frontend::Agent::Stats包含了所有的统计模块基本设置参数。此外，配置参数\$Self->{'Frontend::Module'}->{'AgentStats'}控制统计模块内模块和图标的布局和注册信息。

## 9.2.3. 用命令行生成统计

可以在命令行执行命令bin/otrs.Console.pl Maint::Stats::Generate来生成统计。举个例子，查看下面脚本中的命令行调用。

```
shell> bin/otrs.Console.pl Maint::Stats::Generate --number 10004 --target-directory /tmp
Generating statistic number 10004...
  Writing file /tmp/List_of_the_most_time-consuming_tickets_Created_2015-09-08_14-51.csv.
Done.
```

从统计配置“Stat#10004”生成报表并保存为一个CSV文件到/tmp目录。

生成的报表还可以作为邮件发送，可以执行下面脚本中的命令获取更多信息。

```
shell> bin/otrs.Console.pl Maint::Stats::Generate --help
```

通常通过命令行手动生成报表没有什么意义，因为统计模块有一个便利的图形界面。但是，结合一个cron任务来手动生成报表就有意义了。

想像下列场景：每月的第一天，部门领导要收到一份过去一月的报表。通过结合cron任务和命令行生成报表可以通过邮件自动将报表发送给他们。

# 10. 动态字段

## 10.1. 介绍

动态字段是OTRS中一种特殊种类的字段，创建来扩展信息，存储在工单或信件中。这些字段不是固定在系统中的，它们可能仅在特定的窗口中出现，它们可以是必填的或可选的，它们在窗口中的呈现方式取决于创建时根据该字段存储的数据定义的字段类型。例如，存储一行文本、一个日期、一个选择项等等的动态字段。

动态字段是由TicketFreeText（工单自定义文本）、TicketFreeKey（工单自定义键）、TicketFreeTime（工单自定义时间）、ArticleFreeText（信件自定义文本）和ArticleFreeKey（信件自定义键）字段（一般在OTRS 3.0之前中使用）演变而来。这些“自定义字段”的限制是：一个工单最多定义16个（文本或下拉）字段和6个时间字段，每个信件3个（文本或下拉）字段，不能更多。

现在使用动态字段后每个工单或信件的这种字段数量限制被移除了，你可以为工单或信件创建想要的任意数量的动态字段。除此之外，动态字段背后的框架已经准备好处理其它对象的定制字段了，而不仅仅是工单和信件。

这个新的处理动态字段的框架使用了模块化方法来构建，每种动态字段可以被看作是这个框架的插件模块。这意味着通过公共的OTRS模块、OTRS功能插件、OTRS定制开发和其它定制开发，可以轻易扩展大量的动态字段。

这个版本包括了下列动态字段类型：

- 文本（一行文本）
- 多行文本（多行文本）
- 复选框
- 下拉列表框（单个选择，多个值）
- 多选框（多个选择，多个值）
- 日期
- 日期 / 时间

## 10.2. 配置

By default, a clean installation of OTRS comes with two pre-installed internal dynamic fields. If you plan to use such fields in tickets or articles you need to create new, custom dynamic fields.

动态字段的配置分为两个部分，通过系统管理页面的“动态字段”链接添加一个新的动态字段或管理已有的动态字段。要在一个窗口中显示、强制显示或隐藏一个动态字段，你需要在系统配置窗口中修改OTRS设置。

### 10.2.1. 新增一个动态字段

点击导航栏中的“系统管理”按钮，然后点击“动态字段”链接（位于窗口中下的工单设置框中），就会显示动态字段概览，如下图所示：

**图 4.81. Dynamic fields overview screen.**

**Dynamic Fields Management - Overview**

**Actions**

**Ticket**

  
Add new field for object: Ticket

**Article**

  
Add new field for object: Article

**Hint**

To add a new field, select the field type from one of the object's list, the object defines the boundary of the field and it can't be changed after the field creation.

**Dynamic Fields List** 1-2 of 2

NAME	LABEL	ORDER	TYPE	OBJECT	VALIDITY	DELETE
ProcessManagementProcessID	Process	1	ProcessID	Ticket	valid	
ProcessManagementActivityID	Activity	2	ActivityID	Ticket	valid	

注意：这个窗口的内容会随着你添加更多的动态字段而改变，如果系统是从老版本的OTRS升级而来可能已经有一些动态字段。

窗口左侧的操作栏中有两个操作对象：工单和信件，每个都有自己的动态字段下拉列表框。

### 注意

随着安装的OTRS软件包还会有更多的对象出现在操作栏中。

创建一个动态字段的一般流程是：

- 在操作栏点击期望的动态字段下拉列表框。



- 点击你想从列表中添加的动态字段类型。
- 填写相关配置信息。
- 保存。

动态字段配置对话框分为两个部分，上部分是所有动态字段的通用设置，下部分则根据动态字段类型可能不同。

动态字段常规设置：

- 名称：强制、唯一，只允许字母和数字。

这是动态字段的内部名称，用于在一个窗口中显示或隐藏一个动态字段。对一个动态字段名称的任何修改（不推荐）都需要手动更新使用了该字段的系统配置。

- 标签：强制。

这是显示在窗口中的动态字段名称，它可以翻译。

### 注意

标签的翻译必须手动添加到语言翻译文件中。

- 字段顺序：强制。

定义动态字段显示在窗口的相对顺序，每个新字段默认都在最后的位置，这个设置会影响其它已创建的动态字段顺序。

- 有效性：强制。

一个无效的动态字段不能在任何窗口中显示，不管它是否配置为显示。

- 字段类型：强制，只读。

显示当前选择的动态字段类型。

- 对象类型：强制，只读。

显示动态字段所属的模块范围。

### 注意

为了描述每个特定的动态字段类型的设置，我们的例子中包含了几个不同的字段。这些新字段在后面部分详细描述。

下面的例子创建的所有动态字段的类型属于工单对象。如果你需要创建一个信件对象的动态字段，只需从信件下拉列表中选择你要的动态字段类型即可。

**表 4.6.** 下列字段将被添加到系统中：

名称	标签	类型
Field1	My Field 1	文本
Field2	My Field 2	文本块
Field3	My Field 3	复选框
Field4	My Field 4	下拉选项框
Field5	My Field 5	多选框
Field6	My Field 6	日期
Field7	My Field 7	日期 / 时间

## 10.2.2. 文本动态字段配置

文本动态字段用于存储一个单行字符串。

文本动态字段设置：

- 默认值：可选。

这是在编辑窗口（如新建电话工单或工单编写）显示的默认值。

- 显示链接：可选。

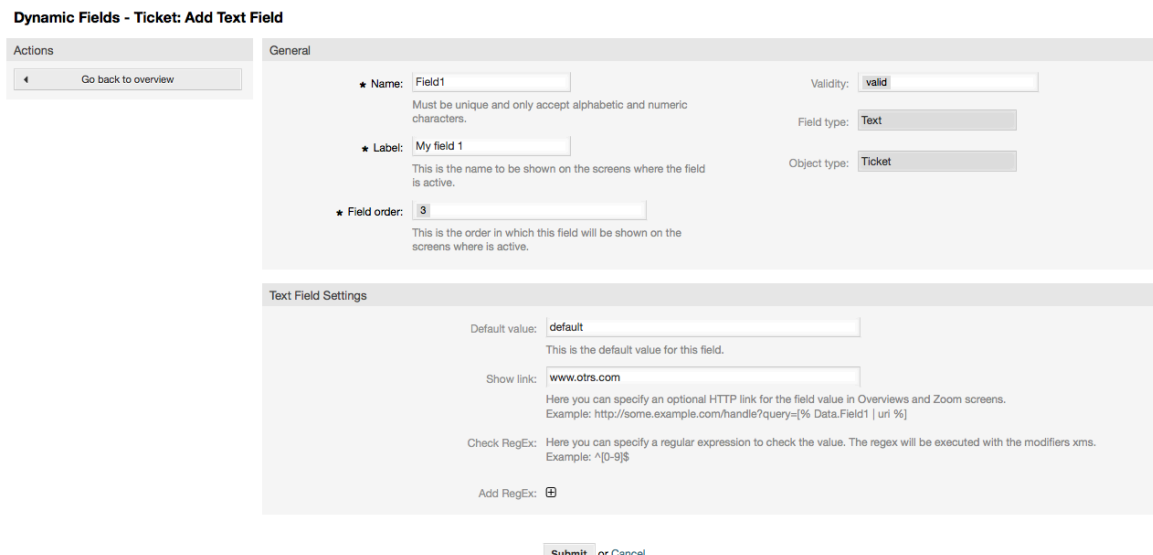
如果设置了，这个字段的值会在显示窗口（如工单详情窗口或工单概览窗口）被转化成一個可点击的链接。

例如，如果“显示链接”设置为“http://www.otrs.com”，点击这个字段填充的值会让你的浏览器打开OTRS网站。

### 注意

在设置链接的值中使用[% Data.NameX | uri %]（NameX是字段的名称），会将字段的值添加到链接引用中作为其一部分。

图 4.82. 文本动态字段配置对话框



**Dynamic Fields - Ticket: Add Text Field**

**Actions**  
 Go back to overview

**General**

- ★ Name: Field1  
Must be unique and only accept alphabetic and numeric characters.
- Validity: valid
- Field type: Text
- Object type: Ticket
- ★ Label: My field 1  
This is the name to be shown on the screens where the field is active.
- ★ Field order: 3  
This is the order in which this field will be shown on the screens where is active.

**Text Field Settings**

- Default value: default  
This is the default value for this field.
- Show link: www.otrs.com  
Here you can specify an optional HTTP link for the field value in Overviews and Zoom screens.  
 Example: http://some.example.com/handle?query={% Data.Field1 | uri %}
- Check RegEx:   
Here you can specify a regular expression to check the value. The regex will be executed with the modifiers xms.  
 Example: ^[0-9]\$
- Add RegEx:

Submit or Cancel

## 10.2.3. 多行文本动态字段配置

多行文本动态字段用于存储一个多行的字符串。

多行文本动态字段设置：

- 行数：可选，整数。

用于定义该字段在编辑窗口（如新建电话工单或工单编写）高度。

- 列数：可选，整数。

这个值用于定义该字段在编辑窗口中的宽度。

- 默认值：可选。

这是在编辑窗口显示的默认值（可以是多行文本）。

图 4.83. 多行文本动态字段配置对话框

**Dynamic Fields - Ticket: Change Textarea Field**

Actions

Go back to overview

General

★ Name:  Validity:   
Must be unique and only accept alphabetic and numeric characters.

★ Label:  Field type:   
This is the name to be shown on the screens where the field is active.

★ Field order:  Object type:   
This is the order in which this field will be shown on the screens where is active.

---

Textarea Field Settings

Number of rows:   
Specify the height (in lines) for this field in the edit mode.

Number of cols:   
Specify the width (in characters) for this field in the edit mode.

Default value:   
  
This is the default value for this field.

Check RegEx:   
Here you can specify a regular expression to check the value. The regex will be executed with the modifiers xms.  
 Example: ^[0-9]\$

Add RegEx:

Submit or Cancel

## 10.2.4. 复选框动态字段配置

复选框动态字段用于存储‘真’或‘假’值，代表一个选中或未选中的复选框。

复选框动态字段设置：

- 默认值：强制。

这是在编辑窗口（如新建电话工单或工单编写）显示的默认值，该字段的默认值只能是选中或未选中。

图 4.84. 复选框动态字段配置对话框

**Dynamic Fields - Ticket: Add Checkbox Field**

Actions

Go back to overview

General

★ Name:  Validity:   
Must be unique and only accept alphabetic and numeric characters.

★ Label:  Field type:   
This is the name to be shown on the screens where the field is active.

★ Field order:  Object type:   
This is the order in which this field will be shown on the screens where is active.

---

Checkbox Field Settings

Default value:   
This is the default value for this field.

Submit or Cancel

## 10.2.5. 下拉列表框动态字段配置

下拉列表框动态字段

下拉列表框动态字段设置：

- 可能的值：强制。

选择值的列表。如果使用了，一个新值需要指定键（内部值）和值（显示值）。

- 默认值：可选。

这是在编辑窗口（如新建电话工单或工单编写）显示的默认值，默认值只能从定义的可能值中选择。

- 添加empty值：强制，（是/否）。

如果激活了这个选项，在可能的值之外会定义一个额外的值，显示为“-”，这个特殊值在内部是empty。

- 可翻译的值：强制，（是/否）。

这个设置用来标记这个动态字段可能的值是否可以翻译。只有显示的值才能翻译，内部值不受影响，值的翻译需要手动添加到语言文件。

- 显示链接：可选。

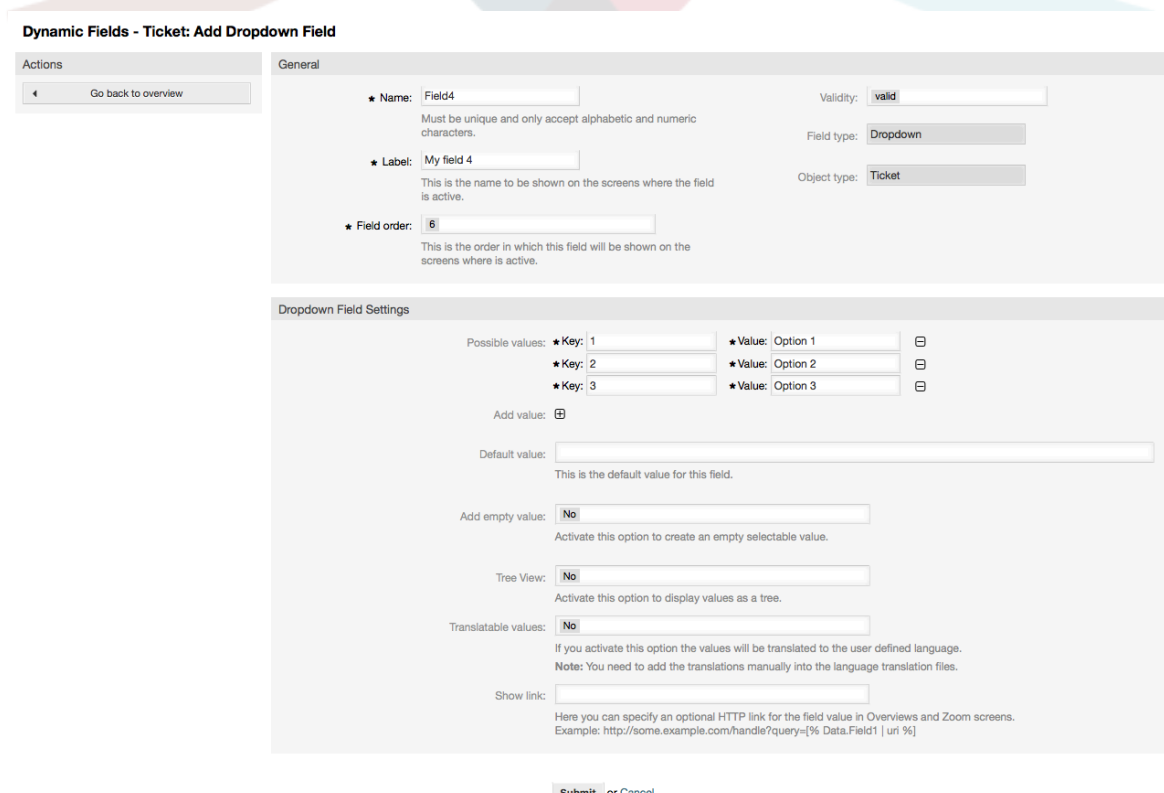
如果设置了，这个字段的值会在显示窗口（如工单详情窗口或工单概览窗口）被转化成一个可点击的链接。

例如，如果“显示链接”设置为“http://www.otrs.com”，点击这个字段填充的值会让你的浏览器打开OTRS网站。

## 注意

在设置链接的值中使用[% Data.NameX | uri %]（NameX是字段的名称），会将字段的值添加到链接引用中作为其一部分。

图 4.85. 下拉列表框动态字段配置对话框



**Dynamic Fields - Ticket: Add Dropdown Field**

**Actions**

Go back to overview

**General**

★ Name:  Validity:

Must be unique and only accept alphabetic and numeric characters.

Field type:

★ Label:  Object type:

This is the name to be shown on the screens where the field is active.

★ Field order:

This is the order in which this field will be shown on the screens where is active.

**Dropdown Field Settings**

Possible values:

★ Key:	★ Value:	
<input type="text" value="1"/>	<input type="text" value="Option 1"/>	<input type="checkbox"/>
<input type="text" value="2"/>	<input type="text" value="Option 2"/>	<input type="checkbox"/>
<input type="text" value="3"/>	<input type="text" value="Option 3"/>	<input type="checkbox"/>

Add value:

Default value:

This is the default value for this field.

Add empty value:

Activate this option to create an empty selectable value.

Tree View:

Activate this option to display values as a tree.

Translatable values:

If you activate this option the values will be translated to the user defined language.  
Note: You need to add the translations manually into the language translation files.

Show link:

Here you can specify an optional HTTP link for the field value in Overviews and Zoom screens.  
Example: http://some.example.com/handle?query=[% Data.Field1 | uri %]

or

## 10.2.6. 多选动态字段配置

多选动态字段用来存储一个固定列表中的多个值。

多选动态字段设置：

- 可能的值：强制。

从中选择值的列表。当添加附件的列表项目时，需要指定键（内部值）和值（显示值）

- 默认值：可选。

这是在编辑窗口（如新建电话工单或工单编写）显示的默认值，默认值只能从定义的可能值中选择。

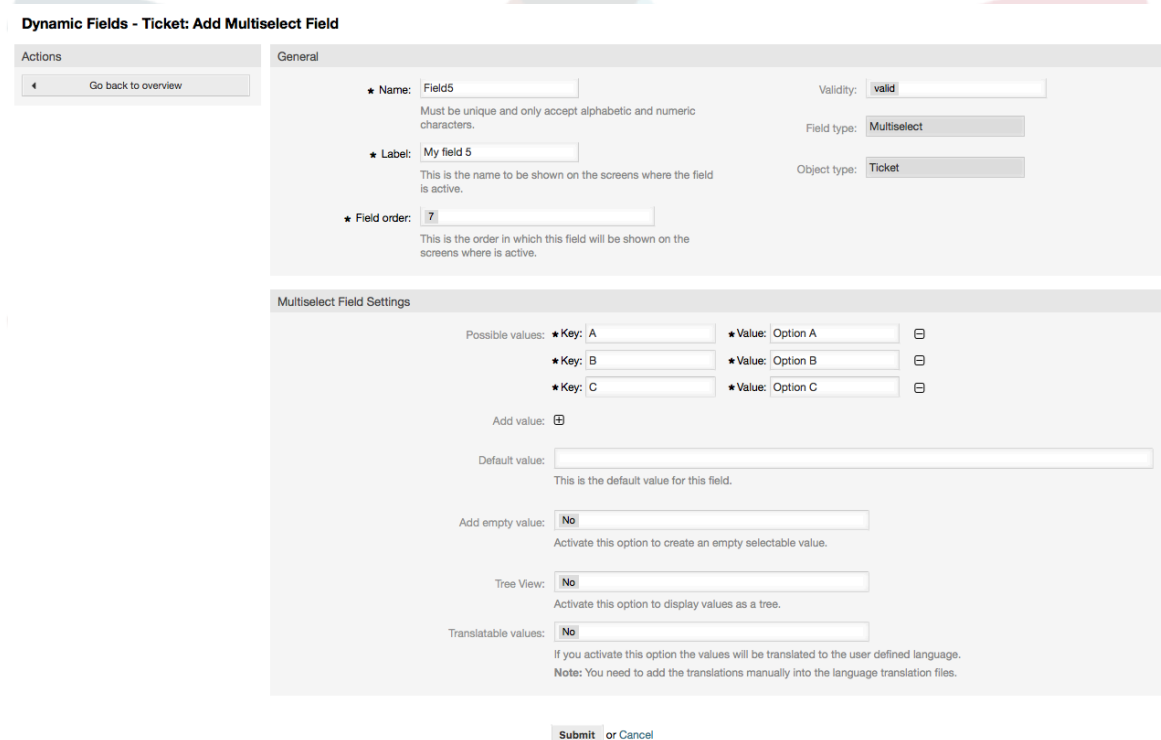
- 添加empty值：强制，（是/否）。

如果激活了这个选项，在可能的值之外会定义一个额外的值，显示为“-”，这个特殊值在内部是empty。

- 可翻译的值：强制，（是/否）。

这个设置用来标记这个动态字段可能的值是否可以翻译。只有显示的值才能翻译，内部值不受影响，值的翻译需要手动添加到语言文件。

图 4.86. 多选动态字段配置对话框



**Dynamic Fields - Ticket: Add Multiselect Field**

**Actions**

Go back to overview

**General**

★ Name:  Validity:

Must be unique and only accept alphabetic and numeric characters.

Field type:

★ Label:  Object type:

This is the name to be shown on the screens where the field is active.

★ Field order:  Object type:

This is the order in which this field will be shown on the screens where is active.

**Multiselect Field Settings**

Possible values: ★ Key:  ★ Value:

★ Key:  ★ Value:

★ Key:  ★ Value:

Add value:

Default value:

This is the default value for this field.

Add empty value:

Activate this option to create an empty selectable value.

Tree View:

Activate this option to display values as a tree.

Translatable values:

If you activate this option the values will be translated to the user defined language.  
Note: You need to add the translations manually into the language translation files.

or

## 10.2.7. 日期动态字段配置

日期动态字段用来存储一个日期值（年、月、日）。

日期动态字段设置：

- 默认日期差：可选，整数

在编辑窗口（如新建电话工单或工单编写）默认显示的时间与当前时间之间相差的秒数（正或负）。

- 定义年份周期：强制，（是/否）。

用来设置在选择这个字段的年份时，是否限定可以选择使用的年份。如果设置为‘是’，则下列选项可用：

- 过去的几年：可选，正整数。

在编辑窗口选择这个动态字段的年份时，能选择从当天算过去的几年。

- 未来的几年：可选，正整数。

在编辑窗口选择这个动态字段的年份时，能选择从当天算未来的几年。

- 显示链接：可选。

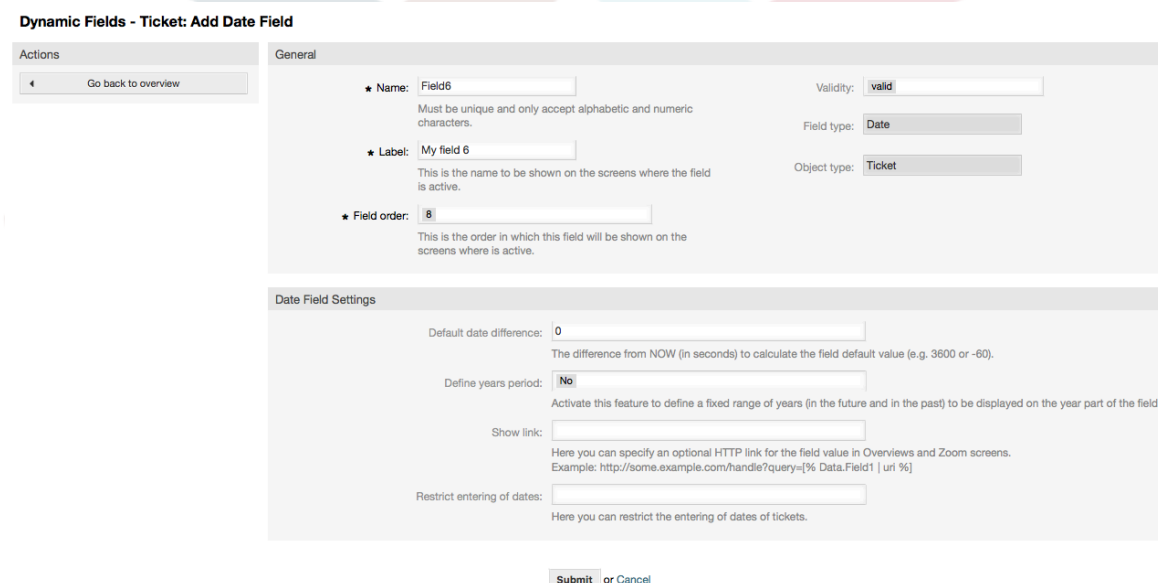
如果设置了，这个字段的值会在显示窗口（如工单详情窗口或工单概览窗口）被转化成可点击的链接。

例如，如果‘显示链接’设置为“http://www.otrs.com”，点击这个字段填充的值会让你的浏览器打开OTRS网站。

## 注意

在设置链接的值中使用[% Data.NameX | uri %]（NameX是字段的名称），会将字段的值添加到链接引用中作为其一部分。

图 4.87. 日期动态字段配置对话框



**Dynamic Fields - Ticket: Add Date Field**

**Actions**

Go back to overview

**General**

★ Name:  Validity:   
Must be unique and only accept alphabetic and numeric characters.

★ Label:  Field type:   
This is the name to be shown on the screens where the field is active.

★ Field order:  Object type:   
This is the order in which this field will be shown on the screens where is active.

**Date Field Settings**

Default date difference:   
The difference from NOW (in seconds) to calculate the field default value (e.g. 3600 or -60).

Define years period:   
Activate this feature to define a fixed range of years (in the future and in the past) to be displayed on the year part of the field.

Show link:   
Here you can specify an optional HTTP link for the field value in Overviews and Zoom screens.  
 Example: http://some.example.com/handle?query=[% Data.Field1 | uri %]

Restrict entering of dates:   
Here you can restrict the entering of dates of tickets.

or

## 10.2.8. 日期/时间动态字段配置

日期/时间动态字段用来存储一个日期时间值（年、月、日、时、分和秒）。

日期/时间动态字段设置：

- 默认日期差：可选，整数

在编辑窗口（如新建电话工单或工单编写）默认显示的时间与当前时间之间相差的秒数（正或负）。

- 定义年份周期：强制，（是/否）。

用来设置在选择这个字段的年份时，是否限定可以选择使用的年份。如果设置为‘是’，则下列选项可用：

- 过去的几年：可选，正整数。

在编辑窗口选择这个动态字段的年份时，能选择从当天算过去的几年。

- 未来的几年：可选，正整数。

在编辑窗口选择这个动态字段的年份时，能选择从当天算未来的几年。

- 显示链接：可选。

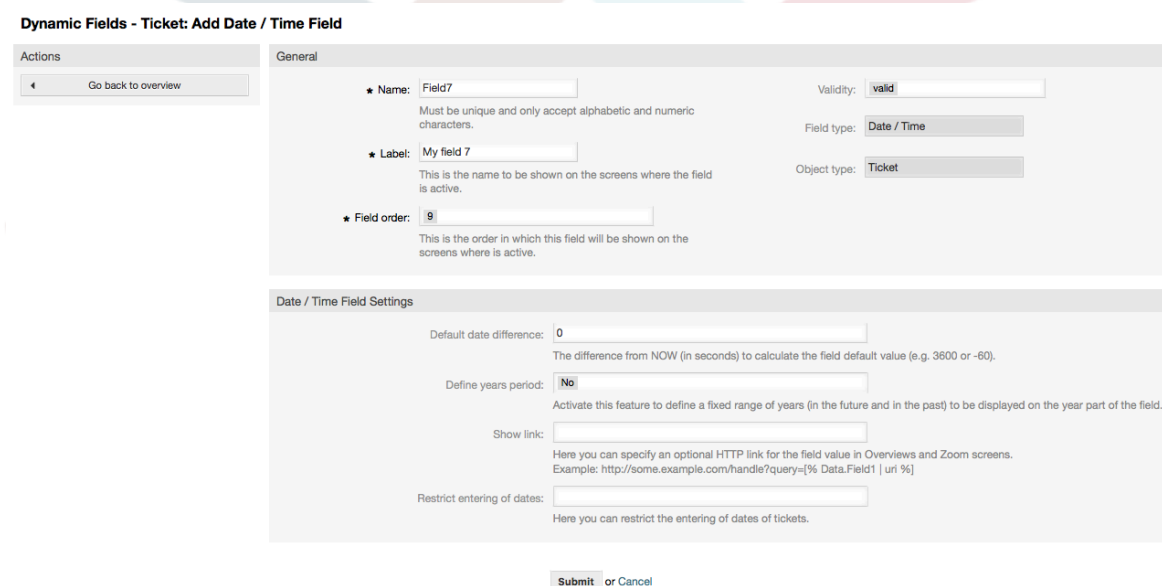
如果设置了，这个字段的值会在显示窗口（如工单详情窗口或工单概览窗口）被转化成成一个可点击的链接。

例如，如果‘显示链接’设置为“http://www.otrs.com”，点击这个字段填充的值会让你的浏览器打开OTRS网站。

## 注意

在设置链接的值中使用[% Data.NameX | uri %]（NameX是字段的名称），会将字段的值添加到链接引用中作为其一部分。

图 4.88. 日期/时间动态字段配置对话框



**Dynamic Fields - Ticket: Add Date / Time Field**

**Actions**

Go back to overview

**General**

★ Name:  Validity:   
Must be unique and only accept alphabetic and numeric characters.

★ Label:  Field type:   
This is the name to be shown on the screens where the field is active.

★ Field order:  Object type:   
This is the order in which this field will be shown on the screens where is active.

**Date / Time Field Settings**

Default date difference:   
The difference from NOW (in seconds) to calculate the field default value (e.g. 3600 or -60).

Define years period:   
Activate this feature to define a fixed range of years (in the future and in the past) to be displayed on the year part of the field.

Show link:   
Here you can specify an optional HTTP link for the field value in Overviews and Zoom screens.  
 Example: http://some.example.com/handle?query=% Data.Field1 | uri %

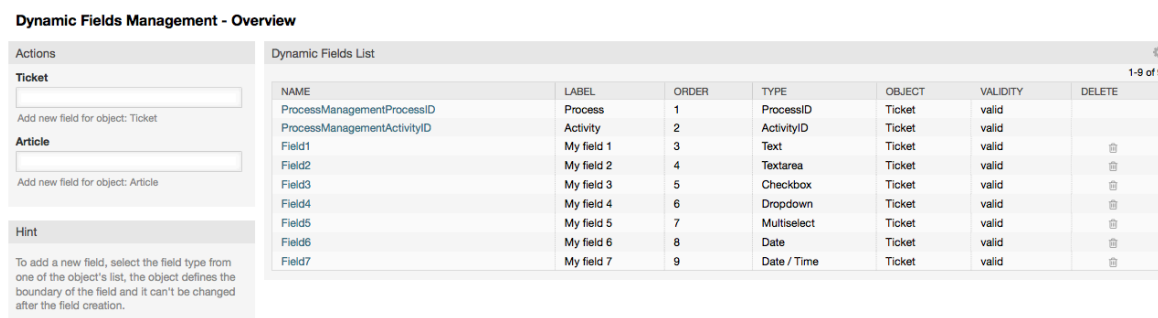
Restrict entering of dates:   
Here you can restrict the entering of dates of tickets.

Submit or Cancel

## 10.2.9. 编辑一个动态字段

一个（用前面的例子）填满动态字段概览窗口像这个样子：

图 4.89. 填满样例数据的动态字段概览窗口



要修改或编辑一个动态字段，你必须至少定义了一个动态字段，从动态字段概览窗口选择一个已经添加的字段然后更新它的设置。

## 注意

不是所有的动态字段设置都可以修改的，字段类型和对对象类型是从选择字段时就固定了的，它们都无法被修改。

不推荐去修改字段的内部名称，但可以随时修改标签。如果修改了内部名称，所有系统配置中使用了该字段的设置都需要更新，同时还有用户偏好设置（如果定义了该动态字段）。

## 10.2.10. 在一个窗口中显示一个动态字段

要在一个特定的窗口显示一个动态字段，有两个强制条件：

1. 动态字段必须有效。
2. 在这个窗口的配置中动态字段必须设置为1或2。

按照下列步骤在一个窗口中显示一个动态字段

- 确保动态字段设置为有效，你可以从动态字段概览窗口查看这个字段的有效性。如需要可以通过编辑该字段设置为有效。
- 打开系统配置并从窗口左侧操作栏的下拉列表框中选择“Ticket（工单）”。

## 注意

你还可以在上面的搜索框中搜索“DynamicField”或准确的系统配置项（如果你已经知道）。

- 定位你要查找的窗口设置所在的子组然后点击它。例如“Frontend::Agent::Ticket::ViewPhoneNew”。
- 搜索以“###DynamicField”结尾的设置，例如“Ticket::Frontend::AgentTicketPhone###DynamicField”。
- 如果这个设置为空或者没有需要的动态字段名称，点击“+”按钮来添加一个新的条目。例如，键：Field1，值：1。

如果这个设置已经列出了这个动态字段的名称，确保对应值设置为1-设置这个字段或2-显示为强制（必填）。

- 通过点击窗口正文的“更新”按钮来保存配置，然后导航到你要显示这个字段的窗口。



### 10.2.10.1. 显示动态字段的例子

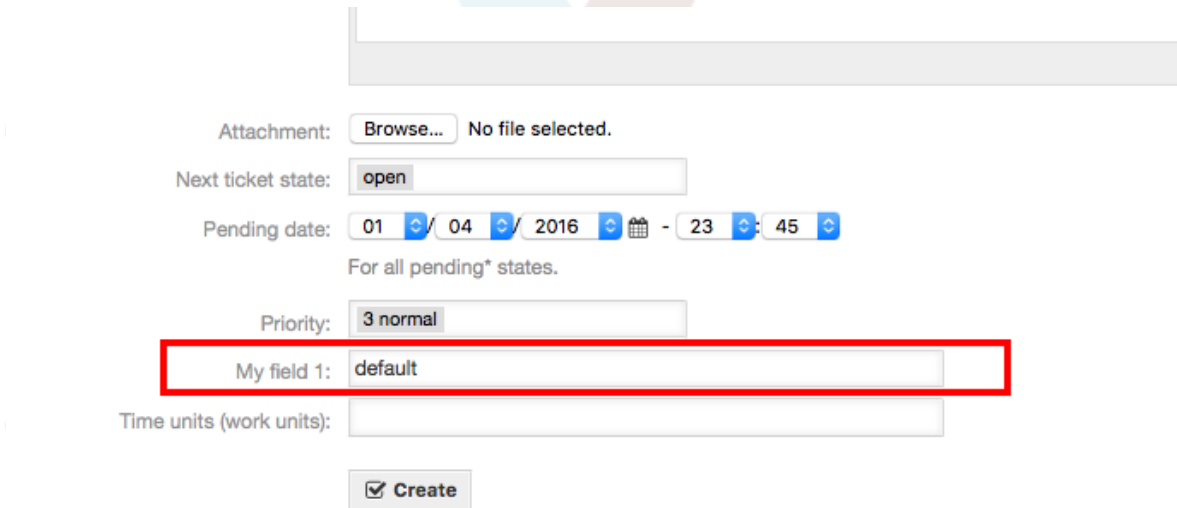
下面是在不同窗口中显示或隐藏动态字段的系统配置的配置示例。

例 **4.22.** 在新建电话工单窗口激活Field1。

- 主组 : Ticket
- 子组 : Frontend::Agent::Ticket::ViewPhoneNew
- 设置 : Ticket::Frontend::AgentTicketPhone###DynamicField
- 值 :

键	值
Field1	1

图 **4.90.** 在新建电话工单窗口显示Field1



Attachment:  No file selected.

Next ticket state:

Pending date:  /  /   :

For all pending\* states.

Priority:

**My field 1:**

Time units (work units):

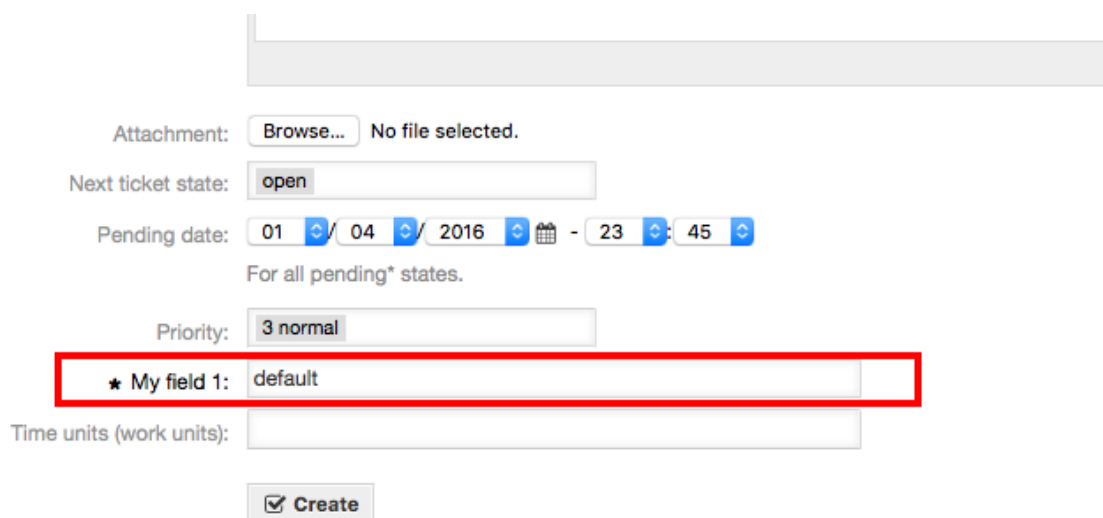
Create

例 **4.23.** 在新建电话工单窗口激活Field1为必填字段。

- 主组 : Ticket
- 子组 : Frontend::Agent::Ticket::ViewPhoneNew
- 设置 : Ticket::Frontend::AgentTicketPhone###DynamicField
- 值 :

键	值
Field1	2

图 4.91. 在新建电话工单窗口显示**Field1**为必填字段



Attachment:  No file selected.

Next ticket state:

Pending date:  /  /   :

For all pending\* states.

Priority:

**★ My field 1:**

Time units (work units):

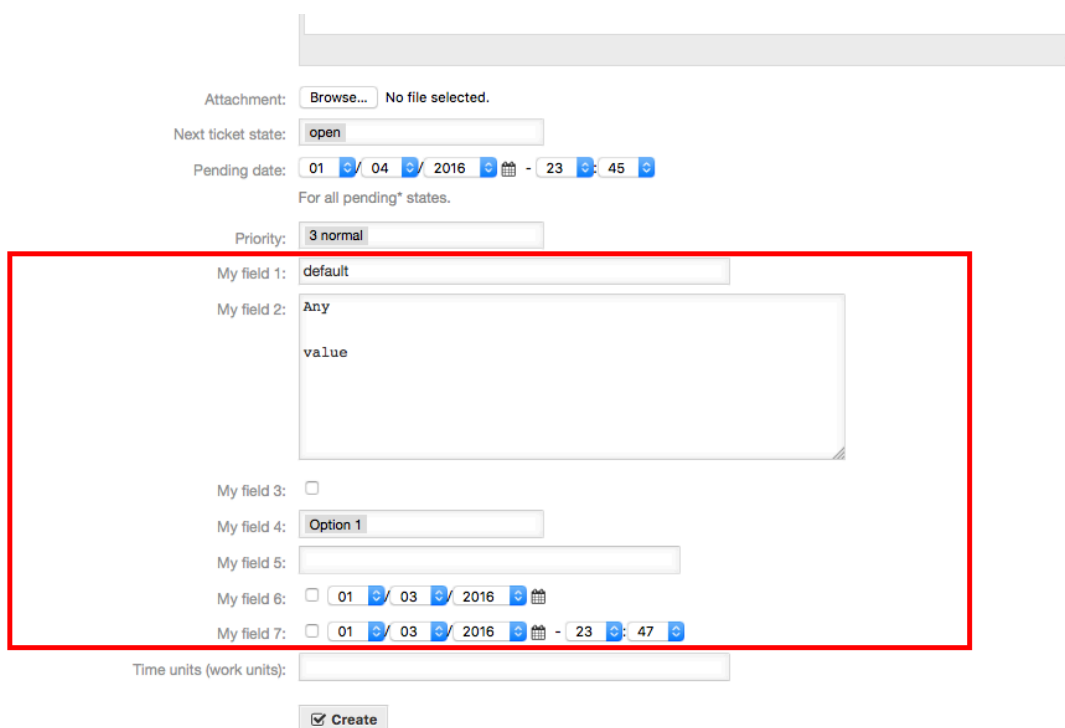
Create

例 4.24. 在新建电话工单窗口激活数个字段。

- 主组 : Ticket
- 子组 : Frontend::Agent::Ticket::ViewPhoneNew
- 设置 : Ticket::Frontend::AgentTicketPhone###DynamicField
- 值 :

键	值
Field1	1
Field2	1
Field3	1
Field4	1
Field5	1
Field6	1
Field7	1

图 4.92. 在新建电话工单窗口显示数个字段为必填

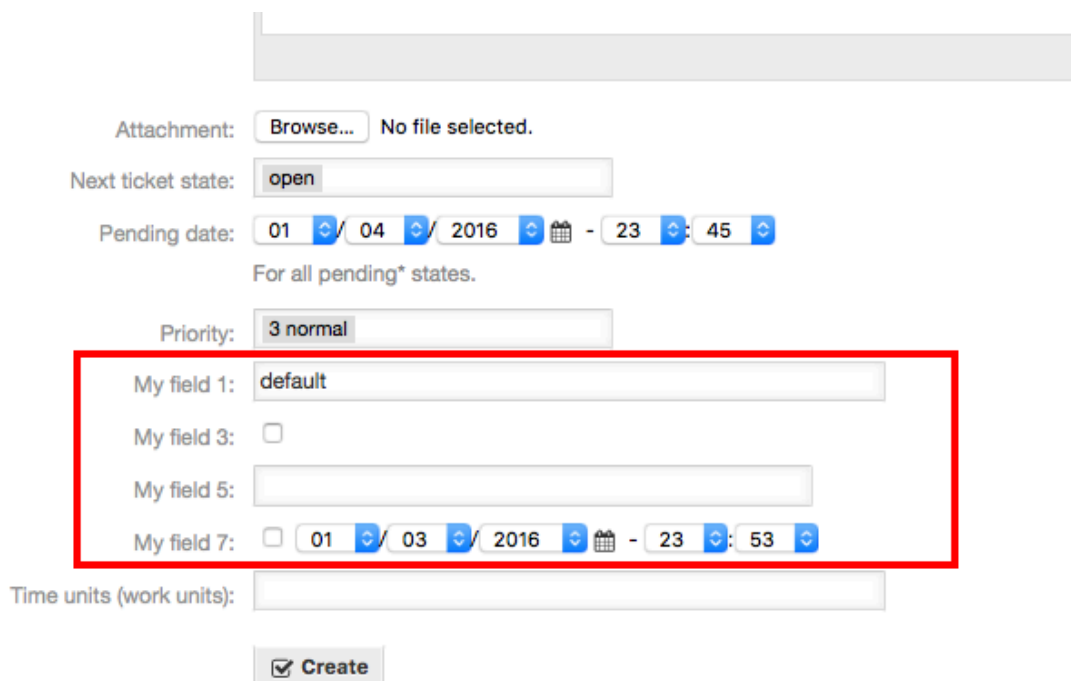


例 4.25. 在新建电话工单窗口去激活一些字段。

- 主组 : Ticket
- 子组 : Frontend::Agent::Ticket::ViewPhoneNew
- 设置 : Ticket::Frontend::AgentTicketPhone###DynamicField
- 值 :

键	值
Field1	1
Field2	0
Field3	1
Field4	0
Field5	1
Field6	0
Field7	1

图 4.93. 在新建电话工单窗口不显示一些字段

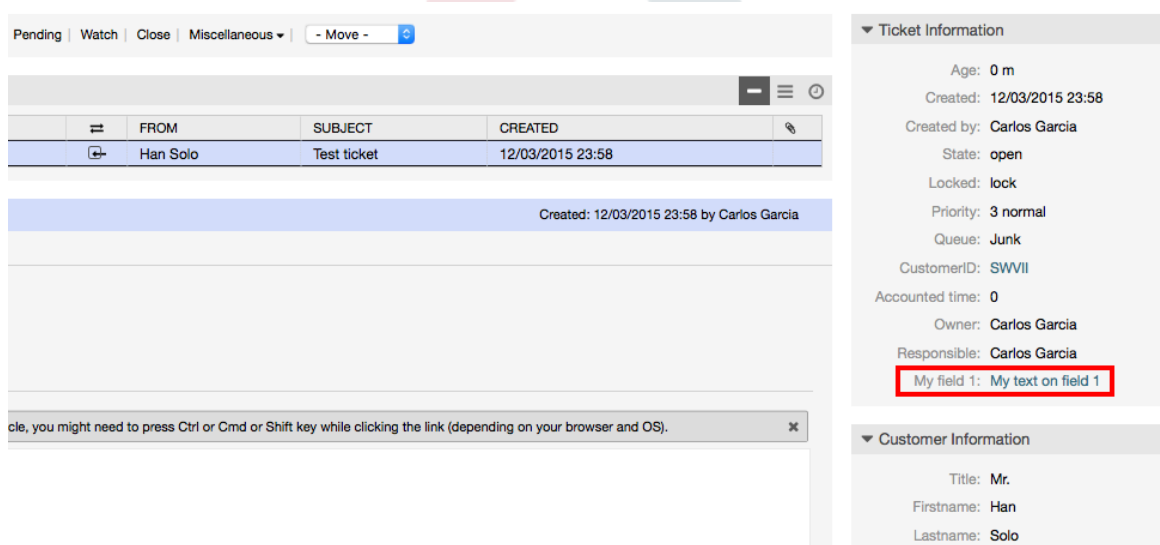


例 4.26. 在工单详情窗口激活Field1。

- 主组 : Ticket
- 子组 : Frontend::Agent::Ticket::ViewZoom
- 设置 : Ticket::Frontend::AgentTicketZoom###DynamicField
- 值 :

键	值
Field1	1

图 4.94. 在工单详情窗口显示Field1



例 4.27. 在工单概览简洁模式视图激活Field1。

- 主组 : Ticket
- 子组 : Frontend::Agent::TicketOverview
- 设置 : Ticket::Frontend::OverviewSmall###DynamicField
- 值 :

键	值
Field1	1

图 4.95. 在工单概览简洁模式视图显示Field1

QueueView: Junk

My Queues (2) Junk (1) Raw (1)

All tickets 2 Available tickets 1

Bulk 1-1 of 1 S M L

TICKET#	▲ AGE	FROM / SUBJECT	STATE	LOCK	QUEUE	MY FIELD 1	OWNER	CUSTOMERID
2016010354000013	11 m	Han Solo Test ticket	open	lock	Junk	My text on field 1	Carlos Garcia	SWVII

这个设置影响 : 升级视图、锁定视图、队列视图、负责人视图、状态视图、服务视图和状态视图窗口。

### 10.2.11. 通过一个工单事件模块设置动态字段的默认值

一个工单事件 (如工单创建) 可以触发设置某个字段的值 (如果该字段还没有一个值)。

#### 注意

通过使用这个方法, 在编辑窗口 (例如新建电话工单) 无法看到这个默认值, 因为它是在工单创建之后设置的。

要激活这个功能, 需要启用下列的设置: "Ticket::EventModulePost###TicketDynamicFieldDefault"。

例 4.28. 在工单创建事件激活Field1。

- 主组 : Ticket
- 子组 : Core::TicketDynamicFieldDefault
- 设置 : Ticket::TicketDynamicFieldDefault###Element1

#### 注意

这个配置可以在16个Ticket::TicketDynamicFieldDefault###Element设置中的任意一个里面设置。

如果超过16个动态字段, 需要设置一个定制的XML文件并放于目录\$OTRS\_HOME/Kernel/Config/files下来扩展这个功能。

- 值 :

键	值
事件	工单创建

键	值
名称	Field1
值	一个新值

## 10.2.12. 通过用户‘偏好设置’设置默认值

动态字段默认值可以被用户定义在偏好设置中的值覆盖。

使用这个方法，这个动态字段将在所有激活了它的窗口显示这个默认值（如果该字段没有一个不同的值）。

系统配置中的“PreferencesGroups###DynamicField”设置位于子组“Frontend::Agent::Preferences”。这个设置是一个例子，关于如何为一个选定的用户在用户偏好设置窗口创建一个条目来设置一个独有的动态字段默认值。这个设置的限制是它只允许使用一个动态字段。如果2个或更多的字段需要使用这个功能，需要创建一个定制的XML配置文件来添加多个类似于这个的设置。

### 注意

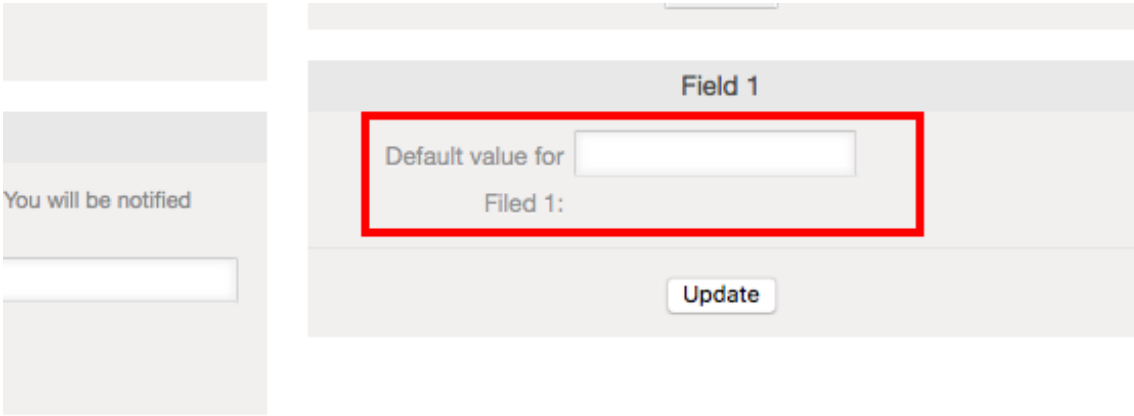
记住：如果在一个新的XML文件中添加更多的设置，每个设置的名称需要是系统唯一的，且不能是“PreferencesGroups###DynamicField”。例如：PreferencesGroups###101-DynamicField-Field1、PreferencesGroups###102-DynamicField-Field2、PreferencesGroups###My-Field1、PreferencesGroups###My-Field2等等。

**例 4.29.** 在用户‘偏好设置’中激活Field1。

- 主组：Ticket
- 子组：Frontend::Agent::Preferences
- 设置：PreferencesGroups###101-DynamicField-Field1
- 值：

键	值
事件	工单创建
Active ( 活动的 )	1
Block ( 块 )	Input ( 输入 )
Column ( 列 )	其它设置
Data: ( 数据 : )	[% Env("UserDynamicField_Field1") %]
Key: ( 键 : )	My Field 1
Label: ( 标签 : )	Default value for: My Field 1 ( Field1的默认值 )
Module: ( 模块 : )	Kernel::Output::HTML::PreferencesGeneric
PrefKey:	UserDynamicField_Field1
Prio:	7000

图 4.96. 在用户‘偏好设置’中激活Field1



The image shows a user preference configuration interface. On the right, a section titled 'Field 1' contains a text input field labeled 'Default value for Filed 1:'. This field is highlighted with a red rectangular border. Below the input field is an 'Update' button. On the left, a partial view of another preference section is visible, showing the text 'You will be notified' and an empty text input field.

## 11. 通用接口

OTRS通用接口包含一个多层框架，让OTRS能够通过一个WEB服务与其它系统通讯，这个通讯可以是双向的：

- OTRS作为提供者：OTRS作为一个服务器运行，监听来自外部系统的请求、处理信息、执行请求的操作并答复请求。
- OTRS作为请求者：OTRS作为一个客户端运行，收集信息、发送请求到远程系统并等待答复。

### 11.1. 通用接口层

通用接口是基于层构建的模型，使用灵活且易于定制。

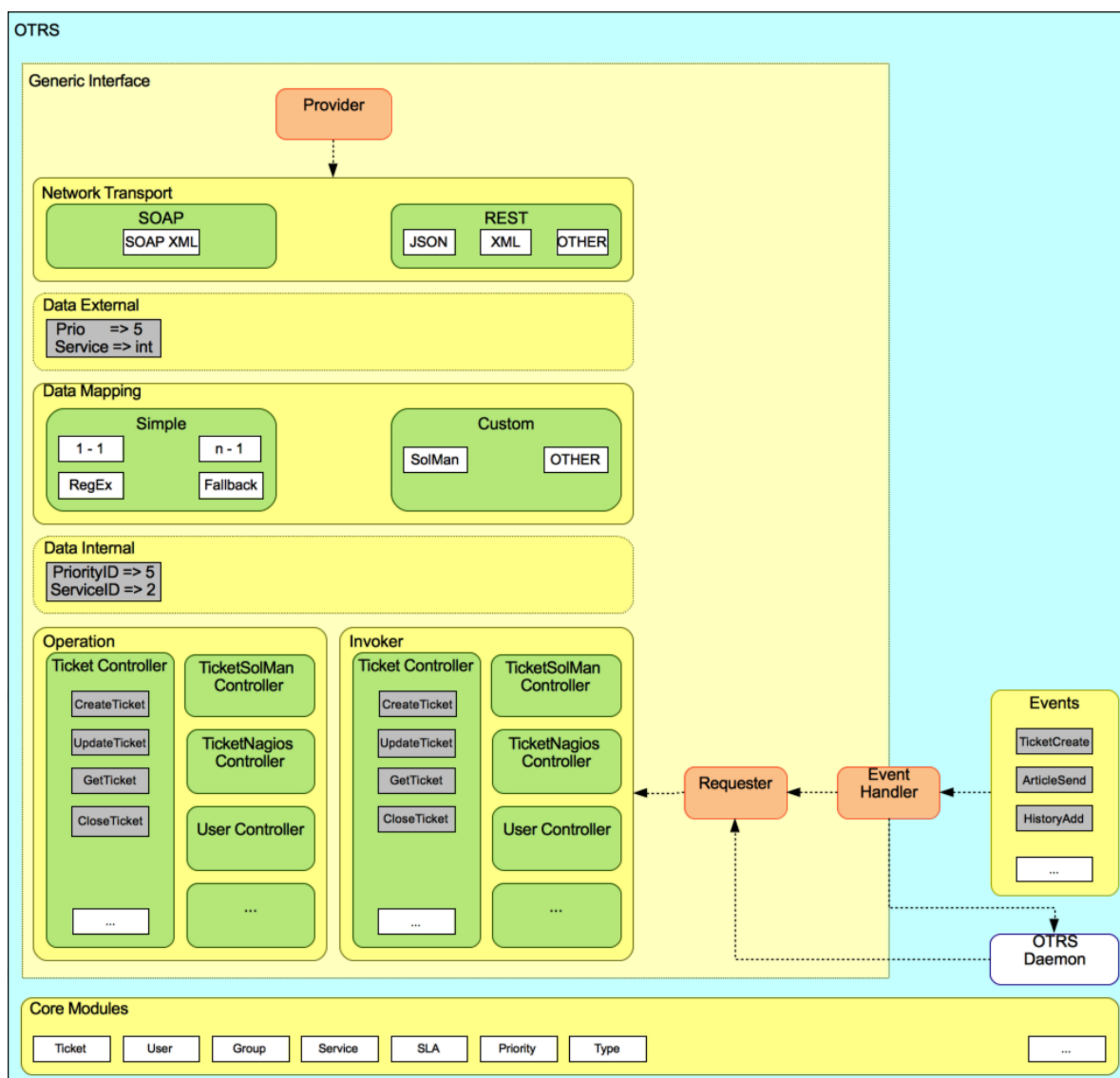
层是一个文件的集合，控制通用接口如何执行一个WEB服务的不同部分。使用正确的配置，你可以给不同的外部系统构建不同的WEB服务，而不用创建新的模块。

#### 注意

如果远程系统不支持目前通用接口捆绑的模块，需要为那个特定的WEB服务开发特殊的模块。

OTRS自带的通用接口模块随着时间流逝会更新和增长。

图 4.97. 接口的‘层’图解



### 11.1.1. 网络传输

本层负责与远程系统的正确通讯。它在OTRS作为提供者时接收请求并生成响应，在OTRS作为请求者时生成请求并接受响应。

提供者通讯由一个新的WEB服务处理程序“nph-genericinterface.pl”处理。

请求者通讯可以在一个通用接口模块或其它任何OTRS模块触发事件过程中初始化。这个事件由事件处理程序捕获，取决于配置，这个事件由请求者对象直接处理或委派给计划程序（一个单独的守护进程，设计来异步处理任务）处理。

### 11.1.2. 数据映射

本层负责在OTRS和远程系统之间转化数据结构（内部数据和外部数据层）。通常远程系统拥有与OTRS不同的数据结构（包括不同的值和这些值的名称），这就是本层的重要性：将收到的信息转为OTRS能够理解的东西，在相反的方向上将这些信息使用远程系统的数据字典发送给每个远程系统。

示例：OTRS中的“Priority（优先级）”在远程系统中可能叫做“Prio”，它的值“1-低”（OTRS）应当映射为远程系统中的“Information”。



### 11.1.3. 控制器

控制器是相似的操作或调用程序的合集。例如，一个工单控制器可能包含数个标准工单操作。可以定制控制器，例如，一个“TicketExternalCompany”控制器可能包含与标准工单控制器相似的功能，但有不同的数据接口或函数名称（适应远程系统的函数名称）或完全不同的代码。

一个通用接口应用可以与一个远程系统同步信息，只可以跟另一个相同类型的远程系统。在这种情况下，需要开发新的控制器，并且操作和调用程序必须模拟远程系统行为，这样OTRS暴露的接口就类似于远程系统的接口。

### 11.1.4. 操作（OTRS作为提供者）

操作是一个能在OTRS内执行的单个动作。所有的操作有相同的编程接口，它们接收数据到一个指定的参数，并返回一个数据结构，包含成功状态、潜在的错误消息和返回数据。

通常操作使用已经映射的数据（内部）来调用核心模块并执行OTRS中的动作如：创建一个工单、更新一个用户、使一个队列无效、发送一个通知等等。一个操作有OTRS API的全部权限以执行这个动作。

### 11.1.5. 调用程序（OTRS作为请求者）

调用程序是OTRS在一个远程系统上执行的动作。调用程序使用OTRS核心模块来处理 and 收集需要的信息以创建一个请求。当信息准备好后它必须映射到远程系统格式以便发送给远程系统，远程系统处理这个信息、执行动作并发回响应，要么处理成功要么处理错误。

## 11.2. 通用接口通讯流程

通用接口有定义的流程来执行作为提供者和请求者的操作

这些流程描述如下：

### 11.2.1. OTRS作为提供者

#### 11.2.1.1. 远程请求：

##### 1. HTTP请求

- OTRS接收到HTTP请求并传递它。
- 提供者模块负责执行和控制这些操作。

##### 2. 网络传输

- 网络传输模块解码数据有效载荷并从剩余数据中分离操作的名称。
- 操作名称和操作数据都返回给提供者。

##### 3. 外部数据

- 从远程系统发来的数据（这不是一个基于模块的层）。

##### 4. 映射

- 按这个操作指定的映射配置（进入的请求数据映射）将数据从外部系统格式转化为OTRS内部格式。
- 已转化的数据返回给提供者。

##### 5. 内部数据

- 数据已经转化并准备传递到操作（这不是一个基于模块的层）。

## 6. 操作

- 接收并验证数据。
- 执行用户访问控制。
- 执行操作。

### 11.2.1.2. OTRS响应：

#### 1. 操作

- 返回结果数据到提供者。

#### 2. 内部数据

- 数据从操作返回。

#### 3. 映射

- 数据按映射配置中指定的映射（出站响应数据的映射）转化回远程系统的格式。
- 已转化的数据返回给提供者。

#### 4. 外部数据

- 转化的数据准备传递给网络传输作为响应。

#### 5. 网络传输

- 接收远程系统格式的数据。
- 为这个网络传输类型构建一个有效的响应。

#### 6. HTTP响应

- 响应发送回WEB服务客户端。
- 如果出现一个错误（例如SOAP错误、HTTP错误等等），发送一个错误响应给远程系统。

### 11.2.2. OTRS作为请求者

#### 11.2.2.1. OTRS请求：

##### 1. 事件触发器处理程序

- 基于WEB服务配置确定请求是同步还是异步。
  - 同步
    - 直接调用请求者以便创建一个新的请求，并通过这个层传递。
  - 异步
    - 为OTRS守护进程创建一个新的通用接口（请求者）任务，通过委派该请求执行到计划程序进程，大大改善了用户体验。否则准备该请求和远程执行需要的所有时间都会加到触发这些请求的OTRS事件上。
    - 在下一个循环中OTRS守护进程读取新任务并创建一个到请求者的调用，请求者会创建一个新的请求并通过这个层传递它。

##### 2. 调用程序

- 接收来自事件的数据。
- 验证收到的数据（如果需要）。
- 调用核心模块来补足数据（如果需要）。
- 返回请求数据结构，或发送一个中止通讯信号给请求者以取消这个请求。

### 3. 内部数据

- 自请求者传来的数据（这不是一个基于模块的层）。

### 4. 映射

- 按映射配置（出站响应数据的映射）指定的格式将数据转化为远程系统的格式。
- 已经转化的数据返回给请求者。

### 5. 外部数据

- 转化后的数据准备发送给远程系统。

### 6. 网络传输

- 从请求者接收远程操作名称和已经转化为远程系统格式的数据
- 为网络传输构建一个有效的请求
- 将请求发送给远程系统并等待响应。

## 11.2.2.2. 远程响应：

### 1. 网络传输

- 接收响应并解码数据的有效载荷。
- 将数据返回给请求者。

### 2. 外部数据

- 从远程系统收到的数据。

### 3. 映射

- 数据已用这个操作的映射配置指定的映射（进入的响应数据映射）从外部系统格式转化为OTRS内部格式。
- 已经转化的数据返回给请求者。

### 4. 内部数据

- 已经转化的数据返回给请求者。

### 5. 调用程序

- 接收返回的数据。
- 根据每个调用程序的特殊需要处理数据（包括可能有的错误处理）。
- 返回调用程序结果和数据到请求者。

### 6. 事件处理程序或OTRS守护进程

- 接收来自请求的数据。在使用OTRS守护进程的情况下，将来在数据中可能还包含了创建一个任务的信息。

## 11.3. Web服务

一个WEB服务是两个系统（在我们的例子中，就是OTRS和一个远程系统）间进行通讯的方法。

WEB服务的核心是它的配置，在配置中定义了WEB服务能够在内部执行什么操作(Operation操作)、OTRS请求能够在远程系统执行什么操作(Invokers调用程序)，数据如何从一个系统转换到另外的系统(Mapping映射)以及通过什么协议进行通讯 (Transport传输)

通用接口是一个框架，使得在OTRS中能够用预定义的方法、使用已经构建的相互独立且可互换的块来创建WEB服务。

## 11.4. WEB服务图形界面

WEB服务图形界面是一个工具，允许你在用户友好、便捷的界面构建复杂的WEB服务配置。它允许：

- 创建和删除WEB服务。
- 导入和导出已有WEB服务的配置（用YAML文件格式）。
- 在WEB服务历史窗口查看、还原和导出已有WEB服务的老配置。
- 在调试窗口跟踪每个WEB服务所有的通讯日志。

### 11.4.1. WEB服务概览

在系统管理页面系统管理员部分的“WEB服务”链接通向WEB服务概览窗口，你可以管理WEB服务的配置。在这个窗口你可以添加新的WEB服务或修改已有WEB服务的配置。

每个WEB服务配置窗口的上部分有一行‘面包条’风格的导航路径，通过这个导航路径可以知道我们在WEB服务配置中的准确位置，还可以让用户随时跳回到配置流程的任何位置（这个跳回操作不会保存任何修改内容）。

#### 注意

要创建一个新的WEB服务，点击“添加WEB服务”按钮，并输入需要的信息。

图 4.98. WEB服务概览

GenericInterface Web Service Management - Overview

You are here: > Web Services

NAME	DESCRIPTION	REMOTE SYSTEM	PROVIDER TRANSPORT	REQUESTER TRANSPORT	VALIDITY
Webservice one	First webservice	Any description	HTTP::SOAP	HTTP::SOAP	valid
Webservice two	-	-	-	-	valid

### 11.4.2. 添加WEB服务

这部分必填的字段只有WEB服务的“名称”，要求系统唯一且不得留空。其它字段也是配置需要的如“调试阈值”和“有效性”，但是这些字段已经有默认值了。

“调试阈值”的默认值是“调试”，当配置为这个方式时所有的通讯日志都会记录到数据库中。每个随后的调试阈值都有更多的限制并会丢弃比当前阈值级别低的通讯日志。

## 调试阈值级别(从低到高)

- Debug ( 调试 )
- Info ( 信息 )
- Notice ( 注意 )
- Error ( 错误 )

还可以为“OTRS作为服务提供方”和“OTRS作为服务请求方”定义网络传输协议。

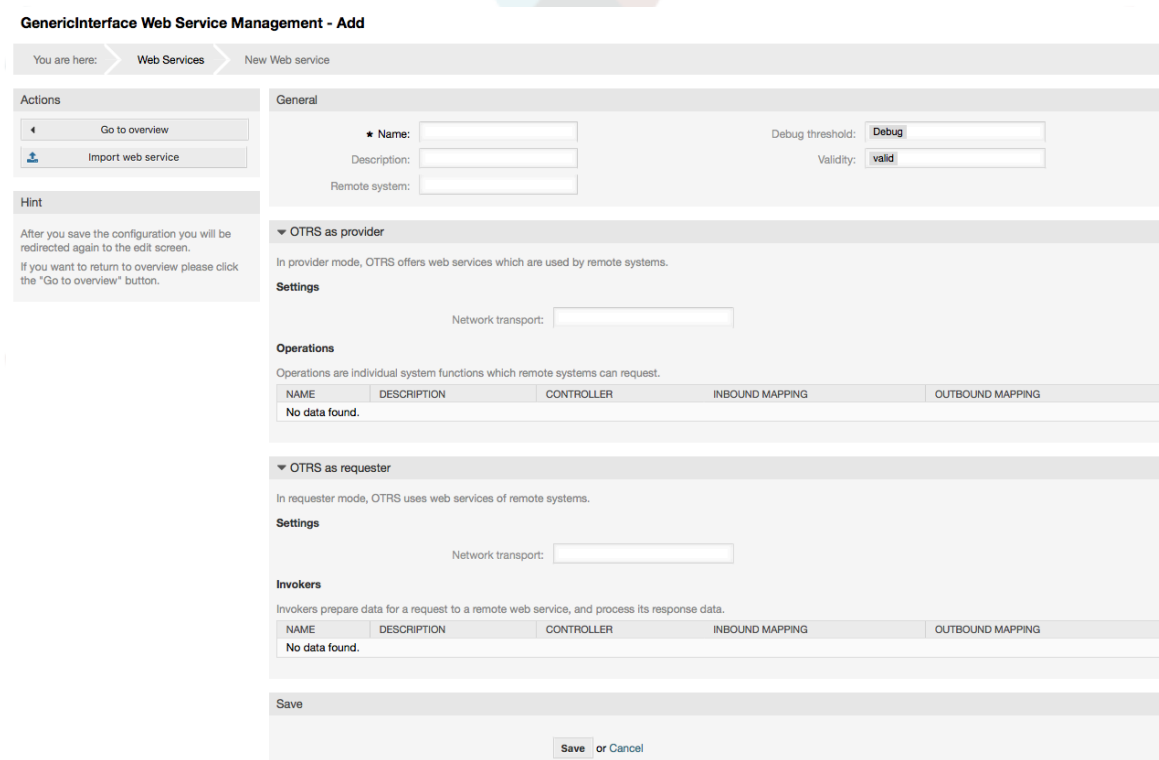
点击“保存”按钮来将新的WEB服务注册到数据库中，或点击“取消”按钮丢弃这个操作。现在可以返回到WEB服务概览窗口。

如果你已经有一个YAML格式的WEB服务配置文件，你可以点击窗口左侧的“导入WEB服务”按钮。导入WEB服务的更多信息请检查下一部分的“修改WEB服务”。

## 注意

要为一个WEB服务添加或修改更多细节，点击WEB服务概览窗口中WEB服务的名称。

图 4.99. 添加WEB服务



**GenericInterface Web Service Management - Add**

You are here: > Web Services > New Web service

**Actions**

- Go to overview
- Import web service

**Hint**

After you save the configuration you will be redirected again to the edit screen. If you want to return to overview please click the "Go to overview" button.

**General**

★ Name:  Debug threshold:

Description:  Validity:

Remote system:

**OTRS as provider**

In provider mode, OTRS offers web services which are used by remote systems.

**Settings**

Network transport:

**Operations**

Operations are individual system functions which remote systems can request.

NAME	DESCRIPTION	CONTROLLER	INBOUND MAPPING	OUTBOUND MAPPING
No data found.				

**OTRS as requester**

In requester mode, OTRS uses web services of remote systems.

**Settings**

Network transport:

**Invokers**

Invokers prepare data for a request to a remote web service, and process its response data.

NAME	DESCRIPTION	CONTROLLER	INBOUND MAPPING	OUTBOUND MAPPING
No data found.				

**Save**

or

### 11.4.3. 导入WEB服务样例

在 OTRS Business Solution™ 中有可用的示例Web服务，你知道吗？

### 11.4.4. 修改WEB服务

在这个窗口你有处理一个WEB服务所有部分的完整功能集。在左侧的操作栏中你可以找到一些按钮，允许你在一个WEB服务上执行所有可能的操作。

- 克隆WEB服务。

- 导出WEB服务。
- 导入WEB服务。
- 配置历史。
- 删除WEB服务
- 调试器

## 注意

“配置历史”和“调试器”会打开不同的窗口。

### 11.4.4.1. 克隆WEB服务

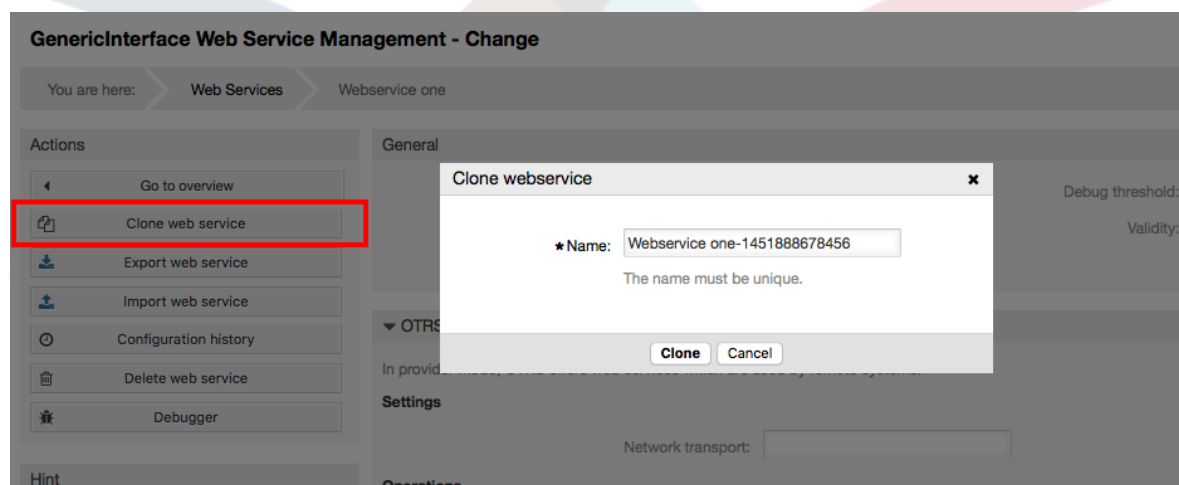
要克隆一个WEB服务，你需要点击“克隆WEB服务”按钮。在打开的对话框中你可以为这个克隆的WEB服务使用默认的名称或设置一个新名称。

## 注意

记住：WEB服务的名称必须是系统中唯一的。

点击“克隆”按钮来创建这个WEB服务的克隆或“取消”按钮关闭这个对话框。

图 4.100. 克隆WEB服务



### 11.4.4.2. 导出WEB服务

“导出WEB服务”按钮让你转储当前WEB服务配件到一个YAML文件，下载并保存到你的文件系统中。这在你要从一个服务器迁移WEB服务到另外的服务器时特别有用，例如从测试环境到生产系统。

## 警告

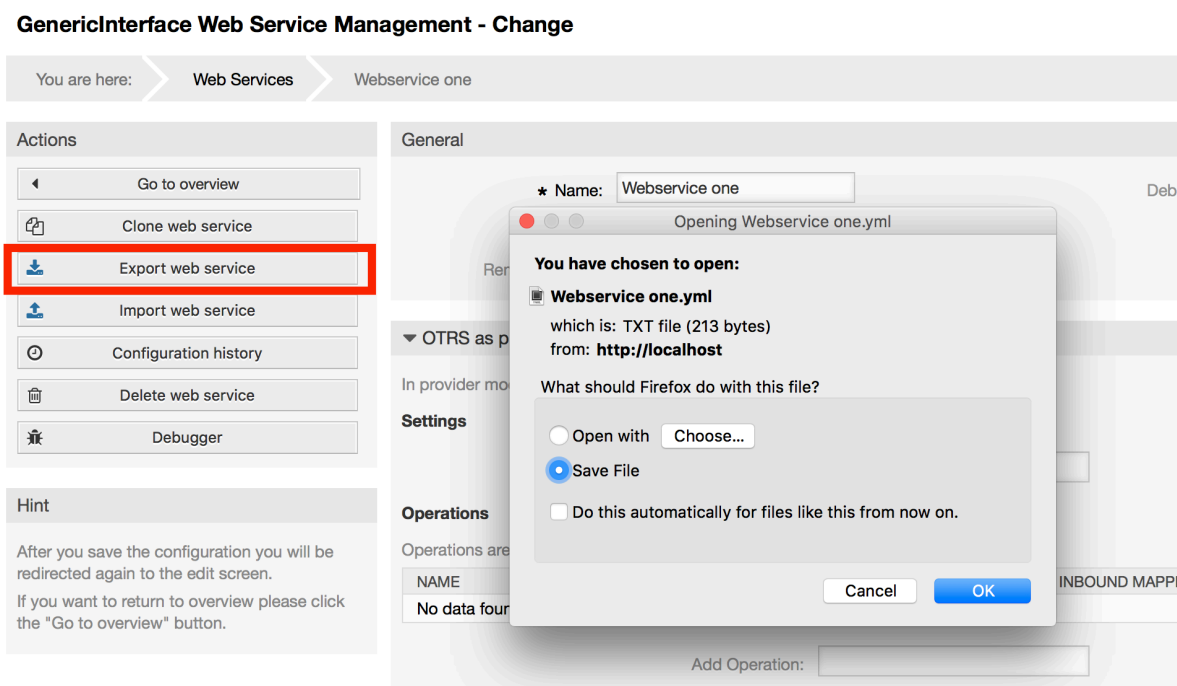
所有存储在WEB服务配置中的密码都将以纯文本格式导出。

在点击“导出WEB服务”按钮后，你的浏览器会出现一个保存文件对话框，就像你在网页上点击了一个文件下载链接一样的。

## 注意

不同操作系统中不同浏览器有它自己的保存对话框窗口和风格。取决于浏览器和它的配置，可能不会显示对话框而是直接将文件保存到一个默认目录中。如有需要请检查你的浏览器文档以获取更多的特定说明。

图 4.101. 导出WEB服务



### 11.4.4.3. 导入WEB服务

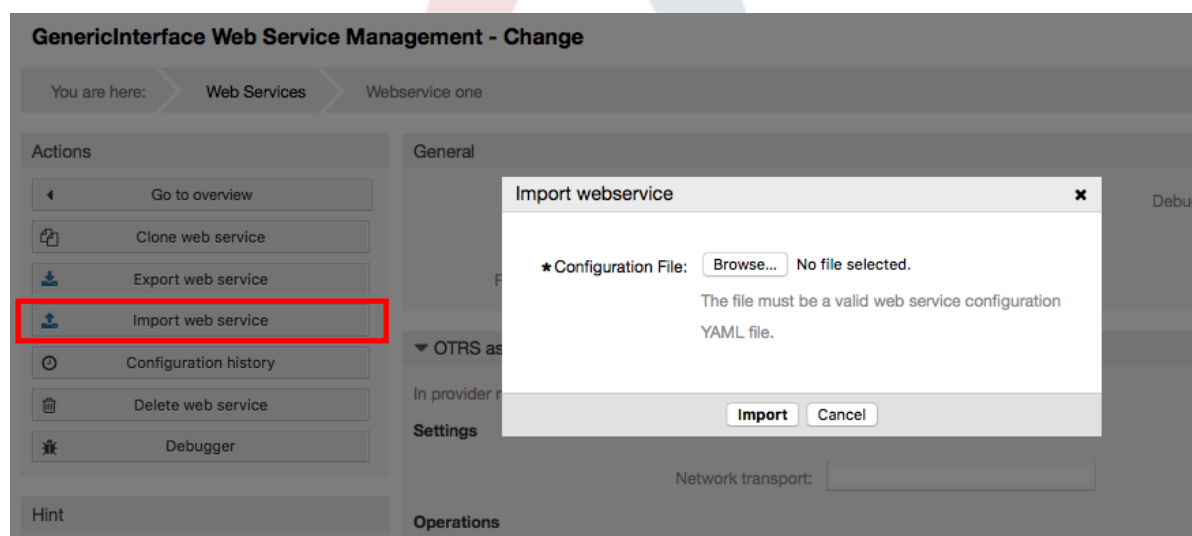
导入WEB服务功能必须使用一个有效的WEB服务配置的YAML文件。点击“导入WEB服务”按钮，浏览配置文件或在输入框中输入文件的完整路径。

点击“导入”按钮以从一个文件创建一个新的WEB服务或“取消”关闭这个对话框。

#### 注意

WEB服务的名称会使用配置文件的文件名（例如文件是MyWebservice.yml，则WEB服务将命名为MyWebservice）。如果在系统中已经有一个与之同名的WEB服务，系统会打开修改WEB服务窗口，让你修改导入的WEB服务的名称。

图 4.102. 导入WEB服务



### 11.4.4.4. WEB服务历史

对WEB服务配置所做的每个修改都会在WEB服务历史中创建一条新条目（就像一个日志）。WEB服务历史窗口显示一个WEB服务配置所有版本的列表。在“配置历史列表”中的每一行（版本）代表WEB服务历史中一个单独的修订。

点击一行可以显示在特定日期/时间的整个配置，这个配置会显示在本窗口的“历史详细信息”部分。你还可以在这里导出选定的WEB服务配置版本或者恢复这个版本到当前的WEB服务配置。

“导出WEB服务配置”与修改WEB服务窗口中的“导出WEB服务”功能完全一样。更多信息请参考那部分内容。

如果对当前WEB服务配置的修改没有按预期工作，可以手动轻易恢复修改内容到原状，你可以点击“恢复WEB服务配置”按钮。这会打开一个对话框问你是否确定恢复到这个WEB服务配置。点击对话框中的“恢复WEB服务配置”按钮用选定版本替换当前的配置，或点击“取消”直接关闭对话框。

#### 警告

记住：存储在WEB服务配置中的任何密码都将以纯文本格式导出。

恢复一个配置时请一定小心，因为这个过程是不可逆的。

图 4.103. WEB服务历史

GenericInterface Configuration History for Web Service Webservice one

You are here: > Web Services > Webservice one > History

**Actions**

Go back to Web Service

**Hint**

Here you can view older versions of the current web service's configuration, export or even restore them.

**Configuration History List**

VERSION	CREATE TIME
11	2016-01-04 14:25:30
10	2016-01-04 14:25:26
9	2016-01-04 14:25:13
8	2016-01-04 14:25:10
7	2016-01-04 14:24:54

Select a single configuration version to see its details.

---

**History Details: Version 11, 2016-01-04 14:25:30**

Export web service configuration | Restore web service configuration

```

---
Debugger:
  DebugThreshold: debug
  TestMode: '0'
Description: First webservice
Provider:
  Operation:
    CreateTicket:
      Description: ''
      MappingInbound: {}
      MappingOutbound: {}
      Type: Ticket::TicketCreate
    GetTicket:
      Description: ''
      MappingInbound: {}
      MappingOutbound: {}
      Type: Ticket::TicketGet
    SearchTicket:
      Description: ''
      MappingInbound: {}
      MappingOutbound: {}
      Type: Ticket::TicketSearch
    UpdateTicket:
      Description: ''
      MappingInbound: {}
      MappingOutbound: {}
      Type: Ticket::TicketUpdate
  Transport:
    Config:
      Authentication: {}
      Type: HTTP::SOAP
  RemoteSystem: Any Description
  Requester:
    Transport:
      Config:
        Authentication: {}
        Type: HTTP::REST
          
```

### 11.4.4.5. 删除WEB服务

有时需要完全删除一个WEB服务，如要这样做你可以点击“删除WEB服务按钮”，会弹出一个确认对话框。

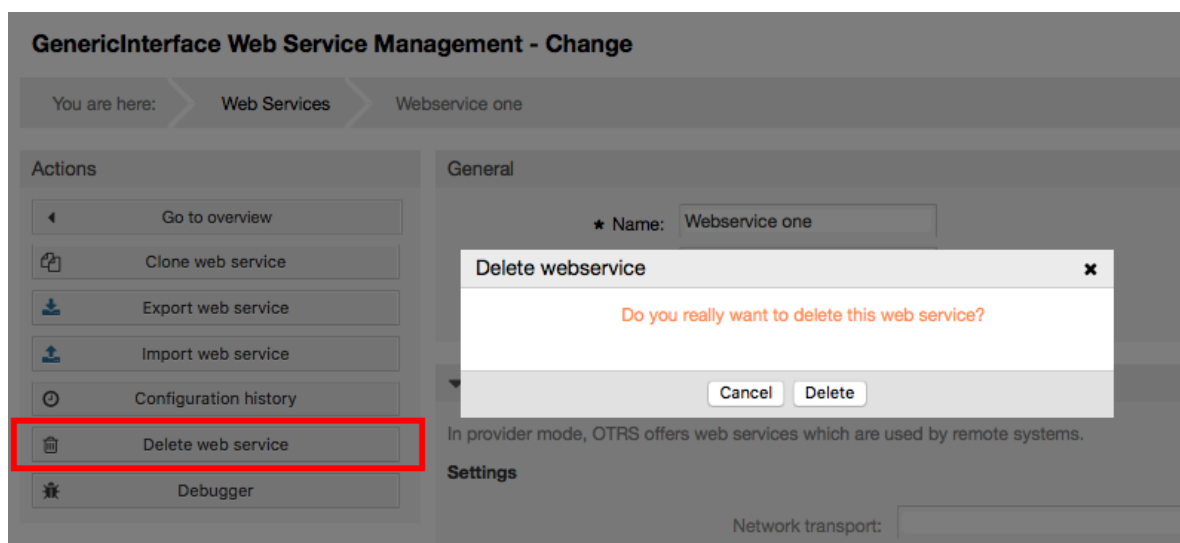
点击“删除”按钮来确认删除这个WEB服务或点击“取消”关闭对话框。

#### 警告

删除一个WEB服务后无法撤销，请在删除一个WEB服务时一定要小心。



图 4.104. 删除WEB服务



#### 11.4.4.6. WEB服务调试器

调试器存储了一个WEB服务的日志。在调试器窗口你可以跟踪该WEB服务作为提供者或请求者类型的所有通讯日志。

显示这个窗口后，请求列表开始载入。在列表完全载入后你可以选择一行（即一条通讯序列）来检查它的详细信息，这个详细信息显示在窗口下方。

你可以使用窗口右边的过滤器来收缩通讯列表，你可以过滤的内容有：

- 通讯类型（提供者或请求者）
- 日期：在一个特定日期之前或之后
- 远程IP地址
- 以上所有组合

在设置了过滤器后，点击“刷新”按钮，就会显示满足搜索条件的新列表。

#### 注意

取决于过滤器的搜索条件，新列表可能无返回结果。

在窗口左侧的操作栏中你可以选择“返回WEB服务”，或点击“清除”按钮清空调试器日志，这会打开一个对话框确认清除日志，点击对话框中的“清除”按钮执行清除操作或点击“取消”关闭这个对话框。

在“请求详细信息”部分你可以看到选定通讯的所有详细信息。在这里你可以跟踪完整流并检查可能的错误或确认成功响应。

**图 4.105. WEB服务调试器**

GenericInterface Debugger for Web Service WebserviceOne

You are here: Web Services > WebserviceOne > Debugger

**Actions**

Go back to web service

Clear

**Request List**

Provider	2016-01-04 19:09:51	127.0.0.1
Provider	2016-01-04 19:10:57	127.0.0.1
Provider	2016-01-04 19:11:20	127.0.0.1
Provider	2016-01-04 19:11:20	127.0.0.1
Provider	2016-01-04 19:13:36	127.0.0.1
Provider	2016-01-04 19:14:14	127.0.0.1

Filter by type:

Filter from: 01 13 2015

Filter to: 01 04 2016

Filter by remote IP:

Limit: 100

Refresh

Select a single request to see its details.

**Request Details**

Communication sequence started (2016-01-04 19:14:14, debug)

Received data by provider from remote system (2016-01-04 19:14:14, debug)

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:tic="http://www.otrs.org/TicketConnector/">
  <soapenv:Header/>
  <soapenv:Body>
    <tic:TicketGet>
      <!--You have a CHOICE of the next 3 items at this level-->
      <!--Optional:-->
      <UserLogin></UserLogin>
      <!--Optional:-->
      <CustomerUserLogin?</CustomerUserLogin>
      <!--Optional:-->
      <SessionID>14j19y84EBwLGGpsEVCs81U0N6kaeRpx</SessionID>
      <ChallengeToken>L0JUK0RgS76kDwYQ1h5zTnrLdeYQG8yg</ChallengeToken>
      <!--Optional:-->
      <Password>test</Password>
      <!--1 or more repetitions:-->
      <TicketID>1</TicketID>
      <!--Optional:-->
      <DynamicFields?</DynamicFields>
      <!--Optional:-->
      <Extended?</Extended>
      <!--Optional:-->
      <AllArticles?</AllArticles>
      <!--Optional:-->
      <ArticleSenderType?</ArticleSenderType>
      <!--Optional:-->
      <ArticleOrder?</ArticleOrder>
      <!--Optional:-->
      <ArticleLimit?</ArticleLimit>
      <!--Optional:-->
      <Attachments?</Attachments>
    </tic:TicketGet>
  </soapenv:Body>
</soapenv:Envelope>
```

Detected operation 'TicketGet' (2016-01-04 19:14:14, debug)

No data provided

Incoming data before mapping (2016-01-04 19:14:14, debug)

```
$VAR1 = {
  'AllArticles' => '?',
  'ArticleLimit' => '?',
  'ArticleOrder' => '?',
  'ArticleSenderType' => '?',
  'Attachments' => '?',
  'ChallengeToken' => 'L0JUK0RgS76kDwYQ1h5zTnrLdeYQG8yg',
  'CustomerUserLogin' => '?',
  'DynamicFields' => '?',
  'Extended' => '?',
  'Operation' => 'TicketGet',
  'Password' => 'test',
  'SessionID' => '14j19y84EBwLGGpsEVCs81U0N6kaeRpx',
  'TicketID' => '1',
  'UserLogin' => ''
};
```

Outgoing data before mapping (2016-01-04 19:14:15, debug)

Returning provider data to remote system (HTTP Code: 200) (2016-01-04 19:14:15, debug)

```
<?xml version="1.0" encoding="UTF-8"?><soap:Envelope soap:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
```

### 11.4.4.7. 修改WEB服务配置

回到修改WEB服务窗口，现在我们准备回顾窗口右侧。在这里我们可以修改一个WEB服务的所有通用信息如名称、描述、调试阈值等等。此外下面还有两个部分允许我们修改通讯类型“OTRS作为提供者”和“OTRS作为请求者”的特定参数。

WEB服务配置需要在每个级别保存，这意味着如果修改了一个设置，到其它部分（这个配置更深层次的部分）的链接将被禁用，强迫你保存当前的配置级别。在保存后禁用了的链接会被重新启用，允许你使用这个配置继续。

在“OTRS作为提供者”部分可以设置或配置网络传输协议，只有注册了的网络传输后端才会显示在列表中。要配置网络传输，点击“配置”按钮。在这部分还可以添加新操作，从“添加操作”列表选择一个可用的操作，这会引领你到操作配置窗口。在保存了这个新操作后它就会在上面的表格里列出。

在“OTRS作为请求者”部分与前面非常类似，但你可以在这里添加调用程序而不是“操作”。

点击“保存”按钮以保存并继续配置这个WEB服务，“保存并完成”保存并返回到WEB服务概览窗口，或者“取消”会丢弃当前配置级别的修改内容并返回到WEB服务概览窗口。

图 4.106. 修改WEB服务

GenericInterface Web Service Management - Change

You are here: > Web Services > Webservice one

**Actions**

- Go to overview
- Clone web service
- Export web service
- Import web service
- Configuration history
- Delete web service
- Debugger

**Hint**

After you save the configuration you will be redirected again to the edit screen. If you want to return to overview please click the "Go to overview" button.

**General**

Name:  Debug threshold:

Description:  Validity:

Remote system:

---

**OTRS as provider**

In provider mode, OTRS offers web services which are used by remote systems.

**Settings**

Network transport:

**Operations**

Operations are individual system functions which remote systems can request.

NAME	DESCRIPTION	CONTROLLER	INBOUND MAPPING	OUTBOUND MAPPING
CreateTicket	-	Ticket::TicketCreate	-	-
GetTicket	-	Ticket::TicketGet	-	-
SearchTicket	-	Ticket::TicketSearch	-	-
UpdateTicket	-	Ticket::TicketUpdate	-	-

Add Operation:

---

**OTRS as requester**

In requester mode, OTRS uses web services of remote systems.

**Settings**

Network transport:

**Invokers**

Invokers prepare data for a request to a remote web service, and process its response data.

NAME	DESCRIPTION	CONTROLLER	INBOUND MAPPING	OUTBOUND MAPPING
No data found.				

Add Invoker:

---

**Save**

or  or

## 注意

你其它的通用接口配置窗口（如网络传输、操作、调用程序和映射窗口）一样，初始化配置（添加）窗口仅提供了两个选项“保存”和“取消”。如果是重新查看这个配置，会出现新的选项“保存并完成”。这个功能的行为定义如下。

“保存”将存储当前配置级别到数据库并且会返回到之前的窗口以回顾你的修改内容或配置更深层次的设置。

“保存并完成”将存储当前配置级别到数据库并且会返回到配置层次之前的窗口（到上一个配置级别）。

“取消”将丢弃当前配置级别的任何配置修改内容，并且会返回到配置层次之前的窗口（到上一个配置级别）。

### 11.4.4.7.1. WEB服务提供都网络传输

将来可用的网络传输列表可能会增加，目前只有“HTTP::SOAP”和“HTTP::REST”传输可用。每个传输有不同的配置选项，并且可能使用不同的前端界面来配置它们。

配置“HTTP::SOAP”作为提供者很简单。只有两个设置“命名空间”和“消息的最大长度”。这些字段是必填的，第一个是一个URI（统一资源标识符），为SOAP方法指定一个上下文，减少歧义；第二个是你指定OTRS能够处理SOAP消息的最大尺寸（单位：字节）。

图 4.107. WEB服务提供者网络传输 ( HTTP::SOAP )

GenericInterface Transport HTTP::SOAP for Web Service Webservice one

You are here: > Web Services > Webservice one > Provider Transport HTTP::SOAP

Actions

Go back to web service

Network transport

Properties

Type: HTTP::SOAP

\* Namespace:   
 URI to give SOAP methods a context, reducing ambiguities.  
 e.g um:otrs-com:soap:functions or http://www.otrs.com/GenericInterface/actions

\* Request name scheme:   
 Select how SOAP request function wrapper should be constructed.  
 'FunctionName' is used as example for actual invoker/operation name.  
 'FreeText' is used as example for actual configured value.

\* Response name scheme:   
 Select how SOAP response function wrapper should be constructed.  
 'FunctionName' is used as example for actual invoker/operation name.  
 'FreeText' is used as example for actual configured value.

\* Maximum message length:   
 Here you can specify the maximum size (in bytes) of SOAP messages that OTRS will process.

Sort options:

Add new first level element:  Add

Outbound sort order for xml fields (structure starting below function name wrapper) - see documentation for SOAP transport.

Save or Save and finish or Cancel

对于“HTTP::REST”，配置可能要稍微复杂一些，因为它的设置会根据配置的操作而动态增长，给每个操作添加两个设置：“路由映射，用于操作 ‘<操作名称>.’和“有效的请求方法，用于操作 ‘<操作名称 <.’到默认的传输设置“消息的最大长度：”和“发送Keep-Alive（保持连接）：”之前。

- 路由映射，用于操作 ‘<操作名称>’:

在这个设置中设置一个资源路径，这个路径必须根据WEB服务考虑的需求来定义，这个路径与HTTP请求方法相结合来确定要执行的通用接口操作。

路径中可以包含变量，格式为‘<VariableName>’。在变量位置的每个路径字符串会用这个设置中定义的变量名称替换后添加到请求的有效载荷中。

例子：

路由映射：/Resource

- 有效的请求：

http://localhost/otrs/nph-genericinterface.pl/Webservice/Test/Resource

http://localhost/otrs/nph-genericinterface.pl/Webservice/Test/Resource?  
Param1=One

- 无效的请求：

http://localhost/otrs/nph-genericinterface.pl/Webservice/Test/Resource/

http://localhost/otrs/nph-genericinterface.pl/Webservice/Test/Resource/  
OtherResource

http://localhost/otrs/nph-genericinterface.pl/Webservice/Test/Resource/  
OtherResource?Param1=One

路由映射：/Resource/:ID

- 有效的请求：

http://localhost/otrs/nph-genericinterface.pl/Webservice/Test/Resource/1

http://localhost/otrs/nph-genericinterface.pl/Webservice/Test/Resource/1?Param1=One

在这两种情况下，ID=1都会发送到操作作为有效载荷的一部分。在这2种情况还添加了Param1=One，取决于HTTP请求方法还会添加其它参数（如果它们在请求头字段中作为一个JSON字符串）。

- 无效的请求：

http://localhost/otrs/nph-genericinterface.pl/Webservice/Test/Resource

http://localhost/otrs/nph-genericinterface.pl/Webservice/Test/Resource?Param1=One

路由映射：/Resource/OtherResource/:ID/:Color

- 有效的请求：

http://localhost/otrs/nph-genericinterface.pl/Webservice/Test/Resource/OtherResource/1/Red

http://localhost/otrs/nph-genericinterface.pl/Webservice/Test/Resource/OtherResource/123/Blue?Param1=One

在第一个例子中，ID = 1且Color = Red，第二个例子中ID = 123且Color = Blue。

- 无效的请求：

http://localhost/otrs/nph-genericinterface.pl/Webservice/Test/Resource/1

http://localhost/otrs/nph-genericinterface.pl/Webservice/Test/Resource/OtherResource/1

http://localhost/otrs/nph-genericinterface.pl/Webservice/Test/Resource/OtherResource/1?Param1=One

在第一个例子中，路径缺失了'/OtherResource'和:Color变量，在第二个例子中只缺失了:Color变量。

- 有效的请求方法，用于操作'<操作名称>'：

HTTP请求方法结合路由映射来确定要使用的操作，可能的选项有：CONNECT、DELETE、GET、HEAD、OPTIONS、PATCH、POST、PUT和TRACE。

完全不同的操作可以共享完全相同的映射路径，但是为了正确地确定使用的操作，每个操作的请求方法必须是唯一的。

**图 4.108. WEB服务提供者网络传输（HTTP::REST）**

GenericInterface Transport HTTP::REST for Web Service Webservice two

You are here: > Web Services > Webservice two > Provider Transport HTTP::REST

Actions

Go back to web service

Network transport

Properties

Type: HTTP::REST

★ Maximum message length: 10000000

Here you can specify the maximum size (in bytes) of REST messages that OTRS will process.

★ Send Keep-Alive: No

This configuration defines if incoming connections should get closed or kept alive.

Save or Save and finish or Cancel

### 11.4.4.7.2. WEB服务操作

当你使用OTRS作为一个WEB服务提供者时可以执行的动作叫“操作”。每个操作属于一个控制器。控制器是操作或调用程序的集合，一般地，同一控制器中的操作需要类似的设置并共享相同的配置对话框，但是如果需要每个操作可以有独立的配置对话框。

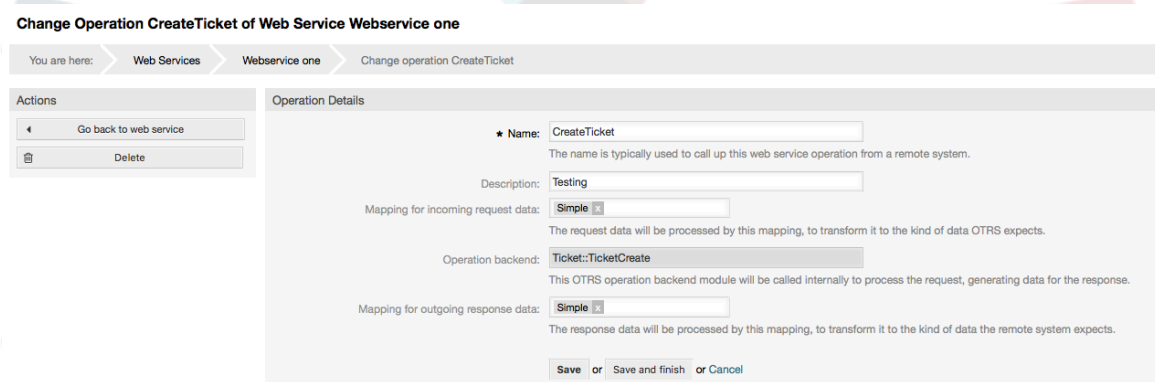
一般出现在操作中的字段是名称、描述、后端和映射，其它特殊字段可以出现在非默认配置对话框中以达到操作的特殊需要。

每个操作一般有两个映射配置部分，一个是进入数据映射，另外一个为出站数据映射。你可以为每个映射方向选择不同的映射类型（后端），因为它们的配置是彼此独立的，并且操作后端也是独立的。通常和最普遍的做法是操作在这两种情况下（反向的配置）使用相同的映射类型。完整的映射配置在一个单独的窗口（取决于映射类型）中完成。

操作后端是预先选择的，不能编辑。如果在WEB服务编辑窗口选择了该操作你就会看到这个参数。这个字段仅用于提供信息。

在窗口左侧的操作栏中你有这些选项：“返回到WEB服务”（丢弃上次保存后的所有修改内容）和“删除”。如果你点击了“删除”，会打开一个对话框问你是否要删除这个操作，点击“删除”按钮确认删除该操作和它的配置，或点击“取消”关闭对话框。

**图 4.109. WEB服务的操作**



### 11.4.4.7.3. WEB服务请求者网络传输

WEB服务请求者的网络传输配置类似于提供者的配置。对于请求者的“HTTP::SOAP”网络传输需要设置更多的字段。

除了“端点”（远程系统上的WEB服务用来接受请求的URI）和“命名空间”是必填字段外，你还要指定：

- SOAP消息的编码（如utf-8、latin1、iso-8859-1、cp1250等等）
- SOAP动作头：你可以使用这个来发送一个空的或填满的SOAPAction头。设置为“否”且SOAP消息中的SOAP动作头会是一个空的字符串，或者设置为“是”以发送‘命名空间’#‘动作’格式的SOAP动作并定义分隔符（通常.NET WEB服务是“/”，REST是“#”）。
- 身份认证：设置身份认证机制，设置为“-”不使用任何身份认证，或从列表中选择一个身份认证后会出现相关的详细字段。

#### 注意

目前只实现了“HTTP基本认证”机制。你可以根据远程系统配置决定是否要用它。如果要使用“HTTP基本认证”，你必须提供用户名和密码来访问远程系统。

## 警告

如果你提供了一个身份认证密码并且随后你导出这个WEB服务到一个YAML文件，这个密码会以明文字符串导出到YAML文件中。请知悉这一点并在需要时采取必要的预防措施。

图 4.110. WEB服务请求者的网络传输（HTTP::SOAP）

GenericInterface Transport HTTP::SOAP for Web Service Webservice one

You are here: > Web Services > Webservice one > Requester Transport HTTP::SOAP

Actions

Go back to web service

Network transport

Properties

Type: HTTP::SOAP

★ Endpoint:   
URI to indicate a specific location for accessing a service.  
 e.g. http://local.otrs.com:8000/Webservice/Example

★ Namespace:   
URI to give SOAP methods a context, reducing ambiguities.  
 e.g. urn:otrs-com:soap:functions or http://www.otrs.com/GenericInterface/actions

★ Request name scheme:   
Select how SOAP request function wrapper should be constructed.  
 'FunctionName' is used as example for actual invoker/operation name.  
 'FreeText' is used as example for actual configured value.

★ Response name scheme:   
Select how SOAP response function wrapper should be constructed.  
 'FunctionName' is used as example for actual invoker/operation name.  
 'FreeText' is used as example for actual configured value.

Encoding:   
The character encoding for the SOAP message contents.  
 e.g. utf-8, latin1, iso-8859-1, cp1250, Etc.

SOAPAction:   
Set to "Yes" to send a filled SOAPAction header.  
 Set to "No" to send an empty SOAPAction header.

SOAPAction separator:   
Character to use as separator between name space and SOAP method.  
 Usually .Net web services uses a "/" as separator.

Authentication:   
The authentication mechanism to access the remote system.  
 A "\*" value means no authentication.

Proxy Server:   
URI of a proxy server to be used (if needed).  
 e.g. http://proxy\_hostname:8080

Proxy User:   
The user name to be used to access the proxy server.

Proxy Password:   
The password for the proxy user.

Use SSL Options:   
Show or hide SSL options to connect to the remote system.

Sort options:    
Outbound sort order for xml fields (structure starting below function name wrapper) - see documentation for SOAP transport.

or

至于HTTP::REST，这个配置也会根据配置的调用程序动态增长，给每个调用程序添加两个设置：“控制器映射，用于调用程序 ‘<调用程序名称>:’和“有效的请求命令，用于调用程序 ‘<调用程序名称>:’”。身份认证和SSL选项类似于HTTP::SOAP。

- 主机

远程系统的主机名称或IP地址和端口号，如果没有端口号，默认使用80端口。

- 控制器映射，用于调用程序<调用程序名称>:

在这个设置中设置一个资源路径，这个路径必须根据远程WEB服务的需求来定义，并遵从它的规定。

路径可以包含‘<变量名称>’格式的变量，每个变量名称匹配（要发送的）当前数据将被对应的数据值替换。匹配的变量名称和值将从当前数据中移除。取决于HTTP请求命令，剩余的数据可以在请求正文中作为一个JSON字符串或在URI中作为查询参数发送。

例子：

当前数据 : Var1 = One, Var2 = Two, Var3 = Three and Var4 = Four。

控制器映射 : /Resource

- 替换后 :

/Resource

- 剩余数据 :

Var1 = One, Var2 = Two, Var3 = Three and Var4 = Four

控制器映射 : /Resource/:Var1

- 替换后 :

/Resource/One

- 剩余数据 :

Var2 = Two, Var3 = Three and Var4 = Four

控制器映射 : /Resource/:Var1?Param1=:Var2&Var3=:Var3

- 替换后 :

/Resource/One?Param1=Two&Var3=Three

- 剩余数据 :

Var4 = Four

- 有效的请求命令 , 用于调用程序 '<调用程序名称>' :

这个字段确定使用的HTTP请求方法 , 可能的选项有 :

CONNECT、DELETE、GET、HEAD、OPTIONS、PATCH、POST、PUT和TRACE。如果没有选择命令 , 则使用默认命令。

- 默认命令

用作所有没有定义请求命令的调用程序的最后依靠。



图 4.111. WEB服务请求者网络传输 (HTTP::REST)

GenericInterface Transport HTTP::REST for Web Service Webservice one

You are here: Web Services > Webservice one > Requester Transport HTTP::REST

Actions

Go back to web service

Network transport

Properties

Type: HTTP::REST

\* Host:   
 Remote host URL for the REST requests.  
 e.g https://www.otrs.com:10745/api/v1.0 (without trailing backslash)

\* Controller mapping for Invoker 'InvokerOne':   
 The controller that the invoker should send requests to. Variables marked by a ':' will get replaced by the data value and passed along with the request. (e.g. /Ticket/:TicketID?UserLogin=:UserLogin&Password=:Password).

Valid request command for Invoker 'InvokerOne':   
 A specific HTTP command to use for the requests with this Invoker (optional).

Default command:   
 The default HTTP command to use for the requests.

Authentication:   
 The authentication mechanism to access the remote system.  
 A "-" value means no authentication.

Use SSL Options:   
 Show or hide SSL options to connect to the remote system.

or  or

#### 11.4.4.7.4. WEB服务调用程序

当你使用OTRS作为一个WEB服务请求者时能够执行的操作称之为“调用程序”。每个调用程序属于一个控制器（控制器是操作或调用程序的集合）。通常在相同控制器中的调用程序需要类似的设置和共同相同的配置对话框。每个调用程序如果需要可以有独立的配置对话框。

在每个调用程序配置中通常是名称、描述、后端和映射字段，附加的事件触发器和其它特殊字段的列表可以出现在非默认的配置对话框中以满足调用程序特殊的需求。

每个调用程序通常有两个映射配置部分，一个是对进入的数据，另外一个是对发出的数据。你可以给每个映射方向选择不同的映射类型（后端），因为它们的配置是彼此独立的，调用程序后端也是独立的。通常和最普遍的做法是调用程序在这两种情况下（反射的配置）使用相同的映射类型。完整的映射配置在一个单独的窗口（取决于映射类型）中完成。

操作后端是预先选择的，不能编辑。如果在WEB服务编辑窗口选择了该调用程序你就会看到这个参数。这个字段仅用于提供信息。

事件触发器是OTRS中的事件比如“工单创建”、“信件发送”等等。这些事件可以作为执行调用程序的触发器。每个调用程序需要注册至少一个事件触发器，否则调用程序就没用，因为它永远不会被调用。事件触发器的异步属性定义是由OTRS进程处理这个调用程序还是委托给OTRS守护进程处理。

### 注意

OTRS守护进程是一个执行后台任务的单独的进程集，使用守护进程就不会影响OTRS进程本身，比如服务不可用或有网络问题，远程系统花费大量时间才响应。如果你不使用OTRS守护进程，则使用WEB服务会导致OTRS变慢或无法响应。所以强烈推荐尽可能使用异步事件触发器。

要添加一个事件触发器，首先从第一个列表中选择事件家族，然后从第二个列表中选择事件名称，然后设置异步属性（未选中即意味着事件触发器不是异步的），最后点击“+”按钮。就会创建一个新的事件触发器，会在调用程序的“事件触发器”列表中列出。

要删除一个事件触发器，简单定位到“事件触发器”列表中要删除的事件触发器，点击行尾的“垃圾桶”图标即可。这会打开一个对话框问你是否确认删除该事件触发器，点击“删除”从列表中删除这个事件触发器，或“取消”关闭对话框。

在窗口左侧操作栏你有两个选项：“返回到WEB服务”（丢失上次保存以来的所有修改内容）和“删除”。如果你点击了“删除”按钮，会弹出一个对话框问你是否要删除这个调用程序，点击“删除”删除这个调用程序和它的配置，或“取消”关闭删除对话框。

**图 4.112. WEB服务调用程序**

Change Invoker InvokerOne of Web Service Webservice one

You are here: Web Services > Webservice one > Change invoker InvokerOne

Actions

Go back to web service

Delete

Invoker Details

**Name:** InvokerOne  
The name is typically used to call up an operation of a remote web service.

**Description:** User for remote ticket creation

**Invoker backend:** Ticket::TicketCreate  
This OTRS invoker backend module will be called to prepare the data to be sent to the remote system, and to process its response data.

**Mapping for outgoing request data:** Simple   
The data from the invoker of OTRS will be processed by this mapping, to transform it to the kind of data the remote system expects.

**Mapping for incoming response data:** XSLT   
The response data will be processed by this mapping, to transform it to the kind of data the invoker of OTRS expects.

**Event Triggers:**

EVENT	ASYNCHRONOUS	DELETE
TicketCreate	Yes	<input type="button" value=""/>

This invoker will be triggered by the configured events.

**Add Event Trigger:** Ticket  EscalationResponseTimeNotifyBefore   Asynchronous

To add a new event select the event object and event name and click on the "+" button.  
 Asynchronous event triggers are handled by the OTRS Scheduler Daemon in background (recommended).  
 Synchronous event triggers would be processed directly during the web request.

or  or

#### 11.4.4.7.5. WEB服务映射

有这种情况：你需要将一种格式的数据转化为另外的格式（映射或修改数据结构），因为通常一个WEB服务用来与一个远程系统交互，很可能不是另外一个OTRS系统并且/或者不能理解OTRS数据结构和值。在这些情况下一些或所有值必须被修改，有些时候甚至值的名称（键）或整个结构都需要修改，才能匹配另外一端期望的数据。

每个远程系统都有它自己的数据结构，可以为每种情况创建新的映射模块（例如：OTRS自带有一个为SAP Solution Manager定制的映射模块），但并不总是需要这样单独定制，Mapping::Simple模块可以涵盖大部分的映射需求。

### 注意

当Mapping::Simple无法处理一个WEB服务所有的映射需求时，就需要创建一个映射模块。要学习关于如何创建一个新的映射模块的更多内容，请参阅OTRS开发手册。

这个模块让你有机会设置默认值来映射整个通讯数据中的每个键或值。

在窗口的开始部分你会看到一个通用部分，在这里你可以设置默认规则，适用于所有没有映射的键和值。还有3个选项可用，如下列表：

- 保持（保持不变）：不对键或值做任何改动。
- 忽略（丢弃键/值对）：当这个选项用于键时它会删除键和值，因为当一个键被删除时结果就是键相关的值也被删除。当这个选项用于值时，只有这个值被删除，保留键（现在会关联一个空值）。
- 映射到（使用提供键或值为默认）：所有没有定义映射规则的键和/或值，使用这个映射作为默认映射，当你选择这个选项后会出现一个新的文本字段，用来设置默认键或值。

点击“+”按钮来添加新的键映射，会显示一个新框用于一个特定映射配置。你可以根据需要的随意添加键映射。只需再次点击“+”按钮，在已有映射的下面会出现一个新的映射框。从这个映射你可以定义一个特定键的映射，有下列选项：

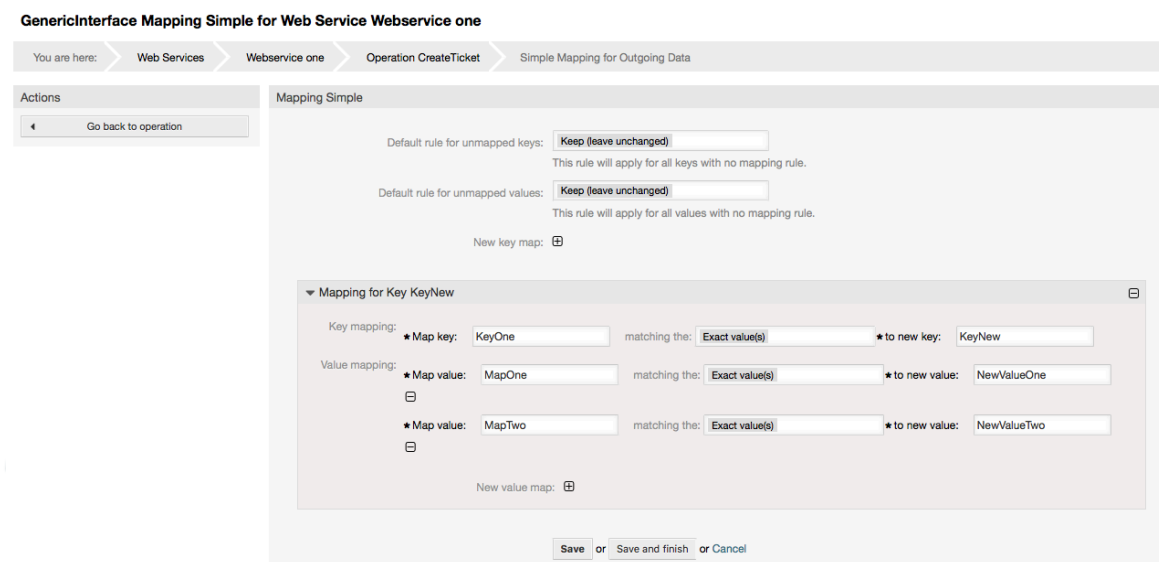
- 准确值：如果原键完全匹配，原键字符串会变为一个新的键。
- 正则表达式：键字符串会按照一个正则表达式规则进行替换。

点击新增值映射的“+”按钮，会显示一个新行用于值映射。在这里也有键映射那样的选项（准确值和正则表达式）来定义每个值映射。你可以根据需要随意添加值映射，如果你要删除一个值映射，只需点击每个值映射行后面的“-”按钮即可。

可以删除完整的键映射部分（框），只需点击要删除的键映射框右上角的“-”按钮即可。

如果你要删除一个完整的映射配置，返回到相应的操作或调用程序窗口，找到之前选择的映射方向并设置其值为“-”，然后保存配置即可。

**图 4.113. WEB服务映射**



The screenshot shows the 'GenericInterface Mapping Simple for Web Service Webservice one' configuration page. It features a breadcrumb trail: 'You are here: > Web Services > Webservice one > Operation CreateTicket > Simple Mapping for Outgoing Data'. Below this is an 'Actions' section with a 'Go back to operation' button. The main 'Mapping Simple' section contains two default rules: 'Default rule for unmapped keys: Keep (leave unchanged)' and 'Default rule for unmapped values: Keep (leave unchanged)'. A 'New key map:' field is present. The 'Mapping for Key KeyNew' section is expanded, showing 'Key mapping' with 'Map key: KeyOne' and 'to new key: KeyNew', and 'Value mapping' with two entries: 'Map value: MapOne' to 'NewValueOne' and 'Map value: MapTwo' to 'NewValueTwo'. A 'New value map:' field is also visible. At the bottom, there are 'Save', 'Save and finish', and 'Cancel' buttons.

## 11.5. WEB服务命令行接口

bin/otrs.Console.pl Admin::WebService::\*系列命令被设计为一组基本但快速、强大的工具以处理WEB服务配置。它们让你能够执行下列操作：

- Add（添加）：使用一个YAML文件作为配置源来创建WEB服务。
- Update（更新）：修改一个已有的WEB服务，使用一个不同的或修改过的YAML文件来修改WEB服务的配置。
- Dump（转储）：将当前的WEB服务的配置保存到一个文件。
- List（列表）：获取当前系统中注册的所有WEB服务的完整清单。
- Delete（删除）：从系统中删除一个WEB服务。使用它的时候要小心，因为这个操作无法复原。

例子：创建一个新的WEB服务配置：

```
shell> bin/otrs.Console.pl Admin::WebService::Add --name <webservice_name> --source-path /
path/to/yaml/file
```

## 11.6. WEB服务配置

从WEB服务的设计来说，它主要考虑从一个OTRS系统到另外一个OTRS系统的可移植性，例如从一个测试或开发环境移植到生产系统。所以需要有一种容易的方式来从数据库提取WEB服务配置，并将它导入到另外一个系统。为完成这个任务，通用接口使用YAML文件作为WEB服务的配置基础。

为什么是YAML？YAML是一种标记语言，设计为人类友好的读和写（它比JSON更容易理解）。它没有XML的一些限制如数字标签，它是开放的和标准化的，并且完全足够用来存储整个WEB服务配置。

## 注意

要学习关于YAML的更多内容，请访问<http://www.yaml.org/>。

下面是一个YAML格式的WEB服务配置文件例子：

```
---
Debugger:
  DebugThreshold: debug
Description: This an example of a web service configuration
Provider:
  Operation:
    CloseIncident:
      Description: This is a test operation
      MappingInbound: {}
      MappingOutbound: {}
      RemoteSystemGuid: ''
      Type: Test::Test
    Test:
      Description: This is a test operation
      MappingInbound:
        Config:
          KeyMapDefault:
            MapTo: ''
            MapType: Keep
          KeyMapExact:
            Prio: Priority
          ValueMap:
            Priority:
              ValueMapExact:
                Critical: 5 Very High
                Information: 1 Very Low
                Warning: 3 Normal
            ValueMapDefault:
              MapTo: 3 Normal
              MapType: MapTo
          Type: Simple
        MappingOutbound:
          Config:
            KeyMapDefault:
              MapTo: ''
              MapType: Ignore
            KeyMapExact:
              Priority: Prio
          ValueMap:
            Prio:
              ValueMapExact:
                1 Very Low: Information
                3 Normal: Warning
                5 Very High: Critical
            ValueMapDefault:
              MapTo: ''
              MapType: Ignore
          Type: Simple
        Type: Test::Test
      Transport:
        Config:
          MaxLength: 10000000
          NameSpace: http://www.example.com/actions
          Type: HTTP::SOAP
      RemoteSystem: remote.system.description.example.com
  Requester:
    Invoker:
      Test:
        Description: This is a test invoker
        Events:
          - Asynchronous: 1
            Event: TicketCreate
          - Asynchronous: 0
            Event: ArticleUpdate
```

```
MappingInbound:
  Type: Simple
MappingOutbound:
  Type: Simple
Type: Test::Test
Transport:
  Config:
    Authentication:
      Password: '*****'
      Type: BasicAuth
      User: otrs
    Encoding: utf-8
    Endpoint: http://www.example.com:8080/endpoint
    NameSpace: http://www.example.com/actions
    SOAPAction: Yes
    SOAPActionSeparator: '#'
  Type: HTTP::SOAP
```

## 11.6.1. 配置详细信息

### 11.6.1.1. 一般

- Description (描述) : 描述WEB服务的简短文本。
- RemoteSystem (远程系统) : 远程系统的简短描述。
- Debugger (调试器) : 调试器设置的一个容器。
- Provider (提供者) : 提供者设置的一个容器。
- Requester (请求者) : 请求者设置的一个容器。

### 11.6.1.2. 调试器

- 调试阈值 : 调试器的级别。

#### 可能的值

- debug (调试) : 所有日志都存储到数据库。
- info (信息) : info (信息)、notice (注意) 和error (错误) 级别的日志存储到数据库。
- notice (注意) : notice (注意) 和error (错误) 级别的日志存储到数据库。
- error (错误) : 只有error (错误) 级别的日志存储到数据库。

### 11.6.1.3. 提供者

- Operation (操作) : 每个操作设置的一个容器。
- Transport (传输) : 提供者网络传输设置的一个容器。

#### 11.6.1.3.1. 操作

- <OperationName操作名称> : 操作的唯一名称, 它自己的操作设置的容器 (可以加基数0..n, 但不能重复)。

##### 11.6.1.3.1.1. <OperationName>

这部分基于来自“Test::Test”类型的操作, 其它操作可能包含更多或不同的设置。

- Description (描述) : 描述这个操作的简短文本。
- MappingInbound (入站映射) : 进入的请求数据的映射设置容器。

- MappingOutbound (出站映射) : 外发的响应数据的映射设置容器。
- Type (类型) : 操作后端, 格式为Controller::Operation。

#### 11.6.1.3.1.1.1. MappingInbound (入站映射)

本部分内容基于映射类型“Simple (简单)”, 其它映射类型可能包含更多或不同的设置。

- Config (配置) : 这个映射设置的容器。
- Type (类型) : 映射后端。

#### 11.6.1.3.1.1.1.1. 配置

- KeyMapDefault (默认的键映射) : 所有未映射的键的设置的容器。
- ValueMapDefault (默认的值映射) : 所有未映射的值的设置的容器。
- KeyMapExact (准确的键映射) : 所有准确的键映射 (基数0 .. 1) 的容器。
- KeyMapRegEx (正则键映射) : 所有正则表达式的键映射 (基数0 .. 1) 的容器。
- ValueMap (值映射) : 所有值映射 (基数0 .. 1) 的容器。

#### 11.6.1.3.1.1.1.1.1. KeyMapDefault (默认的键映射)

- MapTo (映射到) : 使用新的值 (只适用于MapTo映射类型)。
- MapType (映射类型) : 映射的规则。

##### 可能的值

- Keep (保持) : 保持不变
- Ignore (忽略) : 丢弃。
- MapTo (映射到) : 修改为MapTo的值。

#### 11.6.1.3.1.1.1.1.2. ValueMapDefault (默认的值映射)

与KeyMapDefault (默认的键映射) 类似。

#### 11.6.1.3.1.1.1.1.3. KeyMapExact (准确的键映射)

- <oldkey原键>: <newkey新键> (基数 0 .. n, 但不能重复)。

#### 11.6.1.3.1.1.1.1.4. KeyMapRegEx (正则表达式键映射)

- <oldkey(RegEx)匹配正则表达式的原键>: <newkey新键> (基数 0 .. n, 但不能重复)。

#### 11.6.1.3.1.1.1.1.5. ValueMap (值映射)

- <newkey (新键)> : 这个新键的值映射容器 (基数取决于来自KeyMapExact和KeyMapRegEx的新键)。

#### 11.6.1.3.1.1.1.1.5.1. <newkey (新键)>

- : 所有准确的价值映射容器 (基数 0 .. 1)。
- ValueMapRegEx (正则表达式的值映射) : 所有的正则表达式的值映射容器 (基数 0 .. 1)。

#### 11.6.1.3.1.1.1.1.5.1.1. ValueMapExact (准确的价值映射)

- <oldvalue原值>: <newvalue新值> (基数 0 .. n但不能重复)。

#### 11.6.1.3.1.1.1.1.5.1.2. ValueMapRegEx (正则表达式的值映射)

- <oldvalue(RegEx)匹配正则表达式的原值>: <newvalue新值> (基数 0 .. n, 但不能重复)。

### 11.6.1.3.1.1.2. MappingOutbound (出站映射)

与MappingInbound (进站映射) 相同。

### 11.6.1.3.1.1.3. 传输

本部分基于提供者的网络传输HTTP::SOAP，其它的传输可能包含更多或不同的设置。

- Config (配置)：这个特定的网络传输配置的容器。
- Type (类型)：提供者网络传输后端。

### 11.6.1.3.1.1.3.1. 配置

- MaxLength (最大长度)：OTRS中一个SOAP消息能够读取的最大长度 (单位：字节)。
- NameSpace (命名空间)：一个URI，给定一个属于这个WEB服务所有操作的上下文环境。

### 11.6.1.4. Requester (请求者)

- Invoker (调用程序)：每个调用程序的设置的容器。
- Transport (传输)：请求者网络传输设置的容器。

### 11.6.1.4.1. 调用程序

- <InvokerName调用程序名称>：该调用程序的唯一名称，它自己的调用程序设置的容器 (基数 0 .. n，但不能重复)。

### 11.6.1.4.1.1. <InvokerName调用程序名称>

本部分内容基于调用程序类型“Test::Test”，其它调用程序类型可能包含更多或不同的设置。

- Description (描述)：描述这个调用程序的简短文本。
- Events (事件)：事件触发器的未命名列表的容器。
- MappingInbound (进站映射)：进入的响应数据的映射设置容器。
- MappingOutbound (出站映射)：外发的请求数据的映射设置容器。
- Type (类型)：Controller (控制器)::Invoker (调用程序) 格式的调用程序后端。

### 11.6.1.4.1.1.1. Events (事件)

- 列表元素：(基数 0 .. n)。
  - 异步：设置是否委派给OTRS守护进程执行调用程序。

#### 可能的值

- 0：不由OTRS守护进程执行调用程序。
- 1：由OTRS守护进程执行调用程序。
- Event (事件)：事件触发器的名称。

#### 可能的值 (用于工单事件)

- 工单创建
- TicketDelete (删除工单)
- TicketTitleUpdate (更新工单标题)
- TicketUnlockTimeoutUpdate (更新工单解锁超时)

- TicketQueueUpdate (更新工单队列)
- TicketTypeUpdate (更新工单类型)
- TicketServiceUpdate (更新工单服务)
- TicketSLAUpdate (更新工单SLA)
- TicketCustomerUpdate (更新工单客户联系人)
- TicketPendingTimeUpdate (更新工单挂起时间)
- TicketLockUpdate (更新工单锁定)
- TicketArchiveFlagUpdate (更新工单归档标识)
- TicketStateUpdate (更新工单状态)
- TicketOwnerUpdate (更新工单所有者)
- TicketResponsibleUpdate (更新工单负责人)
- TicketPriorityUpdate (更新工单优先级)
- HistoryAdd (添加历史)
- HistoryDelete (删除历史)
- TicketAccountTime (工单占用时间)
- TicketMerge (合并工单)
- TicketSubscribe (关注工单)
- TicketUnsubscribe (取消关注工单)
- TicketFlagSet (设置工单标识)
- TicketFlagDelete (删除工单标识)
- TicketSlaveLinkAdd (添加'从工单'链接)
- TicketSlaveLinkDelete (删除'从工单'链接)
- TicketMasterLinkDelete (删除'主工单'链接)

#### 可能的值 (用于信件事件)

- ArticleCreate (创建信件)
- ArticleUpdate (更新信件)
- ArticleSend (发送信件)
- ArticleBounce (退回信件)
- ArticleAgentNotification (服务人员通知信件)
- ArticleCustomerNotification (客户通知信件)
- ArticleAutoResponse (自动响应信件)



- ArticleFlagSet ( 设置信件标识 )
- ArticleFlagDelete ( 删除信件标识 )

#### **11.6.1.4.1.1.2. MappingInbound ( 进站映射 )**

与操作的MappingInbound ( 进站映射 ) 相同。

#### **11.6.1.4.1.1.3. MappingOutbound ( 出站映射 )**

与操作的MappingInbound ( 进站映射 ) 相同。

#### **11.6.1.4.1.1.4. 传输**

本部分基于请求者的网络传输HTTP::SOAP，其它的传输可能包含更多或不同的设置。

- Config ( 配置 ) : 这个特定的网络传输配置的容器。
- Type ( 类型 ) : 请求者的网络传输后端。

##### **11.6.1.4.1.1.4.1. 配置**

- Authentication ( 身份认证 ) : 身份认证设置的容器。
- Encoding ( 编码 ) : 需要将SOAP消息编码。
- Endpoint ( 端点 ) : 远程服务器WEB服务的URI，用来接受OTRS的请求。
- NameSpace ( 命名空间 ) : 一个URI，给定一个属于这个WEB服务的所有调用程序的上下文环境。
- SOAPAction ( SOAP动作 ) : 在SOAP消息中发送一个空的或填满的SOAPAction头 ( 用“<NameSpace> <Separator> <Action>”格式 )。

##### 可能的值

- Yes ( 是 ) : 发送一个填满的SOAPAction头。
- No ( 否 ) : 发送一个空的SOAPAction头。
- SOAPActionSeparator ( SOAP动作分隔符 ) : 设置填满的SOAP动作头的<分隔符>。

##### 可能的值

- '/' : 用于.NET WEB服务。
- '#' : 用于所有其它的WEB服务。

##### **11.6.1.4.1.1.4.1.1. Authentication ( 身份验证 )**

- User ( 用户名 ) : 有权访问远程WEB服务的用户名。
- Password ( 密码 ) : 特权用户的纯文本密码。
- Type ( 类型 ) : 身份验证的类型。

## **11.7. 连接器**

连接器本质上是动作 ( 在OTRS作为WEB服务提供者时叫操作，在OTRS作为WEB服务请求者时啊调用程序 ) 的集合。但是它也可以包含特殊的映射或传输。

一个连接器可以只有操作、只有调用程序或者两者都有。一个连接器甚至可以使用其它连接器的一部分如映射或传输（如果它们不是部署为特定的连接器的话）。

换句话说，连接器不局限于控制器层，在需要时还能扩展到数据映射或网络传输层。

由于通用接口的模块化设计，连接器可以看作是一个插件；这意味着通过添加连接器，通用接口的能力可以扩展使用在：OTRS功能插件、OTRS定制模块、第三方模块等等。

## 11.7.1. 捆绑的连接器

在当前版本的OTRS中有下列连接器可以使用：

- 会话
- 工单

### 11.7.1.1. 会话连接器

这个连接器有能力创建一个有效的会话ID，能够用在其它任何操作中。

提供：

- 操作：
  - 创建会话

#### 11.7.1.1.1. 操作

##### 11.7.1.1.1.1. 创建会话

创建一个新的有效会话ID，用在其它连接器比如创建工单或其它操作中。

### 注意

要在其它连接器的其它操作中使用这个会话ID，需要该操作通过会话ID部署身份认证。所有剩下的捆绑操作都有能力接受一个有效的会话ID作为一个身份认证方法。

可能的属性：

```
<SessionCreate>
  <!--You have a MANDATORY CHOICE of the next 2 items at this level-->
  <!--Optional:-->
  <UserLogin>?</UserLogin>
  <!--Optional:-->
  <CustomerUserLogin>?</CustomerUserLogin>
  <!--Optional:-->
  <Password>?</Password>
</SessionCreate>
```

### 11.7.1.2. 工单连接器

这个连接器提供与工单交互的基本功能。

提供：

- 操作：

- 工单创建
- 更新工单
- 获取工单
- 搜索工单

### 11.7.1.2.1. 操作

#### 11.7.1.2.1.1. 工单创建

提供一个接口以在OTRS中创建一个工单。一个工单必须包含一个信件并包含数个附件，所有定义的动态字段也可以在创建工单操作时设置。

可能的属性：

```
<TicketCreate>
  <!--You have a MANDATORY CHOICE of the next 3 items at this level-->
  <!--Optional:-->
  <UserLogin?</UserLogin>
  <!--Optional:-->
  <CustomerUserLogin?</CustomerUserLogin>
  <!--Optional:-->
  <SessionID?</SessionID>
  <!--Optional:-->
  <Password?</Password>
  <Ticket>
    <Title?</Title>
    <!--You have a MANDATORY CHOICE of the next 2 items at this level-->
    <!--Optional:-->
    <QueueID?</QueueID>
    <!--Optional:-->
    <Queue?</Queue>
    <!--You have a CHOICE of the next 2 items at this level-->
    <!--Optional:-->
    <TypeID?</TypeID>
    <!--Optional:-->
    <Type?</Type>
    <!--You have a CHOICE of the next 2 items at this level-->
    <!--Optional:-->
    <ServiceID?</ServiceID>
    <!--Optional:-->
    <Service?</Service>
    <!--You have a CHOICE of the next 2 items at this level-->
    <!--Optional:-->
    <SLAID?</SLAID>
    <!--Optional:-->
    <SLA?</SLA>
    <!--You have a MANDATORY CHOICE of the next 2 items at this level-->
    <!--Optional:-->
    <StateID?</StateID>
    <!--Optional:-->
    <State?</State>
    <!--You have a MANDATORY CHOICE of the next 2 items at this level-->
    <!--Optional:-->
    <PriorityID?</PriorityID>
    <!--Optional:-->
    <Priority?</Priority>
    <!--You have a CHOICE of the next 2 items at this level-->
    <!--Optional:-->
    <OwnerID?</OwnerID>
    <!--Optional:-->
    <Owner?</Owner>
    <!--You have a CHOICE of the next 2 items at this level-->
    <!--Optional:-->
    <ResponsibleID?</ResponsibleID>
    <!--Optional:-->
```

```

<Responsible?</Responsible>
<CustomerUser?</CustomerUser>
<!--Optional:-->
<CustomerID?</CustomerID>
<!--Optional:-->
<PendingTime>
  <!--You have a CHOICE of the next and the other 5 items at this level-->
  <Diff?</Diff>
  <Year?</Year>
  <Month?</Month>
  <Day?</Day>
  <Hour?</Hour>
  <Minute?</Minute>
</PendingTime>
</Ticket>
<Article>
  <!--You have a CHOICE of the next 2 items at this level-->
  <!--Optional:-->
  <ArticleTypeID?</ArticleTypeID>
  <!--Optional:-->
  <ArticleType?</ArticleType>
  <!--You have a CHOICE of the next 2 items at this level-->
  <!--Optional:-->
  <SenderTypeID?</SenderTypeID>
  <!--Optional:-->
  <SenderType?</SenderType>
  <!--Optional:-->
  <From?</From>
  <Subject?</Subject>
  <Body?</Body>
  <!--You have a CHOICE of the next 2 items at this level-->
  <!--Optional:-->
  <ContentType?</ContentType>
  <Charset?</Charset>
  <MimeType?</MimeType>
  <!--Optional:-->
  <HistoryType?</HistoryType>
  <!--Optional:-->
  <HistoryComment?</HistoryComment>
  <!--Optional:-->
  <AutoResponseType?</AutoResponseType>
  <!--Optional:-->
  <TimeUnit?</TimeUnit>
  <!--Optional:-->
  <NoAgentNotify?</NoAgentNotify>
  <!--Zero or more repetitions:-->
  <ForceNotificationToUserID?</ForceNotificationToUserID>
  <!--Zero or more repetitions:-->
  <ExcludeNotificationToUserID?</ExcludeNotificationToUserID>
  <!--Zero or more repetitions:-->
  <ExcludeMuteNotificationToUserID?</ExcludeMuteNotificationToUserID>
</Article>
<!--Zero or more repetitions:-->
<DynamicField>
  <Name?</Name>
  <!--1 or more repetitions:-->
  <Value?</Value>
</DynamicField>
<!--Zero or more repetitions:-->
<Attachment>
  <Content>cid:61886944659</Content>
  <ContentType?</ContentType>
  <Filename?</Filename>
</Attachment>
</TicketCreate>

```

### 11.7.1.2.1.2. 更新工单

‘更新工单’操作用来修改已有工单的属性或者添加一个新的信件（包括附件和所有定义的工单和信件的动态字段）。

## 注意

没有必要创建一个信件来修改一个工单的属性。

可能的属性：

```
<TicketUpdate>
  <!--You have a MANDATORY CHOICE of the next 3 items at this level-->
  <!--Optional:-->
  <UserLogin?</UserLogin>
  <!--Optional:-->
  <CustomerUserLogin?</CustomerUserLogin>
  <!--Optional:-->
  <SessionID?</SessionID>
  <!--Optional:-->
  <Password?</Password>
  <!--You have a CHOICE of the next 2 items at this level-->
  <TicketID?</TicketID>
  <TicketNumber?</TicketNumber>
  <!--Optional:-->
  <Ticket>
    <!--Optional:-->
    <Title?</Title>
    <!--You have a CHOICE of the next 2 items at this level-->
    <!--Optional:-->
    <QueueID?</QueueID>
    <!--Optional:-->
    <Queue?</Queue>
    <!--You have a CHOICE of the next 2 items at this level-->
    <!--Optional:-->
    <TypeID?</TypeID>
    <!--Optional:-->
    <Type?</Type>
    <!--You have a CHOICE of the next 2 items at this level-->
    <!--Optional:-->
    <ServiceID?</ServiceID>
    <!--Optional:-->
    <Service?</Service>
    <!--You have a CHOICE of the next 2 items at this level-->
    <!--Optional:-->
    <SLAID?</SLAID>
    <!--Optional:-->
    <SLA?</SLA>
    <!--You have a CHOICE of the next 2 items at this level-->
    <!--Optional:-->
    <StateID?</StateID>
    <!--Optional:-->
    <State?</State>
    <!--You have a CHOICE of the next 2 items at this level-->
    <!--Optional:-->
    <PriorityID?</PriorityID>
    <!--Optional:-->
    <Priority?</Priority>
    <!--You have a CHOICE of the next 2 items at this level-->
    <!--Optional:-->
    <OwnerID?</OwnerID>
    <!--Optional:-->
    <Owner?</Owner>
    <!--You have a CHOICE of the next 2 items at this level-->
    <!--Optional:-->
    <ResponsibleID?</ResponsibleID>
    <!--Optional:-->
    <Responsible?</Responsible>
    <!--Optional:-->
    <CustomerUser?</CustomerUser>
    <!--Optional:-->
    <CustomerID?</CustomerID>
    <!--Optional:-->
    <PendingTime>
```

```

        <!--You have a CHOICE of the next and the other 5 items at this level-->
        <Diff?></Diff>
        <Year?></Year>
        <Month?></Month>
        <Day?></Day>
        <Hour?></Hour>
        <Minute?></Minute>
    </PendingTime>
</Ticket>
<!--Optional:-->
<Article>
    <!--You have a CHOICE of the next 2 items at this level-->
    <!--Optional:-->
    <ArticleTypeID?></ArticleTypeID>
    <!--Optional:-->
    <ArticleType?></ArticleType>
    <!--You have a CHOICE of the next 2 items at this level-->
    <!--Optional:-->
    <SenderTypeID?></SenderTypeID>
    <!--Optional:-->
    <SenderType?></SenderType>
    <!--Optional:-->
    <From?></From>
    <Subject?></Subject>
    <Body?></Body>
    <!--You have a CHOICE of the next 2 items at this level-->
    <!--Optional:-->
    <ContentType?></ContentType>
    <Charset?></Charset>
    <MimeType?></MimeType>
    <!--Optional:-->
    <HistoryType?></HistoryType>
    <!--Optional:-->
    <HistoryComment?></HistoryComment>
    <!--Optional:-->
    <AutoResponseType?></AutoResponseType>
    <!--Optional:-->
    <TimeUnit?></TimeUnit>
    <!--Optional:-->
    <NoAgentNotify?></NoAgentNotify>
    <!--Zero or more repetitions:-->
    <ForceNotificationToUserID?></ForceNotificationToUserID>
    <!--Zero or more repetitions:-->
    <ExcludeNotificationToUserID?></ExcludeNotificationToUserID>
    <!--Zero or more repetitions:-->
    <ExcludeMuteNotificationToUserID?></ExcludeMuteNotificationToUserID>
</Article>
<!--Zero or more repetitions:-->
<DynamicField>
    <Name?></Name>
    <!--1 or more repetitions:-->
    <Value?></Value>
</DynamicField>
<!--Zero or more repetitions:-->
<Attachment>
    <Content>cid:166861569966</Content>
    <ContentType?></ContentType>
    <Filename?></Filename>
</Attachment>
</TicketUpdate>

```

### 11.7.1.2.1.3. 获取工单

这个操作用来获取一个工单的所有属性，包括动态字段、所有信件和属于工单的所有附件。

可能的属性：

```
<TicketGet>
```

```

<!--You have a MANDATORY CHOICE of the next 3 items at this level-->
<!--Optional:-->
<UserLogin?></UserLogin>
<!--Optional:-->
<CustomerUserLogin?></CustomerUserLogin>
<!--Optional:-->
<SessionID?></SessionID>
<!--Optional:-->
<Password?></Password>
<!--1 or more repetitions:-->
<TicketID?></TicketID>
<!--Optional:-->
<DynamicFields?></DynamicFields>
<!--Optional:-->
<Extended?></Extended>
<!--Optional:-->
<AllArticles?></AllArticles>
<!--Optional:-->
<ArticleSenderType?></ArticleSenderType>
<!--Optional:-->
<ArticleOrder?></ArticleOrder>
<!--Optional:-->
<ArticleLimit?></ArticleLimit>
<!--Optional:-->
<Attachments?></Attachments>
<!--Optional:-->
<HTMLBodyAsAttachment?></HTMLBodyAsAttachment>
</TicketGet>

```

#### 11.7.1.2.1.4. 搜索工单

‘搜索工单’操作返回匹配一个预定义条件的工单编号列表。

可能的属性：

```

<TicketSearch>
<!--You have a MANDATORY CHOICE of the next 3 items at this level-->
<!--Optional:-->
<UserLogin?></UserLogin>
<!--Optional:-->
<CustomerUserLogin?></CustomerUserLogin>
<!--Optional:-->
<SessionID?></SessionID>
<!--Optional:-->
<Password?></Password>
<!--Optional:-->
<Limit?></Limit>
<!--Zero or more repetitions:-->
<TicketNumber?></TicketNumber>
<!--Zero or more repetitions:-->
<Title?></Title>
<!--Zero or more repetitions:-->
<Queues?></Queues>
<!--Zero or more repetitions:-->
<QueueIDs?></QueueIDs>
<!--Optional:-->
<UseSubQueues?></UseSubQueues>
<!--Zero or more repetitions:-->
<Types?></Types>
<!--Zero or more repetitions:-->
<TypeIDs?></TypeIDs>
<!--Zero or more repetitions:-->
<States?></States>
<!--Zero or more repetitions:-->
<StateIDs?></StateIDs>
<!--Zero or more repetitions:-->
<StateType?></StateType>
<!--Zero or more repetitions:-->

```

```

<StateTypeIDs>?</StateTypeIDs>
<!--Zero or more repetitions:-->
<Priorities>?</Priorities>
<!--Zero or more repetitions:-->
<PriorityIDs>?</PriorityIDs>
<!--Zero or more repetitions:-->
<Services>?</Services>
<!--Zero or more repetitions:-->
<ServiceIDs>?</ServiceIDs>
<!--Zero or more repetitions:-->
<SLAs>?</SLAs>
<!--Zero or more repetitions:-->
<SLAIDs>?</SLAIDs>
<!--Zero or more repetitions:-->
<Locks>?</Locks>
<!--Zero or more repetitions:-->
<LockIDs>?</LockIDs>
<!--Zero or more repetitions:-->
<OwnerIDs>?</OwnerIDs>
<!--Zero or more repetitions:-->
<ResponsibleIDs>?</ResponsibleIDs>
<!--Zero or more repetitions:-->
<WatchUserIDs>?</WatchUserIDs>
<!--Zero or more repetitions:-->
<CustomerID>?</CustomerID>
<!--Zero or more repetitions:-->
<CustomerUserLogin>?</CustomerUserLogin>
<!--Zero or more repetitions:-->
<CreatedUserIDs>?</CreatedUserIDs>
<!--Zero or more repetitions:-->
<CreatedTypes>?</CreatedTypes>
<!--Zero or more repetitions:-->
<CreatedTypeIDs>?</CreatedTypeIDs>
<!--Zero or more repetitions:-->
<CreatedPriorities>?</CreatedPriorities>
<!--Zero or more repetitions:-->
<CreatedPriorityIDs>?</CreatedPriorityIDs>
<!--Zero or more repetitions:-->
<CreatedStates>?</CreatedStates>
<!--Zero or more repetitions:-->
<CreatedStateIDs>?</CreatedStateIDs>
<!--Zero or more repetitions:-->
<CreatedQueues>?</CreatedQueues>
<!--Zero or more repetitions:-->
<CreatedQueueIDs>?</CreatedQueueIDs>
<!--Zero or more repetitions:-->
<DynamicFields>
  <!--You have a MANDATORY CHOICE of the next 6 items at this level-->
  <!--Optional:-->
  <Equals>?</Equals>
  <!--Optional:-->
  <Like>?</Like>
  <!--Optional:-->
  <GreaterThan>?</GreaterThan>
  <!--Optional:-->
  <GreaterThanEquals>?</GreaterThanEquals>
  <!--Optional:-->
  <SmallerThan>?</SmallerThan>
  <!--Optional:-->
  <SmallerThanEquals>?</SmallerThanEquals>
</DynamicFields>
<!--Optional:-->
<Ticketflag>
  <!--Optional:-->
  <Seen>?</Seen>
</Ticketflag>
<!--Optional:-->
<From>?</From>
<!--Optional:-->
<To>?</To>
<!--Optional:-->
<Cc>?</Cc>

```



```

<!--Optional:-->
<Subject?></Subject>
<!--Optional:-->
<Body?></Body>
<!--Optional:-->
<FullTextIndex?></FullTextIndex>
<!--Optional:-->
<ContentSearch?></ContentSearch>
<!--Optional:-->
<ConditionInline?></ConditionInline>
<!--Optional:-->
<ArticleCreateTimeOlderMinutes?></ArticleCreateTimeOlderMinutes>
<!--Optional:-->
<ArticleCreateTimeNewerMinutes?></ArticleCreateTimeNewerMinutes>
<!--Optional:-->
<ArticleCreateTimeNewerDate?></ArticleCreateTimeNewerDate>
<!--Optional:-->
<ArticleCreateTimeOlderDate?></ArticleCreateTimeOlderDate>
<!--Optional:-->
<TicketCreateTimeOlderMinutes?></TicketCreateTimeOlderMinutes>
<!--Optional:-->
<ATicketCreateTimeNewerMinutes?></ATicketCreateTimeNewerMinutes>
<!--Optional:-->
<TicketCreateTimeNewerDate?></TicketCreateTimeNewerDate>
<!--Optional:-->
<TicketCreateTimeOlderDate?></TicketCreateTimeOlderDate>
<!--Optional:-->
<TicketLastChangeTimeOlderMinutes?></TicketLastChangeTimeOlderMinutes>
<!--Optional:-->
<TicketLastChangeTimeNewerMinutes?></TicketLastChangeTimeNewerMinutes>
<!--Optional:-->
<TicketLastChangeTimeNewerDate?></TicketLastChangeTimeNewerDate>
<!--Optional:-->
<TicketLastChangeTimeOlderDate?></TicketLastChangeTimeOlderDate>
<!--Optional:-->
<TicketChangeTimeOlderMinutes?></TicketChangeTimeOlderMinutes>
<!--Optional:-->
<TicketChangeTimeNewerMinutes?></TicketChangeTimeNewerMinutes>
<!--Optional:-->
<TicketChangeTimeNewerDate?></TicketChangeTimeNewerDate>
<!--Optional:-->
<TicketChangeTimeOlderDate?></TicketChangeTimeOlderDate>
<!--Optional:-->
<TicketCloseTimeOlderMinutes?></TicketCloseTimeOlderMinutes>
<!--Optional:-->
<TicketCloseTimeNewerMinutes?></TicketCloseTimeNewerMinutes>
<!--Optional:-->
<TicketCloseTimeNewerDate?></TicketCloseTimeNewerDate>
<!--Optional:-->
<TicketCloseTimeOlderDate?></TicketCloseTimeOlderDate>
<!--Optional:-->
<TicketPendingTimeOlderMinutes?></TicketPendingTimeOlderMinutes>
<!--Optional:-->
<TicketPendingTimeNewerMinutes?></TicketPendingTimeNewerMinutes>
<!--Optional:-->
<TicketPendingTimeNewerDate?></TicketPendingTimeNewerDate>
<!--Optional:-->
<TicketPendingTimeOlderDate?></TicketPendingTimeOlderDate>
<!--Optional:-->
<TicketEscalationTimeOlderMinutes?></TicketEscalationTimeOlderMinutes>
<!--Optional:-->
<TTicketEscalationTimeNewerMinutes?></TTicketEscalationTimeNewerMinutes>
<!--Optional:-->
<TicketEscalationTimeNewerDate?></TicketEscalationTimeNewerDate>
<!--Optional:-->
<TicketEscalationTimeOlderDate?></TicketEscalationTimeOlderDate>
<!--Optional:-->
<ArchiveFlags?></ArchiveFlags>
<!--Zero or more repetitions:-->
<OrderBy?></OrderBy>
<!--Zero or more repetitions:-->
<SortBy?></SortBy>

```

```
<!--Zero or more repetitions:-->
<CustomerUserID?></CustomerUserID>
</TicketSearch>
```

## 11.7.2. 例子：

### 11.7.2.1. WEB服务配置

下面是一个基本但完整的YAML格式的WEB服务配置文件，通过SOAP网络传输方法使用了所有的工单连接器操作。为了在OTRS中使用它，你需要复制它的内容到一个叫GenericTicketConnectorSOAP.yml的文件，并且在系统管理页面进入WEB服务窗口，然后在WEB服务概览窗口点击“添加WEB服务”按钮，然后在添加WEB服务窗口点击“导入WEB服务”按钮，将这个文件导入到OTRS中。

```
---
Debugger:
  DebugThreshold: debug
  TestMode: 0
Description: Ticket Connector SOAP Sample
FrameworkVersion: 3.4.x git
Provider:
  Operation:
    SessionCreate:
      Description: Creates a Session
      MappingInbound: {}
      MappingOutbound: {}
      Type: Session::SessionCreate
    TicketCreate:
      Description: Creates a Ticket
      MappingInbound: {}
      MappingOutbound: {}
      Type: Ticket::TicketCreate
    TicketUpdate:
      Description: Updates a Ticket
      MappingInbound: {}
      MappingOutbound: {}
      Type: Ticket::TicketUpdate
    TicketGet:
      Description: Retrieves Ticket data
      MappingInbound: {}
      MappingOutbound: {}
      Type: Ticket::TicketGet
    TicketSearch:
      Description: Search for Tickets
      MappingInbound: {}
      MappingOutbound: {}
      Type: Ticket::TicketSearch
  Transport:
    Config:
      MaxLength: 100000000
      NameSpace: http://www.otrs.org/TicketConnector/
      Type: HTTP::SOAP
  RemoteSystem: ''
  Requester:
    Transport:
      Type: ''
```

类似的例子可以用REST网络传输做到，REST WEB服务使用HTTP操作比如“POST”、“GET”、“PUT”、“PATCH”等等。这个操作连同—个叫做资源的URI路径—起定义—个OTRS通用接口操作或调用程序（取决于通讯方式）。

下列的例子为‘创建会话’使用/Session资源，为搜索工单和创建工单使用 /Ticket资源，为获取工单和更新工单使用/Ticket/{TicketID} 郑源（这里{TicketID}是一个工单的实际工单ID，例如/

Ticket/123)。为了在OTRS中使用它，你需要复制它的内容到一个叫GenericTicketConnectorREST.yml的文件，并且在系统管理页面进入WEB服务窗口，然后在WEB服务概览窗口点击“添加WEB服务”按钮，然后在添加WEB服务窗口点击“导入WEB服务”按钮，将这个文件导入到OTRS中。

```
---
Debugger:
  DebugThreshold: debug
  TestMode: '0'
Description: Ticket Connector REST Sample
FrameworkVersion: 3.4.x git
Provider:
  Operation:
    SessionCreate:
      Description: Creates a Session
      MappingInbound: {}
      MappingOutbound: {}
      Type: Session::SessionCreate
    TicketCreate:
      Description: Creates a Ticket
      MappingInbound: {}
      MappingOutbound: {}
      Type: Ticket::TicketCreate
    TicketGet:
      Description: Retrieves Ticket data
      MappingInbound: {}
      MappingOutbound: {}
      Type: Ticket::TicketGet
    TicketSearch:
      Description: Search for Tickets
      MappingInbound: {}
      MappingOutbound: {}
      Type: Ticket::TicketSearch
    TicketUpdate:
      Description: Updates a Ticket
      MappingInbound: {}
      MappingOutbound: {}
      Type: Ticket::TicketUpdate
  Transport:
    Config:
      KeepAlive: ''
      MaxLength: '100000000'
      RouteOperationMapping:
        SessionCreate:
          RequestMethod:
            - POST
          Route: /Session
        TicketCreate:
          RequestMethod:
            - POST
          Route: /Ticket
        TicketGet:
          RequestMethod:
            - GET
          Route: /Ticket/:TicketID
        TicketSearch:
          RequestMethod:
            - GET
          Route: /Ticket
        TicketUpdate:
          RequestMethod:
            - PATCH
          Route: /Ticket/:TicketID
      Type: HTTP::REST
    RemoteSystem: ''
  Requester:
    Transport:
      Type: ''
```

### 11.7.2.2. Perl SOAP请求者

下列代码是一个Perl脚本，可以通过通用接口连接到OTRS。为了执行工单连接器提供的操作，它使用了两个Perl的CPAN模块：SOAP::Lite和Data::Dumper。在你尝试运行这个脚本之前，请确保你的环境能够使用这些模块。

```
#!/usr/bin/perl -w
# --
# otrs.SOAPRequest.pl - sample to send a SOAP request to OTRS Generic Interface Ticket
# Connector
# Copyright (C) 2001-2020 OTRS AG, https://otrs.com/
# --
# This program is free software: you can redistribute it and/or modify
# it under the terms of the GNU General Public License as published by
# the Free Software Foundation, either version 3 of the License, or
# (at your option) any later version.
#
# This program is distributed in the hope that it will be useful,
# but WITHOUT ANY WARRANTY; without even the implied warranty of
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
# GNU General Public License for more details.
#
# You should have received a copy of the GNU General Public License
# along with this program. If not, see https://www.gnu.org/licenses/gpl-3.0.txt.
# --

use strict;
use warnings;

# use ../ as lib location
use File::Basename;
use FindBin qw($RealBin);
use lib dirname($RealBin);

use SOAP::Lite;
use Data::Dumper;

# ---
# Variables to be defined.

# this is the URL for the web service
# the format is
# <HTTP_TYPE>:://<OTRS_FQDN>/nph-genericinterface.pl/Webservice/<WEB_SERVICE_NAME>
# or
# <HTTP_TYPE>:://<OTRS_FQDN>/nph-genericinterface.pl/WebserviceID/<WEB_SERVICE_ID>
my $URL = 'http://localhost/otrs/nph-genericinterface.pl/Webservice/GenericTicketConnector';

# this name space should match the specified name space in the SOAP transport for the web
# service.
my $NameSpace = 'http://www.otrs.org/TicketConnector/';

# this is operation to execute, it could be TicketCreate, TicketUpdate, TicketGet,
# TicketSearch
# or SessionCreate. and they must to be defined in the web service.
my $Operation = 'TicketCreate';

# this variable is used to store all the parameters to be included on a request in XML
# format. Each
# operation has a determined set of mandatory and non mandatory parameters to work
# correctly. Please
# check the OTRS Admin Manual in order to get a complete list of parameters.
my $XMLData = '
<UserLogin>some user login</UserLogin>
<Password>some password</Password>
<Ticket>
  <Title>some title</Title>
  <CustomerUser>some customer user login</CustomerUser>
```

```

    <Queue>some queue</Queue>
    <State>some state</State>
    <Priority>some priority</Priority>
  </Ticket>
  <Article>
    <Subject>some subject</Subject>
    <Body>some body</Body>
    <ContentType>text/plain; charset=utf8</ContentType>
  </Article>
  ';

# ---

# create a SOAP::Lite data structure from the provided XML data structure.
my $SOAPData = SOAP::Data
  ->type( 'xml' => $XMLData );

my $SOAPObject = SOAP::Lite
  ->uri($NameSpace)
  ->proxy($URL)
  ->$operation($SOAPData);

# check for a fault in the soap code.
if ( $SOAPObject->fault ) {
  print $SOAPObject->faultcode, " ", $SOAPObject->faultstring, "\n";
}

# otherwise print the results.
else {

  # get the XML response part from the SOAP message.
  my $XMLResponse = $SOAPObject->context()->transport()->proxy()->http_response()-
  >content();

  # deserialize response (convert it into a perl structure).
  my $Deserialized = eval {
    SOAP::Deserializer->deserialize($XMLResponse);
  };

  # remove all the headers and other not needed parts of the SOAP message.
  my $Body = $Deserialized->body();

  # just output relevant data and no the operation name key (like TicketCreateResponse).
  for my $ResponseKey ( keys %{$Body} ) {
    print Dumper( $Body->{$ResponseKey} );
  }
}

```

### 11.7.2.3. Perl REST请求者

下列的代码是一个Perl脚本，可以通过通用接口连接到OTRS。为了执行工单连接器提供的操作，它使用了三个Perl的CPAN模块：JSON、REST::Client和Data::Dumper。在你尝试运行这个脚本之前，请确保你的环境能够使用这些模块。

```

#!/usr/bin/perl
# --
# otrs.RESTRequest.pl - sample to send a REST request to OTRS Generic Interface Ticket
# Connector
# Copyright (C) 2001-2020 OTRS AG, https://otrs.com/
# --
# This program is free software: you can redistribute it and/or modify
# it under the terms of the GNU General Public License as published by
# the Free Software Foundation, either version 3 of the License, or
# (at your option) any later version.
#
# This program is distributed in the hope that it will be useful,
# but WITHOUT ANY WARRANTY; without even the implied warranty of

```

```

# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
# GNU General Public License for more details.
#
# You should have received a copy of the GNU General Public License
# along with this program. If not, see https://www.gnu.org/licenses/gpl-3.0.txt.
# --

use strict;
use warnings;

## nofilter(TidyAll::Plugin::OTRS::Perl::Dumper)

# use ../ as lib location
use File::Basename;
use FindBin qw($RealBin);
use lib dirname($RealBin);

use JSON;
use REST::Client;

# ---
# Variables to be defined

# This is the HOST for the web service the format is:
# <HTTP_TYPE>:://<OTRS_FQDN>/nph-genericinterface.pl
my $Host = 'http://localhost/otrs/nph-genericinterface.pl';

my $RestClient = REST::Client->new(
    {
        host => $Host,
    }
);

# This is the Controller and Request the format is:
# /Webservice/<WEB_SERVICE_NAME>/<RESOURCE>/<REQUEST_VALUE>
# or
# /WebserviceID/<WEB_SERVICE_ID>/<RESOURCE>/<REQUEST_VALUE>
# This example will retrieve the Ticket with the TicketID = 1 (<REQUEST_VALUE>)
my $ControllerAndRequest = '/Webservice/GenericTicketConnectorREST/Ticket/1';

my $Params = {
    UserLogin      => "some user login",          # to be filled with valid agent login
    Password       => "some user password",      # to be filled with valid agent password
    DynamicFields  => 1,                          # optional, if set to 1,
                                                # ticket dynamic fields included in response
    AllArticles    => 1,                          # optional, if set to 1,
                                                # all ticket articles are included in response
                                                # more options to be found in
        # /Kernel/GenericInterface/Operation/Ticket/TicketGet.pm's
        # Run() subroutine documentation.
};

my @RequestParam;

# As sample web service configuration for TicketGet uses HTTP method GET all other
# parameters needs
# to be sent as URI query parameters

# ----
# For GET method
my $QueryParams = $RestClient->buildQuery( %{ $Params } );

$ControllerAndRequest .= $QueryParams;

# The @RequestParam array on position 0 holds controller and request
@RequestParam = ($ControllerAndRequest);

$RestClient->GET(@RequestParam);
# ----

# # ----
# # For POST method

```

```
# my $JSONParams = encode_json $Params;

# # The @RequestParam array on position 0 holds controller and request
# # on position 1 it holds the JSON data string that gets posted
# @RequestParam = (
#   $ControllerAndRequest,
#   $JSONParams
# );

# $RestClient->POST(@RequestParam);
# # ----

# If the host isn't reachable, wrong configured or couldn't serve the requested page:
my $ResponseCode = $RestClient->responseCode();
if ( $ResponseCode ne '200' ) {
    print "Request failed, response code was: $ResponseCode\n";
    exit;
}

# If the request was answered correctly, we receive a JSON string here.
my $ResponseContent = $RestClient->responseContent();

my $Data = decode_json $ResponseContent;

# Just to print out the returned Data structure:
use Data::Dumper;
print "Response was:\n";
print Dumper($Data);
```

#### 11.7.2.4. REST请求的cURL例子

用上面给出的通用工单连接器REST配置的例子，我们可以：

创建工单：在/Ticket路径上使用POST方法。

搜索工单：在/Ticket路径上使用GET方法。

更新工单：在/Ticket/{TicketID}路径上使用PATCH方法，{TicketID}是代表传输配置中:TicketID的模板。

获取工单：在/Ticket/{TicketID}路径上使用GET方法，{TicketID}是代表传输配置中:TicketID的模板。

##### 11.7.2.4.1. 创建一个新工单

cURL命令：

```
shell> curl "http://localhost/otrs/nph-genericinterface.pl/Webservice/
GenericTicketConnectorREST/Ticket?UserLogin=agent&Password=123" -H "Content-Type:
application/json" -d "{\"Ticket\":{\"Title\":\"REST Create Test\", \"Type\": \"Unclassified
\", \"Queue\": \"Raw\", \"State\": \"open\", \"Priority\": \"3 normal\", \"CustomerUser\":
\"customer\"}, \"Article\": {\"Subject\": \"Rest Create Test\", \"Body\": \"This is only a test
\", \"ContentType\": \"text/plain; charset=utf8\"}}" -X POST
```

响应：

```
{
  "ArticleID":5484,
  "TicketNumber":"1001936",
  "TicketID":"1686"
}
```

#### 11.7.2.4.2. 获取工单详细信息

cURL命令：

```
curl "http://localhost/otrs/nph-genericinterface.pl/Webservice/GenericTicketConnectorREST/Ticket/1686?UserLogin=agent&Password=123"
```

响应：

```
{
  "Ticket": [
    {
      "Age": 777,
      "PriorityID": 3,
      "ServiceID": "",
      "Type": "Unclassified",
      "Responsible": "root@localhost",
      "StateID": 4,
      "ResponsibleID": 1,
      "ChangeBy": 2,
      "EscalationTime": 0,
      "Changed": "2014-06-30 19:08:14",
      "OwnerID": 2,
      "RealTillTimeNotUsed": 0,
      "GroupID": 1,
      "Owner": "agent",
      "CustomerID": "OTRS",
      "TypeID": 1,
      "Created": "2014-06-30 19:08:12",
      "Priority": "3 normal",
      "UntilTime": 0,
      "EscalationUpdateTime": 0,
      "QueueID": 2,
      "Queue": "Raw",
      "State": "open",
      "Title": "REST Create Test",
      "CreateBy": 2,
      "TicketID": 1686,
      "StateType": "open",
      "EscalationResponseTime": 0,
      "UnlockTimeout": 0,
      "EscalationSolutionTime": 0,
      "LockID": 1,
      "TicketNumber": "1001936",
      "ArchiveFlag": "n",
      "Lock": "unlock",
      "CreateTimeUnix": 1404173292,
      "SLAID": "",
      "CustomerUserID": "customer"
    }
  ]
}
```

#### 11.7.2.4.3. 更新工单

cURL命令：

```
curl "http://localhost/otrs/nph-genericinterface.pl/Webservice/GenericTicketConnectorREST/Ticket/1686?UserLogin=agent&Password=123" -H "Content-Type: application/json" -d "{\"Ticket\": {\"Queues\": \"Postmaster\"}}" -X PATCH
```



响应：

```
{
  "TicketNumber": "1001936",
  "TicketID": "1686"
}
```

#### 11.7.2.4.4. 搜索工单

cURL命令：

```
curl "http://localhost/otrs/nph-genericinterface.pl/Webservice/GenericTicketConnectorREST/Ticket?UserLogin=agent&Password=123&Queue=Postmaster"
```

响应：

```
{
  "TicketID": [
    "1686",
    "102",
    "100",
    "1"
  ]
}
```

## 12. OTRS守护进程

OTRS守护进程是一组计划和执行后台任务（循环任务或事件触发任务）的独立的系统进程。OTRS守护进程是正确系统操作的基础保障。

在OTRS之前的版本中（从3.1到4）有另外的进程叫做OTRS调度器，它做了OTRS 5中OTRS守护进程的一部分工作。原来的进程被OTRS守护进程替换，从底重写的OTRS守护进程比它的前任更稳定、更具伸缩性和更健壮。

OTRS守护进程能够同时处理多达10个任务，还能与群集中不同前端服务器的OTRS守护进程进行协同工作。

OTRS守护进程在空闲时包含5个进程：

- 主守护进程（bin/otrs.Daemon.pl）

这个进程负责启动其它子进程并保持其正常运行。

- 任务工作者守护进程（Kernel/System/Daemon/DaemonModules/SchedulerTaskWorker.pm）

这个进程负责按先进先出的原则执行列表中的所有任务。它可以通过创建子进程来处理同时的任务，并每秒检查几次任务列表。任务列表可以由任务管理器进程、事件处理器和系统的其它部分填充。

它的主要使命是尽可能快地处理任务列表中的所有任务。

- 未来任务管理器守护进程（Kernel/System/Daemon/DaemonModules/SchedulerFutureTaskManager.pm）

这个守护进程检查在未来某个时间执行的非循环任务（例如：当一个通用接口调用程序试图抵达一个服务器时而不能抵达时，会产生一个5分钟后执行的计划任务）。在到达正确的时间时它将这个任务发送给任务工作者守护进程。

- Cron任务管理器守护进程 (Kernel/System/Daemon/DaemonModules/SchedulerCronTaskManager.pm)

这个守护进程计算所有循环任务的下一次执行时间（例如：每周清理一次缓存）。这种任务在系统配置中指定。在每个任务合适的时间它将需要的信息发送给任务工作者守护进程以执行它们。

## 注意

如果在系统配置中修改了一个任务执行时间定义，OTRS守护进程可能要花最多1个小时才能自动获得修改后的时间。另外也可以通过重启OTRS守护进程来立即应用修改了的执行时间。

- 自动任务管理器守护进程 (Kernel/System/Daemon/DaemonModules/SchedulerGenericAgentTaskManager.pm)

这个守护进程扫描存储在数据库中的定时自动任务（丢弃所有事件触发的自动任务）。当自动任务运行时间到了时，它将需要的信息发送给任务工作者守护进程以处理任务。

## 注意

活动进程数取决于一个时间点上OTRS守护进程同时执行的任务数。

默认每个守护进程在一个单独的文件（位于 /opt/otrs/var/log/Daemon/\*.log）中记录所有错误消息的日志。这些日志在系统中保留一个预定义的周期。要修改保留时间和/或还要记录非错误消息，请更新系统配置Daemon -> Core::Log中的设置参数。

当一个任务因为某个原因不能成功执行时，会给预定义的收件人发送一封邮件来报告这个问题，邮件内容包括错误消息和可用的跟踪信息。

OTRS守护进程是一个自动进程，通常不需要人工交互。但是也可以在需要时查询它的状态和启动/停止它。

为了确保OTRS守护进程正常运行，有一个Cron任务持续检查这个进程是活的。主守护进程甚至在没有数据库连接时就能工作，所以在系统启动时数据库进程启动之前Cron任务就启动OTRS守护进程是完全没有问题的，它还能容忍数据库失连。

如果因为某个原因需要停止OTRS守护进程（例如在系统维护期间），会保存所有未处理的任务，一旦这个进程重新启动它会继续执行所有保存的任务，对于循环任务它只执行最近一次的任务（如果它的到期时间是在停机时间内的话）。

## 12.1. OTRS守护进程图形界面

除非OTRS守护停止运行，否则它在OTRS图形界面中不可见。

当系统检测到OTRS守护进程没有运行时，会给预定义组（默认是“admin”组）的用户显示一条通知消息。

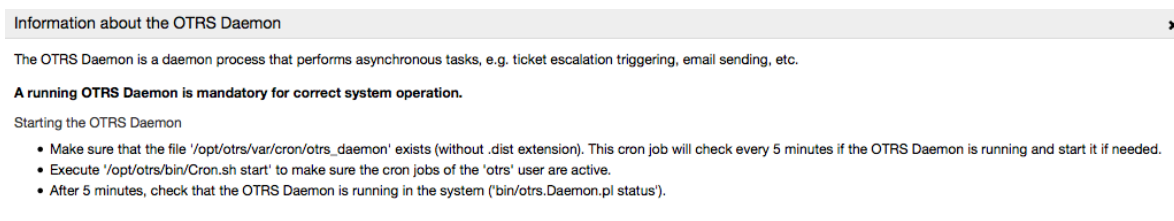
要禁用这个通知（不推荐），修改或添加通知的组，请在系统配置中编辑Frontend::NotifyModule###800-Daemon-Check的设置。

图 4.114. 守护进程通知



点击这个通知系统会显示一个重叠窗口来解释启动OTRS守护进程的步骤。

## 图 4.115. 启动OTRS守护进程



## 12.2. OTRS守护进程命令行接口

OTRS守护进程命令行工具能让你控制主守护进程（启动或停止）或查询它的状态。还有获取其它四个子进程的详细信息的工具。

### 12.2.1. 主守护进程工具

要启动、停止或查询守护进程状态，使用bin/otrs.Daemon.pl脚本。

#### 例 4.30. 启动OTRS守护进程示例

```
shell> cd /opt/otrs/  
shell> OTRS_HOME/bin/otrs.Daemon.pl start
```

#### 可用的选项

- **start** - 启动OTRS守护进程。
- **stop** - 停止OTRS守护进程。
- **status** - 查询OTRS守护进程状态。
- **start --debug** - 以调试模式启动OTRS守护进程。

在这个模式下每个守护进程报告不同的消息（取决于已经执行的操作）。这个模式不推荐在生产环境中使用。

- **stop --force** - 不等待子进程完成立即停止OTRS守护进程。

强制停止减少了主守护进程等待其它子进程成功完成的时间，从30秒（正常）减少到5秒（强制）。

### 12.2.2. 其它守护进程的工具

列出所有配置的子守护进程，即主守护进程要使用控制台命令Maint::Daemon::List启动并保持其运行的：

#### 例 4.31. 列出所有配置的子守护进程的示例

```
shell> cd /opt/otrs/  
shell> bin/otrs.Console.pl Maint::Daemon::List
```

要列出所有守护进程的详细信息，使用控制台命令：Maint::Daemon::Summary。

#### 例 4.32. 汇总所有守护进程信息的示例

```
shell> cd /opt/otrs/  
shell> bin/otrs.Console.pl Maint::Daemon::Summary
```



# 第 5 章 定制

## 1. 访问控制列表 (ACL)

### 1.1. 介绍

从OTRS 2.0开始，访问控制列表 (ACL) 能用来控制到工单、模块、队列等的访问权限，或在某些状况下影响工单操作（关闭、转移等）。ACL可以作为现有系统权限角色和组的补充。使用ACL在系统中映射基本的基于工单属性的工作流。

一般来说，ACL用基于一个预定义的规则集来减少一个工单可能的选项。

ACL定义可以直接写入文件 `Kernel/Config.pm`。但是现在不再推荐这种方式，因为在系统管理面板Access Control Lists访问控制列表图形界面允许首先将ACL保存到数据库中，然后在准备使用时部署到一个文件。

本章有一些ACL示例可以帮助你熟悉定义ACL的流程，还有ACL所有可能重要的设置的参考说明。

#### 警告

默认用户'`root@localhost`'不受工单ACL的影响。

### 1.2. 定义

ACL定义可以拆分为两大部分：'匹配'和'修改'。在ACL匹配部分包含用来满足使用ACL的属性，如果ACL中定义的属性不匹配传递过来的属性，则ACL不会产生任何影响，但匹配了ACL就会产生影响。修改部分包含减少工单可能选项的规则。

#### 匹配部分

- 属性

本部分包含可动态修改的匹配选项。例如：在一个工单创建时随着服务人员设置工单信息而动态修改工单数据。如果一条ACL匹配了一个工单属性，则只有当匹配的属性被选择时ACL将会激活并可能减少其它的工单属性，但只要选择了另一个值，ACL就不会产生任何影响。

- PropertiesDatabase数据库中的属性

本部分与'属性'部分类似，但在工单保存到数据库前不会对工单属性有任何修改，这意味着不提交工单就不会对工单属性有任何影响。本部分不适用于工单创建窗口（因为工单还没有在数据库中创建）。

#### 修改部分

- 可能的

'可能的'部分重置数据，以便将数据减少为只有在本部分列出的元素。

- 可能的添加

在'添加可能的'部分的元素添加因在其它ACL减少而缺失的元素。'添加可能的'只能连同其它有'可能的'或'不可能的'部分的ACL使用。

- 不可能的

本部分用于从当前数据中移除特定元素。它可以单独使用，也可以与连同其它有'可能的'或'添加可能的'部分的ACL使用。

为了使ACL的开发更容易且更强大，每个部分还有一些所谓的属性‘修改器’。这个修改器的解释如下：

#### 修改器

- [Not非]

这个修改器用来取反一个值，例如：有关工单优先级的'[Not]2-低'，与以下结果相同：'1-非常低'、'3-正常'、'4-高'、'5-非常高'。

- [RegExp正则表达式（区分大小写）]

用来定义一个正则表达式以匹配多个值，例如：关于工单优先级的'[RegExp]低'就是'1-非常低'、'2-低'。

- [regex正则表达式（忽略大小写）]

它与[RegExp正则表达式（区分大小写）]很相似，但它是不区分大小写的。

- [NotRegExp-正则表达式取反]

对正则表达式取反，如关于工单优先级的'[NotRegExp]低'就是'3-正常'、'4-高'、'5-非常高'

- [Notregex-正则表达式取反（忽略大小写）]

它与[NotRegExp-正则表达式取反]很相似，但它是不区分大小写的。

## 1.3. 示例

下面的示例按图形界面和文本格式两种方式展示。

### 例 5.1. 仅允许工单优先级5的工单转移到一个队列的ACL

本示例为你展示一条ACL的基本结构。首先，ACL需要有一个名称。本案例中，ACL的名称是“100-Example-ACL”。注意：ACL在执行前会按数字排序，所以你要斟酌命名。

其次，你要有一个“Properties属性”部分作为工单的过滤器。这部分的所有条件都用来确定ACL是否适用或不适用一个工单。在我们的示例中，如果工单在队列“RAW”中且优先级为“5-非常高”就匹配ACL，这还会受窗口中数据变更的影响（例如：此时工单在队列“RAW”中且优先级为“3-正常”就不会匹配ACL，但随后选择了优先级下拉菜单并将优先级变更为“5-非常高”后就匹配了ACL）。

最后，“Possible可能的”部分定义窗口中的修改内容。在本示例中，在工单窗口可用的队列只能选择队列“Alert”。

## 图 5.1. ACL 100-Example-ACL

▼ Edit ACL structure

**Match settings**

- ▼ Properties
  - ▼ Ticket
    - Queue:
      - Raw x Exact match [dropdown] [input]
    - Priority:
      - 5 very high x Exact match [dropdown] [input]
    - [+]
    - [+]
    - [+]

---

**Change settings**

- ▼ Possible
  - ▼ Ticket
    - Queue:
      - Alert x Exact match [dropdown] [input]
    - [+]
    - [+]
    - [+]

```
# ticket acl
$self->{TicketAcl}->{'100-Example-ACL'} = {
  # match properties
  Properties => {
    # current ticket match properties
    Ticket => {
      Queue => ['Raw'],
      Priority => ['5 very high'],
    }
  },
  # return possible options (white list)
  Possible => {
    # possible ticket options (white list)
    Ticket => {
      Queue => ['Alert'],
    },
  },
};
```

**例 5.2. ACL**仅允许已保存到数据库中的优先级为**5**的工单移到一个队列。

本示例与上一个示例很相似，但本示例仅匹配已保存到数据库在队列“Raw”中且优先级为“5-非常高”的工单。这种ACL在工单的修改内容真正更新到数据库之前不会在窗口中对工单有任何修改。

## 图 5.2. ACL 102-Example-ACL

▼ Edit ACL structure

**Match settings**

- ▼ PropertiesDatabase
  - ▼ Ticket
    - Queue:
      - Raw x Exact match [dropdown] [input]
    - Priority:
      - 5 very high x Exact match [dropdown] [input]
    - [input]
    - [dropdown]
    - [input]

---

**Change settings**

- ▼ Possible
  - ▼ Ticket
    - Queue:
      - Alert x Exact match [dropdown] [input]
    - [input]
    - [dropdown]
    - [input]

```
# ticket acl
$self->{TicketAcl}->{'102-Example-ACL'} = {
  # match properties
  PropertiesDatabase => {
    # current ticket match properties
    Ticket => {
      Queue => ['Raw'],
      Priority => ['5 very high'],
    }
  },
  # return possible options (white list)
  Possible => {
    # possible ticket options (white list)
    Ticket => {
      Queue => ['Alert'],
    },
  },
};
```

### 例 5.3. ACL 禁止关闭‘Raw’队列的工单，并隐藏“关闭”按钮。

在这里可以看到一个工单字段（状态）用多个可能值过滤选择。还可以限制某些工单能够执行的操作。在本示例中，工单不能被关闭。



### 图 5.3. ACL 102-Second-Example-ACL

▼ Edit ACL structure

**Match settings**

- ▼ Properties
  - ▼ Ticket
    - Queue:
      - Raw  Exact match
      - 
      - 
      -

---

**Change settings**

- ▼ Possible
  - ▼ Ticket
    - State:
      - new  open  pending reminder  Exact match
      - 
      -
  - ▼ PossibleNot
    - ▼ Action
      - AgentTicketClose  Exact match
      - 
      -

```

$Self->{TicketAcl}->{'102-Second-Example-ACL'} = {
  # match properties
  Properties => {
    # current ticket match properties
    Ticket => {
      Queue => ['Raw'],
    }
  },
  # return possible options (white list)
  Possible => {
    # possible ticket options (white list)
    Ticket => {
      State => ['new', 'open', 'pending reminder'],
    },
  },
  # return also not possible options (black list)
  PossibleNot => {
    # not possible action options
    Action => [ 'AgentTicketClose' ],
  },
};

```

#### 例 5.4. ACL总是移除‘成功关闭’状态。

本示例展示如何定义取反过滤器（移除‘成功关闭’状态）。还可以看到没有定义工单属性将匹配任意工单，如ACL总是被应用到工单。如果你要默认隐藏某些值时很有用，只在特殊情况下启用这些值（例如在特殊组中的服务人员）。

## 图 5.4. ACL 103-Third-ACL-Example

▼ Edit ACL structure

**Match settings**

-

---

**Change settings**

▼ PossibleNot

▼ Ticket

State:

closed successful x Exact match

-

-

```
$Self->{TicketAcl}->{'103-Third-ACL-Example'} = {
  # match properties
  Properties => {
    # current ticket match properties (match always)
  },
  # return possible options
  PossibleNot => {
    # possible ticket options
    Ticket => {
      State => ['closed successful'],
    },
  },
};
```

例 5.5. ACL为创建在以“HW”开头的队列中的工单仅显示硬件服务。

本示例还展示了如何使用正则表达式来匹配工单和过滤可用选项。

## 图 5.5. ACL 104-Only-Hardware-Services-for-HW-Queues-ACL

▼ Edit ACL Structure

**Match settings**

▼ Properties

▼ Ticket

Queue:

[RegExp]HW x Regular expression

-

-

-

---

**Change settings**

▼ Possible

▼ Ticket

Service:

[RegExp]^Hardware x Regular expression

-

-

-

```
$Self->{TicketAcl}->{'104-Only-Hardware-Services-for-HW-Queues-ACL'} = {
```

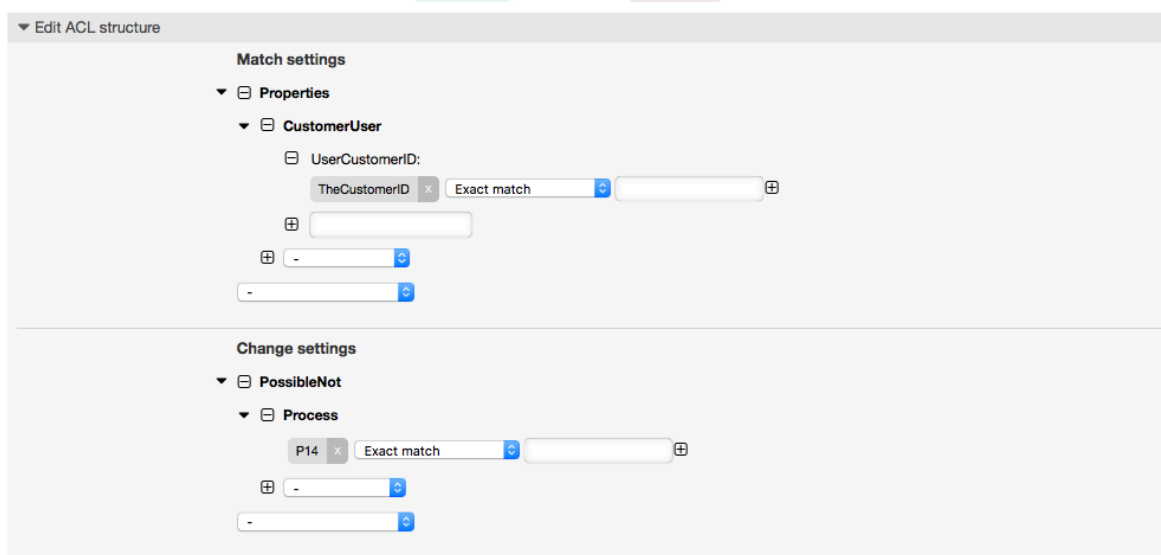
```

# match properties
# note we don't have "Ticket => {" because there's no ticket yet
Properties => {
Queue => {
    Name => ['[RegExp]HW'],
    }
},
# return possible options
Possible => {
    # possible ticket options
    Ticket => {
        Service => ['[RegExp]^(Hardware)'],
    },
},
};

```

例 5.6. ACL使用CustomerID禁用客户界面的一个流程

图 5.6. ACL 105-Disallow-Process-For-CustomerID



```

$self->{TicketAcl}->{"105-Disallow-Process-For-CustomerID"} = {
    'Possible' => {},
    'PossibleNot' => {
        'Process' => [
            'P14'
        ]
    },
    'Properties' => {
        'CustomerUser' => {
            'UserCustomerID' => [
                'CustomerID'
            ]
        }
    },
    'PropertiesDatabase' => {},
    'StopAfterMatch' => 0
};

```

## 1.4. ACL设置参考

下面的示例列出了能用于ACL的所有参数。

关于流程工单如何使用ACL的详细描述，请参阅流程管理那一章的访问控制列表（ACL）部分。

## 例 5.7. 所有可能的重要ACL设置的参考

```
# ticket acl
$Self->{TicketAcl}->{'200-ACL-Reference'} = {
  # match properties (current values from the form)
  # 匹配属性 (窗口中的当前值)
  Properties => {

    # the used frontend module
    Frontend => {
      Action => ['AgentTicketPhone', 'AgentTicketEmail'],
    },

    # the logged in agent登录的服务人员
    User => {
      UserLogin => ['some login'],
      Group_rw => [
        'hotline',
      ],
      Role => [
        'admin',
      ],
      # ...
    },

    # the logged in customer登录的客户
    CustomerUser => {
      UserLogin => ['some login'],
      UserCustomerID => ['some customer id'],
      Group_rw => [
        'hotline',
      ],
      # ...
    },

    # process properties流程属性
    Process => {
      ProcessEntityID => ['Process-9c378d7cc59f0fce4cee7bb9995ee3eb'],
      # the Process that the current ticket is part of
      ActivityEntityID => ['Activity-f8b2fdebe54eeb7b147a5f8e1da5e35c'],
      # the current Activity of the ticket
      ActivityDialogEntityID => ['ActivityDialog-aff0ae05fe6803f38de8fff6cf33b7ce'],
      # the current ActivityDialog that the Agent/Customer is using
    },

    # ticket properties工单属性
    Queue => {
      Name => ['Raw'],
      QueueID => ['some id'],
      GroupID => ['some id'],
      Email => ['some email'],
      RealName => ['OTRS System'],
      # ...
    },
    Service => {
      ServiceID => ['some id'],
      Name => ['some name'],
      ParentID => ['some id'],
      # ...
    },
    Type => {
      ID => ['some id'],
      Name => ['some name'],
      # ...
    },
    Priority = {
      ID => ['some id'],
      Name => ['some name'],
      # ...
    },
    SLA = {
```

```

    SLAID    => ['some id'],
    Name     => ['some name'],
    Calendar => ['some calendar'],
    # ...
  },
  State = {
    ID       => ['some id'],
    Name     => ['some name'],
    TypeName => ['some state type name'],,
   TypeID   => ['some state type id'],
    # ...
  },
  Owner => {
    UserLogin => ['some login'],
    Group_rw => [
      'some group',
    ],
    Role => [
      'admin',
    ],
    # ...
  },
  Responsible => {
    UserLogin => ['some login'],
    Group_rw => [
      'some group',
    ],
    Role => [
      'admin',
    ],
    # ...
  },
  DynamicField => {
    # Names must be in DynamicField_<field_name> format.名称必须是
DynamicField_<field_name>格式。
    # Values in [ ... ] must always be the untranslated internal data keys
    #   specified in the dynamic field definition and
    #   not the data values shown to the user.
    # [...]中括号里的值必须是没有翻译的定义动态字段的内部数据，而不是显示给用户看的数据。
    DynamicField_Field1      => ['some value'],
    DynamicField_OtherField  => ['some value'],
    DynamicField_TicketFreeText2 => ['some value'],
    # ...
  },
  # alternatively, ticket properties can be specified in the ticket hash或者，可以用工单
哈希表来指定工单属性
  Ticket => {
    Queue           => ['Raw'],
    State           => ['new', 'open'],
    Priority         => ['some priority'],
    Lock            => ['lock'],
    CustomerID      => ['some id'],
    CustomerUserID  => ['some id'],
    Owner           => ['some owner'],
    DynamicField_Field1 => ['some value'],
    DynamicField_MyField => ['some value'],
    # ...
  },
},

# match properties (existing values from the database)匹配属性（数据库中存在的值）
PropertiesDatabase => {
  # See section "Properties", the same config can be used here.参阅"Properties"部分，那里的
配置完全适用于此处。
  # ...
}

# reset possible options (white list)重置可用的选项（白名单）
Possible => {
  # possible ticket options (white list)可用的选项（白名单）
  Ticket => {
    Queue => ['Hotline', 'Coordination'],

```

```

    State => ['some state'],
    Priority => ['5 very high'],
    DynamicField_Field1 => ['some value'],
    DynamicField_MyField => ['some value'],
    # ...
    NewOwner => ['some owner'],
    OldOwner => ['some owner'],
    # ...
  },

  # Limit the number of possible ActivityDialogs the Agent/Customer
  # can use in a process ticket.
  #限制服务人员/客户联系人在一个流程工单中可用的活动对话框数量
  ActivityDialog => ['AD1', 'AD3'],

  # Limit the number of possible Processes that can be started限制能启动的可用流程数量
  Process => ['Process-9c378d7cc59f0fce4cee7bb9995ee3eb',
'Process-12345678901234567890123456789012'],

  # possible action options (white list)可用的操作选项 (白名单)
  Action => [
    'AgentTicketBounce',
    'AgentTicketPhone',      # only used to show/hide the Split action
    'AgentLinkObject',      # only used to show/hide the Link action
    # ...
  ],
},
# add options (white list)添加选项 (白名单)
PossibleAdd => {
  # See section "Possible"参阅"Possible"部分
  # ...
},
# remove options (black list)移除选项 (黑名单)
PossibleNot => {
  # See section "Possible"参阅"Possible"部分
  # ...
},
};

```

## 注意

匹配ACL时，如果传递了CustomerUserID参数，ACL机制会使用提供的CustomerUserID来比较的ACL定义，以便收集客户联系人的详细信息来填充客户联系人哈希表，在'Properties'匹配时还会覆盖工单哈希表中的客户信息。另一方面，在'PropertiesDatabase'匹配时也会计算，但是用工单客户为依据来收集信息数据。

注意：在客户界面，始终会传递当前登录的客户联系人的CustomerUserID。

要知道在工单搜索窗口（AgentTicketSearch和CustomerTicketSearch）唯一受ACL影响的就是动态字段。这意味着在工单搜索窗口你不能限制其它任何属性比如工单类型、状态、队列等等。

从OTRS 4开始，'操作'参数不再是一个哈希表，而是一个数组引用，并能使用任何修改器用于'修改'部分。

## 2. 流程管理

### 2.1. 介绍

可以利用OTRS系统定制流程(工作流)模型。基本思路在于定义可重复执行的流程，并将工作内容指派给不同的人员，同时确保流程按照预期进入不同阶段并最终顺利完成。

### 2.2. 流程示例

让我们看一个例子，更易于阐述流程管理。我们将定义一个订书流程：

## 2.2.1. 记录需求

在生成订单之前，雇员对书籍的需求将被记录。下列书籍是我们示例中所需要的：

标题: Prozessmanagement für Dummies  
作者: Thilo Knuppertz  
ISBN: 3527703713

## 2.2.2. 经理审批

部门经理需要确认需求。若经理拒绝，需要记录决定原因。若同意，流程将会跳转到采购部。

## 2.2.3. 采购部处理

现在采购人员需要找出如何以最好的条件购买所需书籍。若书籍缺货，需要在工单中进行记录。如果成功下单，将需要记录供应商、售价与交货时间。

## 2.2.4. 收发室处理

这批货物将到达公司。进货部门负责检查货物，并记录货品到达日期。接着员工将被告知他们的订单已经到达，并已准备好进行接收。

## 2.3. 实施示例

假设在这个工作流中，工单可以像随同文档一样接收变更记录，那么我们就有了很清晰的工单处理流程图。

通过示例流程分析，我们可以确定下列需要的条目：

- 数据记录功能，我们称其为 *Activity Dialogs* 活动对话框，
- 根据数据变化自动反应的检查，我们称之为 *Transitions* 转换，
- 在流程单成功转换后，对流程工单的变更，我们称之为 *Transition Actions* 转换操作。

我们同样还需要一些不那么复杂的额外项：

- 可能存在不止一个活动对话框(*Activity Dialog*)。在示例中，经理必须在"Approve 同意"与 "Deny 拒绝"中做出决定。我们称之为 *Activity* 活动。

现在，结合活动(*Activities*)、活动对话框(*Activity Dialogs*)、转换(*Transitions*)与转换操作(*Transition Actions*)，我们可以对示例的各个独立步骤进行建模。现在还缺少为每个工作流步骤设置具体顺序。我们称之为 *Process* 流程。为了能在之后指代所有这些实体，我们将分配给他们括号中的缩写。这个缩写基于内部识别机制 *EntityID*。

*EntityID*以1~2个字母开头(取决于流程组件或实体)，接着跟随连续编号，例如：

- 流程: 'P1', 'P2' ... 'Pn'.
- 活动: 'A1', 'A2' ... 'An'.
- 活动对话框: 'AD1', 'AD2' ... 'ADn'.
- Transition: 'T1', 'T2' ... 'Tn'.
- 转换操作: 'TA1', 'TA2' ... 'TAn'.

在新建流程和相关内容之前，需要提前准备好系统，我们需要定义一些队列、客户联系人和动态字段，还需要在系统配置中设置一些参数。

创建如下队列：

- 管理

- 员工
- 采购
- 邮局

创建如下用户：

- 经理
- 员工

创建如下动态字段：

- 标题

标签	标题
类型	文本
对象	工单

- 作者

标签	作者
类型	文本
对象	工单

- ISBN

标签	ISBN
类型	文本
对象	工单

- 状态

标签	状态
类型	下拉选项框
对象	工单
可能的值	<ul style="list-style-type: none"> <li>• 审批</li> <li>• 审批拒绝</li> <li>• 通过审批</li> <li>• 采购拒绝</li> <li>• 生成订单</li> <li>• 货品收到</li> </ul>

注意：需要在创建动态字段的“键”和“值”时尽可能使用最接近的值。

- 供应商

标签	供应商
类型	文本
对象	工单



- 价格

标签	价格
类型	文本
对象	工单

- 交货日期

标签	交货日期
类型	日期
对象	工单

- 收货日期

标签	收货的日期
类型	日期
对象	工单

在系统配置(SysConfig)中设置：

- 'Ticket::Responsible': 是
- 'Ticket::Frontend::AgentTicketZoom###ProcessWidgetDynamicFieldGroups':

键：	值：
书籍	标题, 作者, ISBN
一般	状态
订单	价格, 供应商, 交货日期
运输	收货日期

- 'Ticket::Frontend::AgentTicketZoom###ProcessWidgetDynamicField':

键：	值：
作者	1
收货日期	1
交货日期	1
ISBN	1
价格	1
状态	1
供应商	1
标题	1

现在我们可以开始真正的流程管理配置了。下一步，我们将定义所需的独立实体。

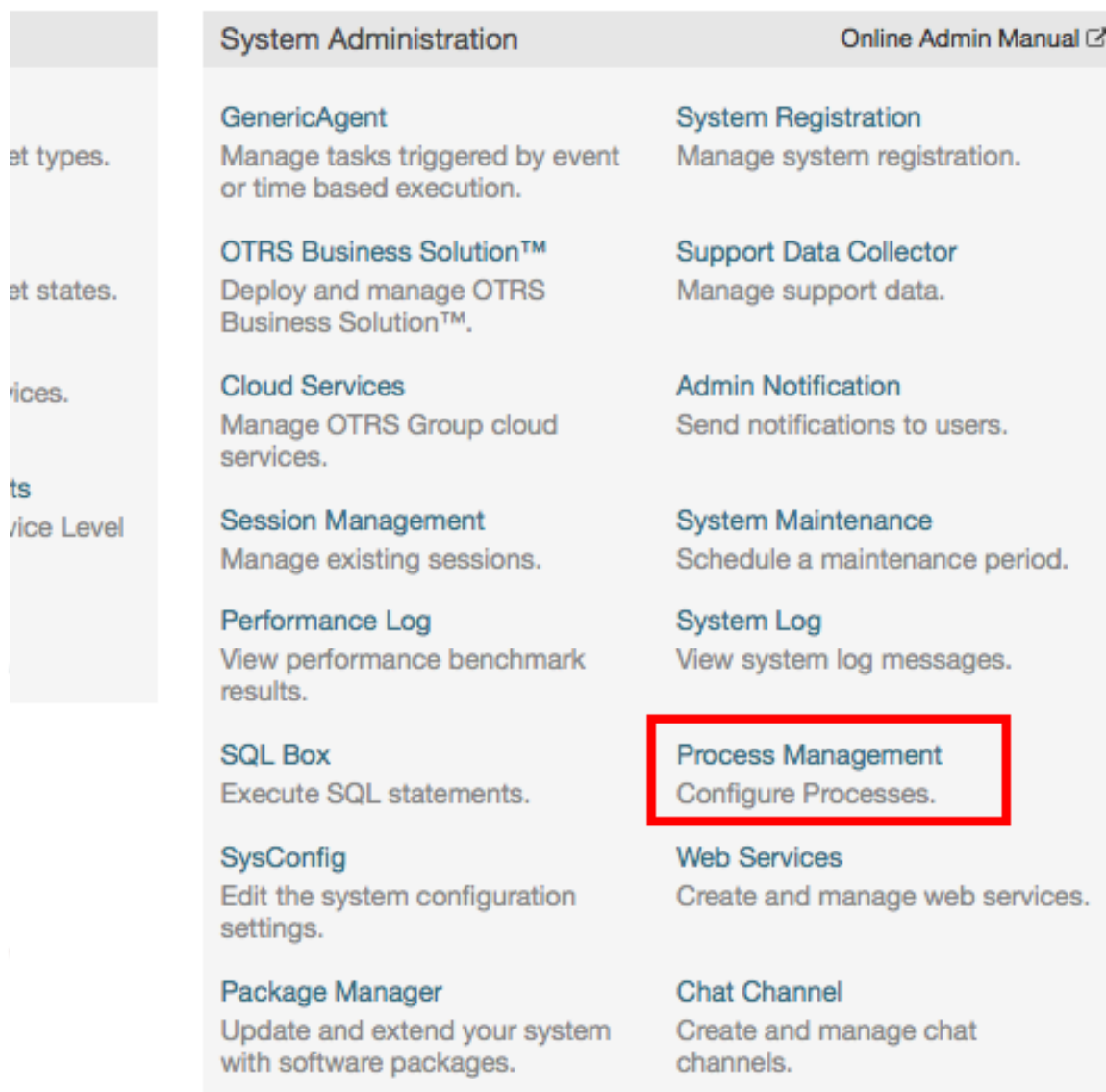
### 2.3.1. 流程(作为容器)

创建一个新流程，需要点击管理员面板中系统管理的"Process Management 流程管理"，我们会进入流程管理概览窗口。在创建流程后，我们就可以创建其他全部实体(或流程组件)。

#### 注意

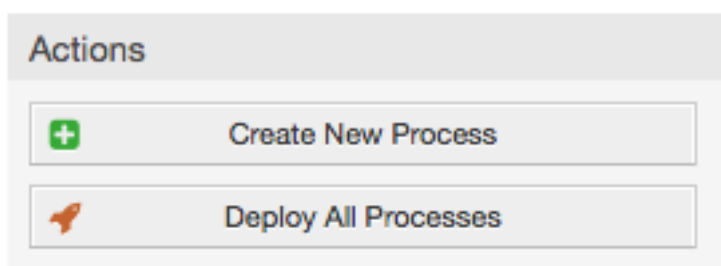
一个流程中所定义的Activity(活动)、Activity Dialog(活动对话框)、Transition(转换) 与 Transition Action(转换操作)，都可以被系统中的所有流程使用。

图 5.7. OTRS 系统管理窗口 - 系统管理



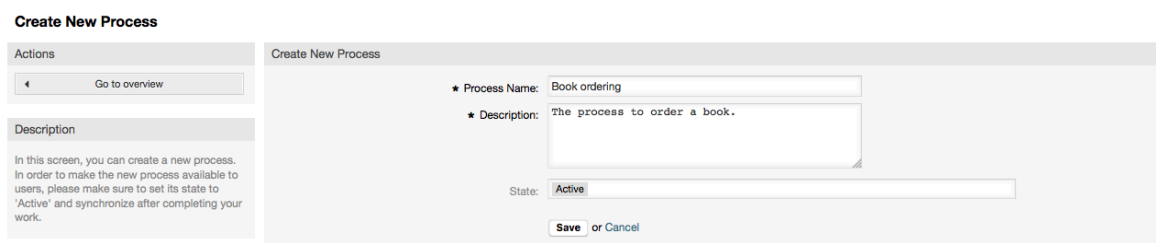
点击左侧操作框的“创建新的流程”按钮。

图 5.8. 创建新的流程按钮



填写流程信息、设置流程名与描述，我们在完成所有配置任务之前，我们先把流程状态设为“非活动”。

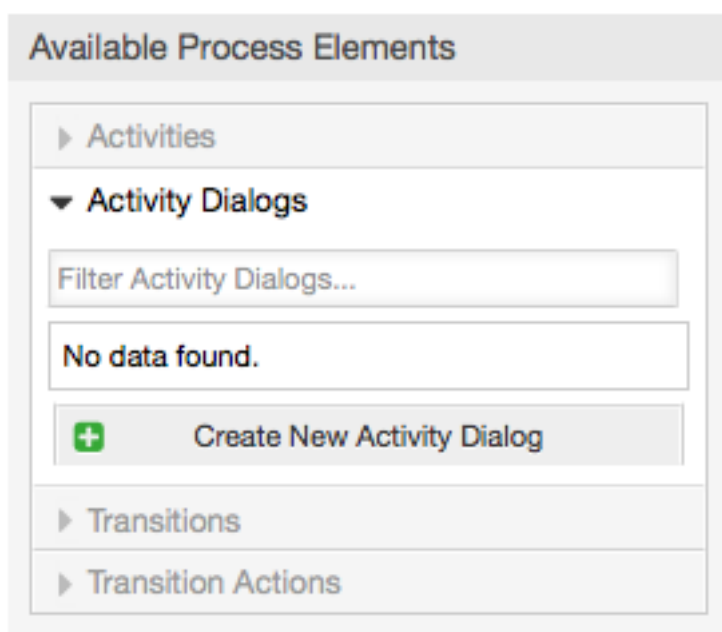
图 5.9. 添加新流程



## 2.3.2. 活动对话框

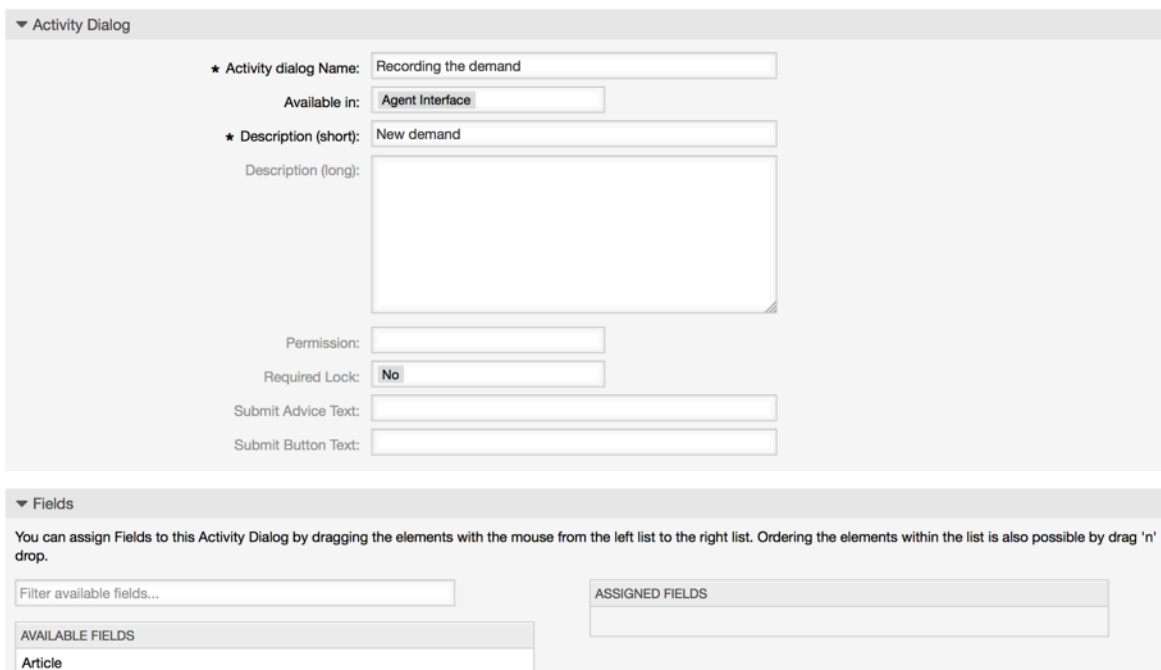
在流程管理概览窗口点击新建的流程名称，然后在“可用的流程元素”中点击“活动对话框”（点击会展开活动对话框选项，并收起其它具有可折叠效果的选项），再点击“创建新的活动对话框”。

图 5.10. 创建新的活动对话框按钮



在打开的弹出窗口中填写“活动对话框名称”和“描述（简短）”相同内容，本例中我们将其它字段都保持默认值，要给活动对话框分配字段，仅需从“可用的字段”池中拖动必填字段，并放到“分配的字段”池中即可。“分配的字段”池中的字段顺序为将字段放进去的先后顺序，可通过拖放操作进行字段重新排序。

图 5.11. 添加新的活动对话框



▼ Activity Dialog

★ Activity dialog Name: Recording the demand

Available in: Agent Interface

★ Description (short): New demand

Description (long):

Permission:

Required Lock: No

Submit Advice Text:

Submit Button Text:

▼ Fields

You can assign Fields to this Activity Dialog by dragging the elements with the mouse from the left list to the right list. Ordering the elements within the list is also possible by drag 'n' drop.

Filter available fields...

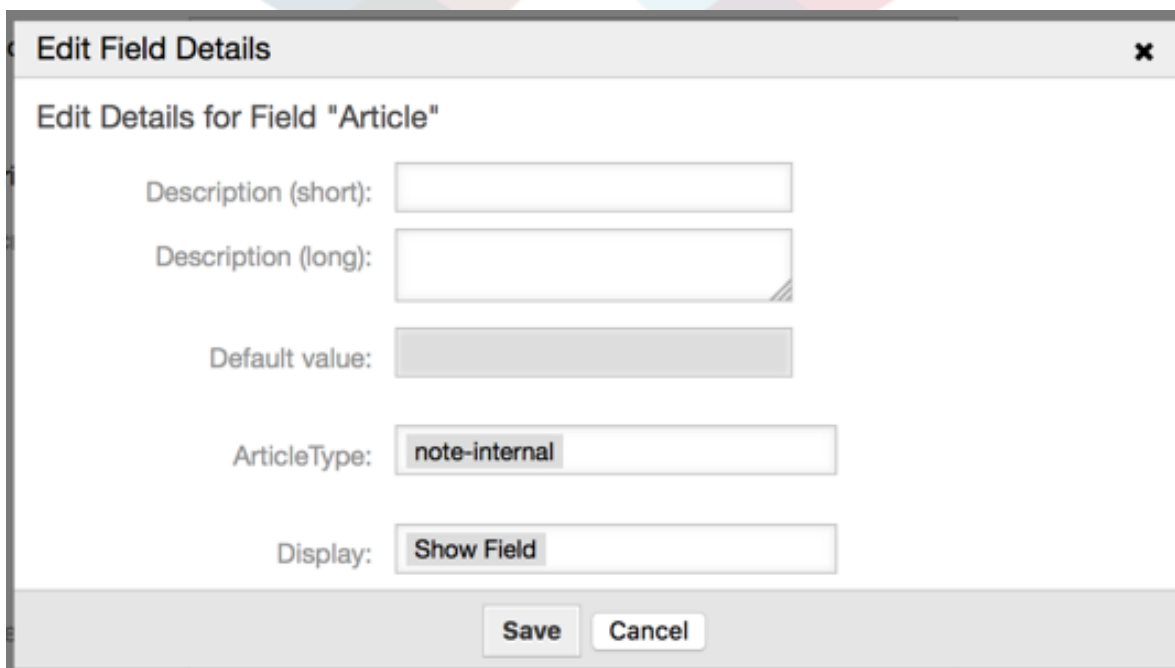
ASSIGNED FIELDS

AVAILABLE FIELDS

Article

将选择的字段拖入"分配的字段"池后，会弹出另一个窗口，显示该字段详细信息，保留默认选项即可，接着将信件字段的信件类型(ArticleType)字段设置为 "备注-内部"。

图 5.12. 编辑字段详细信息(信件)



✕

Edit Field Details

Edit Details for Field "Article"

Description (short):

Description (long):

Default value:

ArticleType: note-internal

Display: Show Field

Save Cancel

在所有字段都分配好之后，点击主弹出窗口的提交按钮，保存修改记录。

本例中，我们将使用信件字段来记录备注，但是可以用另外一个选项创建一个多行文本类型的动态字段，在下面的行中上述剩余的字段是我们之前定义好的动态字段。

请注意本窗口中的所有动态字段都有“DynamicField\_”前缀如“DynamicField\_Title”，不要与工单标题的字段“Title标题”搞混淆了，

创建如下活动对话框：

- “记录需求” (AD1)

此活动对话框包含了订单所需的所有数据字段(标题、作者与ISBN号), 以及一个标识审批状态的字段“审批”。

- “审批拒绝” (AD2)

此活动对话框拥有一个用于注释的字段 (信件)与一个选择“审批拒绝”状态的字段“审批拒绝”。

- “通过审批” (AD3)

只需要一个状态字段, 选项为“通过审批”。

- “采购拒绝” (AD4)

此活动对话框设计来拒绝一笔无效的采购单(书籍缺货)。这里同样需要一个注释字段与一个状态字段选项“采购拒绝”。

- “生成订单” (AD5)

活动对话框中包含了采购涉及的供应商、价格与发货日期字段以及状态字段选项“生成订单”。

- “货品收到” (AD6)

此活动需要为收发室添加一个收到日期的字段, 还需要一个状态字段选项“货品收到”。

为了限制每个活动对话框的状态字段值, 我们需要在Kernel/Config.pm或一个新的Perl文件Kernel/Config/Files中添加一些ACL。

```
$Self->{TicketAcl}->{'P1-AD1-1'} = {
  Properties => {
    Process => {
      ActivityDialogEntityID => ['AD1'],
    },
  },
  Possible => {
    Ticket => {
      DynamicField_Status => ['Approval'],
    },
  },
};

$Self->{TicketAcl}->{'P1-AD2-1'} = {
  Properties => {
    Process => {
      ActivityDialogEntityID => ['AD2'],
    },
  },
  Possible => {
    Ticket => {
      DynamicField_Status => ['Approval denied'],
    },
  },
};

$Self->{TicketAcl}->{'P1-AD3-1'} = {
  Properties => {
    Process => {
      ActivityDialogEntityID => ['AD3'],
    },
  },
  Possible => {
    Ticket => {
      DynamicField_Status => ['Approved'],
    },
  },
};
```

```

$Self->{TicketAcl}->{'P1-AD4-1'} = {
  Properties => {
    Process => {
      ActivityDialogEntityID => ['AD4'],
    },
  },
  Possible => {
    Ticket => {
      DynamicField_Status => ['Order denied'],
    },
  },
};

$Self->{TicketAcl}->{'P1-AD5-1'} = {
  Properties => {
    Process => {
      ActivityDialogEntityID => ['AD5'],
    },
  },
  Possible => {
    Ticket => {
      DynamicField_Status => ['Order placed'],
    },
  },
};

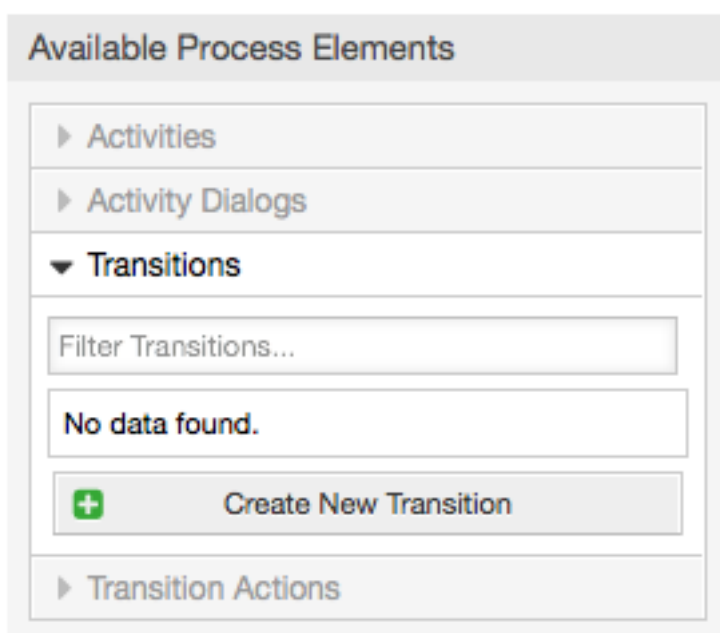
$Self->{TicketAcl}->{'P1-AD6-1'} = {
  Properties => {
    Process => {
      ActivityDialogEntityID => ['AD6'],
    },
  },
  Possible => {
    Ticket => {
      DynamicField_Status => ['Shipment received'],
    },
  },
};

```

### 2.3.3. 转换

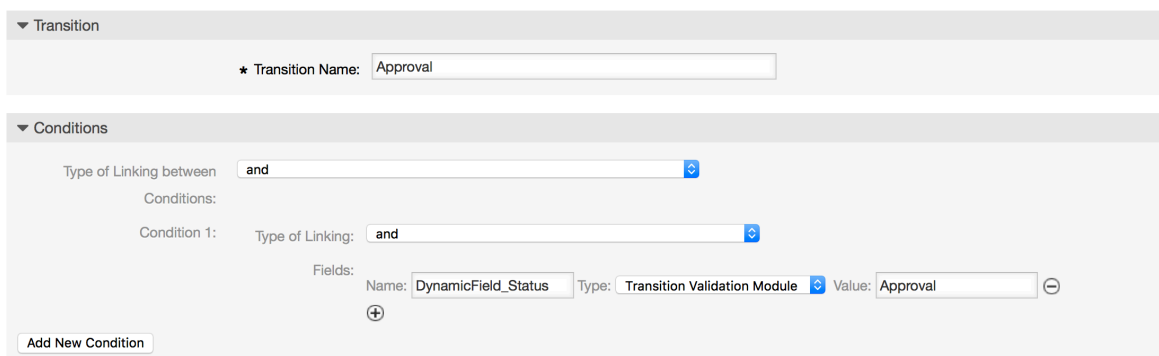
在“可用的流程元素”中点击“转换”，接着点击“创建新的转换”。

图 5.13. 创建新的转换按钮



在弹出窗口填写“转换名称”，接着我们需要配置条件选项，本例中只需要设置一个条件和一个字段，可以设置链接类型为“与”以及字段匹配类型值为“字符串”。

**图 5.14. 创建新转换**



所有条件设置好后，点击提交按钮，保存结果。

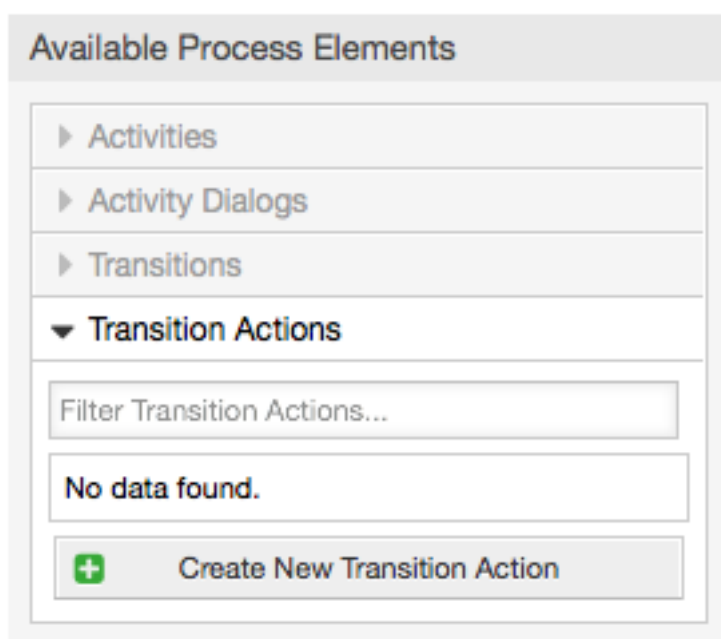
创建如下转换：

- “审批” (T1)  
用于检查状态字段是否被设置为“审批”的转换。
- “审批拒绝” (T2)  
用于检查状态字段是否被设置为“审批拒绝”的转换。
- “通过审批” (T3)  
用于检查状态字段是否被设置为“通过审批”的转换。
- “采购拒绝” (T4)  
用于检查状态字段是否被设置为“采购拒绝”的转换。
- “生成订单” (T5)  
用于检查状态字段是否被设置为“生成订单”的转换。
- “货品收到” (T6)  
用于检查状态字段是否被设置为“货品收到”的转换。

### 2.3.4. 转换操作

点击“可用的流程元素”中的“转换操作”，接着点击“创建新的转换操作”。

图 5.15. 创建新的转换操作按钮



在弹出窗口填写“转换操作名称”和“转换操作模块”，然后添加必填的和可选的参数名和参数值。

所有的转换操作模块位于文件Kernel/System/ProcessManagement/TransitionAction中，下面是本版本中捆绑的转换操作清单：

- DynamicFieldSet动态字段设置
- TicketArticleCreate工单信件创建
- 工单创建
- TicketCustomerSet工单客户设置
- TicketLockSet工单锁定设置
- TicketOwnerSet工单所有者设置
- TicketQueueSet工单队列设置
- TicketResponsibleSet工单负责人设置
- TicketServiceSet工单服务设置
- TicketSLASet工单SLA设置
- TicketStateSet工单状态设置
- TicketTitleSet工单标题设置
- TicketTypeSet工单类型设置

每个模块都拥有其独立与不同的参数。请查阅module documentation 模块文档来了解所有必填与可选参数。

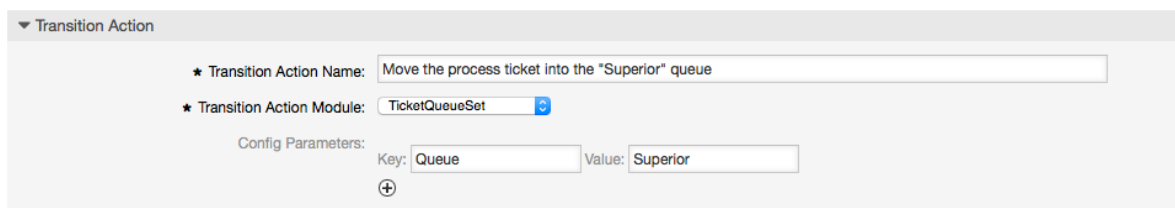
## 注意

从OTRS 4.0.1开始，参数不再强制使用固定值，而是通过格式<OTRS\_Ticket\_property>从原始工单继承。



从OTRS 4.0.6开始支持格式<OTRS\_TICKET\_property>，老格式仍然可以使用，但是会在将来的版本中废弃。

**图 5.16. 创建新的转换操作**



当所有参数与数值都配置好后，点击提交按钮保存变更。

新建如下转换操作：

- “将工单移入‘管理’队列” (TA1)  
此动作预计在转换“审批” (T1)应用后执行。
- “变更工单负责人为‘经理’ ” (TA2)  
此动作在转换“审批” (T1)应用后执行。
- “移动流程工单到‘员工’队列” (TA3)  
当遇到如下情况时执行：
  - 应用了转换“审批拒绝” (T2)
  - 应用了转换 “采购拒绝” (T4)
  - 应用了转换 “货品收到” (T6)
- “变更工单负责人为‘员工’” (TA4)  
当遇到如下情况时执行：
  - 应用了转换“审批拒绝” (T2)
  - 应用了转换 “采购拒绝” (T4)
  - 应用了转换 “货品收到” (T6)
- “将工单转入‘采购’队列” (TA5)  
当应用了转换 “通过审批” (T3)时执行。
- “移动工单至‘邮局’队列” (TA6)  
当应用了“生成订单” (T5)时执行。
- “工单成功关闭” (TA7)  
当遇到如下情况时执行：
  - 应用了转换“货品收到” (T6)
- “工单失败关闭” (TA8)  
当遇到如下情况时执行：
  - 应用了转换“审批拒绝” (T2)

- 应用了转换“采购拒绝”(T4)

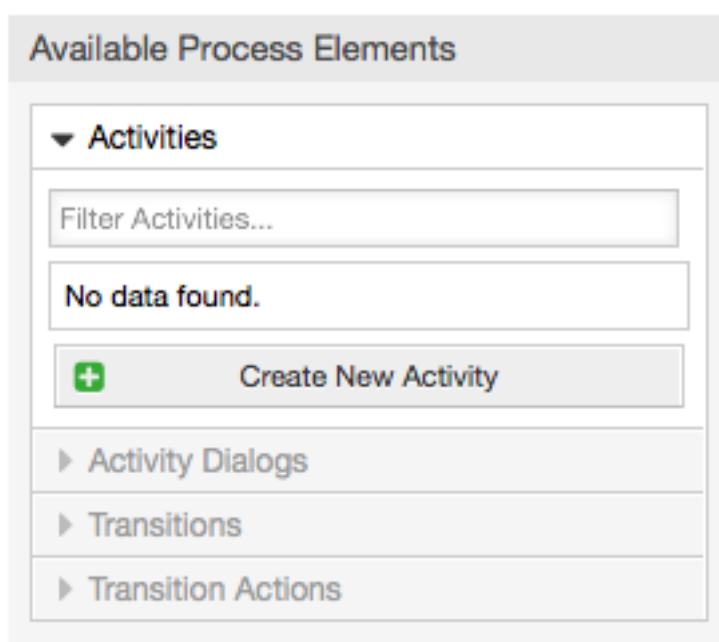
正如你所见，相同的转换操作会在很多地方被执行。因此设计允许转换操作与转换自由连接，是为了更好的复用资源。

### 2.3.5. 活动

我们选择将活动(Activities)看作一个篮子，这个篮子可以包含一个或多个活动对话框(Activity Dialogs)。

点击“可用的流程元素”中的“活动”，接着点击“创建新活动”。

图 5.17. 创建新活动按钮



在打开的弹出窗口中填写“活动名称”，然后仅需从“可用的活动对话框”池中拖动需要的活动对话框，并放到“分配的活动对话框”池中即可。这些对话框会在工单详情窗口按本窗口定义的从上到下、从左到右的顺序显示出来。

在第一个活动中这个顺序特别重要，因为流程开始后唯一显示的就是这个活动的第一个活动对话框。

创建下列活动：

- “记录需求”(A1)  
包含活动对话框“记录需求”(AD1)
- “审批”(A2)  
包含活动对话框“审批拒绝”(AD2)和“通过审批”(AD3)
- “订单”(A3)  
包含活动对话框“采购拒绝”(AD4)与“生成订单”(AD5)
- “即将到达”(A4)  
包含活动对话框“货品收到”(AD6)
- “流程结束”(A5): 这是一个没有活动对话框的活动。它需要被设置在“审批拒绝”、“采购拒绝”或者“货品收到”之后，代表流程结束。

现在我们可以清晰地看到，活动是一个流程工单精确定义的状态。在转换成功后，流程工单会从一个活动移动到另外一个。

### 2.3.6. 书籍采购流程路径

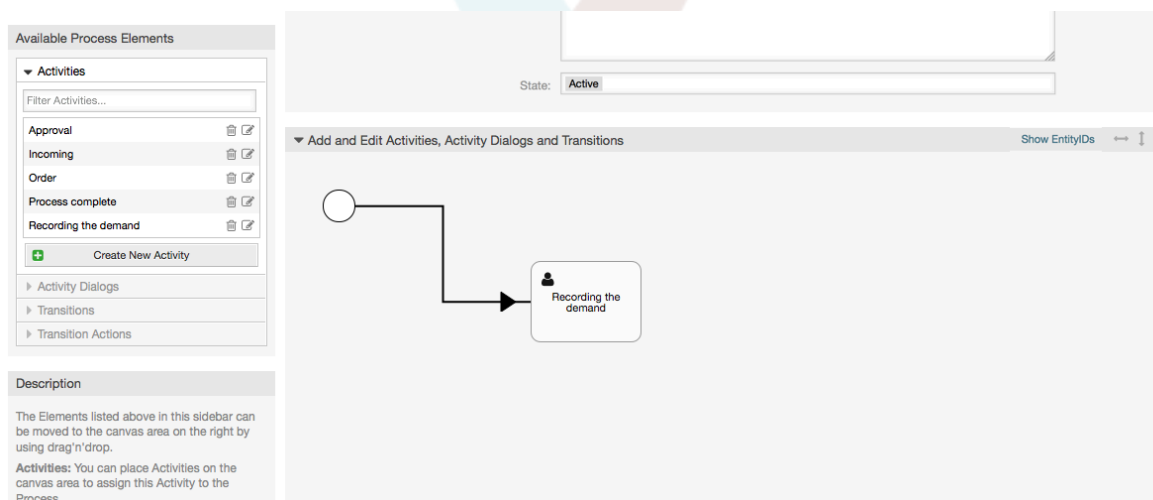
让我们用拼图所缺失的最后一块来结束我们的示例-将流程作为流记录器。在我们的例子中，就是完整的订购 workflow。其他流程可能还有办公用品订购或其他完全不同的流程。

此流程有一个起始点，它包括了初始活动和初始活动对话框。对于任何新书采购，初始活动对话框(第一个活动的第一个活动对话框)是最先显示的。如果这个活动完成并保存，流程工单将被创建并可以根据配置好的 workflow 进行处理。

流程中还包含了流程工单在流程中如何移动的方向。我们称之为“路径”。它由初始活动、一个或多个转换(可能包含转换操作)、以及其他活动组成。

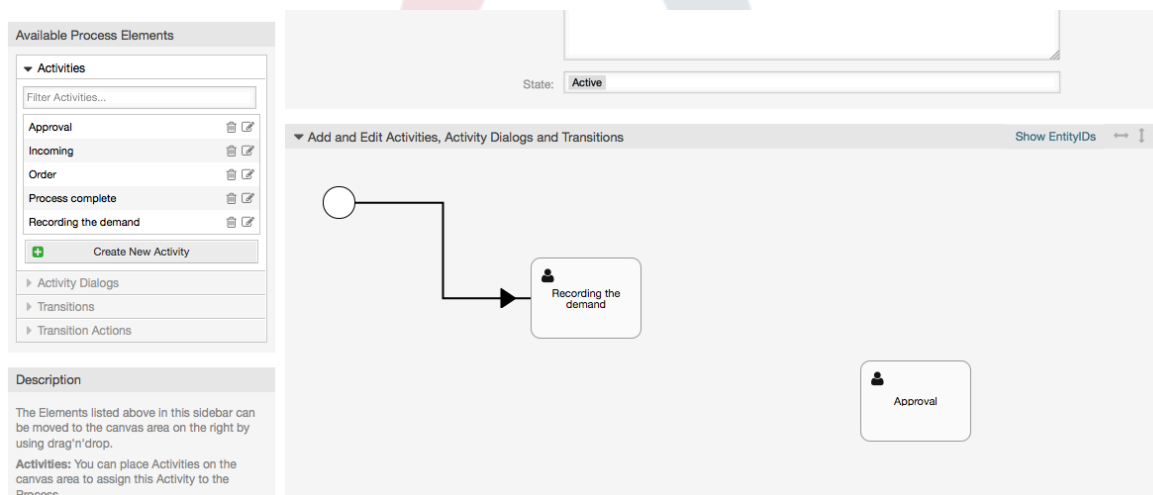
假设该活动已经指派了相关的活动对话框，将其从折叠区域(屏幕的左侧部分)拖放到画布区域(在流程信息下面)。请注意，流程起始(绿色圆圈)会自动放置一个到该活动的箭头。(这是第一个活动，并且其第一个活动对话框会在流程开始后在窗口第一个显示)。

图 5.18. 将第一个活动拖放到画布中



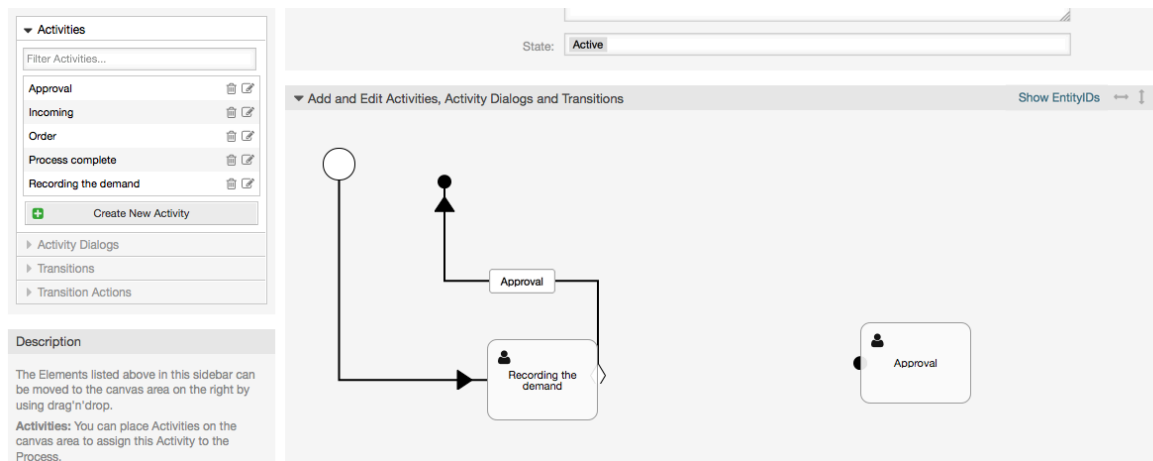
下一步，将另一个活动也拖放到画布中。现在画布中我们有了两个活动。第一个活动连接到起始点，第二个活动还没有连接。你可以将鼠标移动到活动上，会显示它们自己的活动对话框。

图 5.19. 拖动第二个活动到画布中



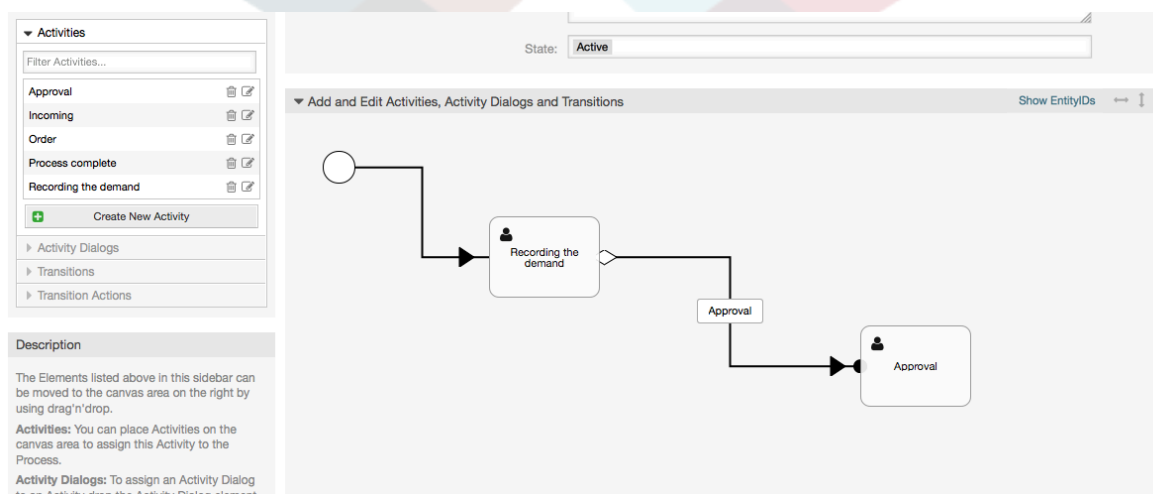
接下来让我们在这两个活动间创建“路径”（连接），创建路径需要使用转换。在折叠区域点击“转换”并拖动一个转换到第一个活动里面放下。注意活动变化了颜色，表示附上了转换。转换一放下，转换箭头的终点就放置在邻近流程的起始点位置。拖动转换的箭头终点到另一活动里面来创建这两个活动的连接。

图 5.20. 拖放一个转换到画布上



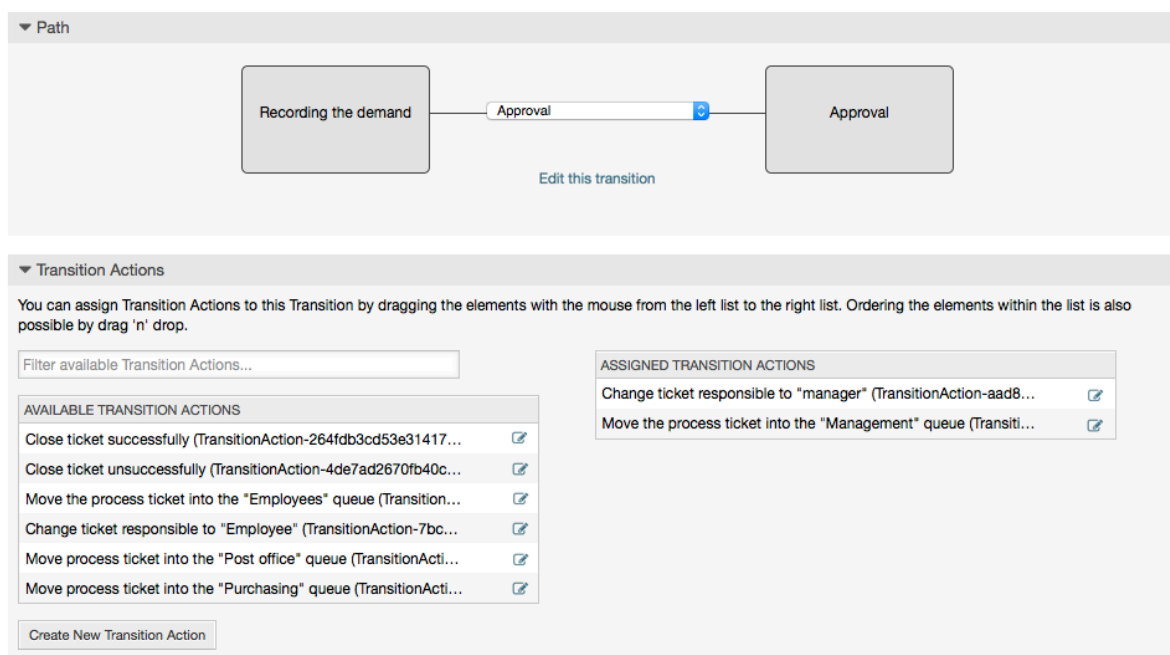
现在操作之间的“路径”已经定义好了，接下来我们需要分配转换操作到这个转换上，在画布中双击转换标签，这会弹出新窗口。

图 5.21. 使用转换连接活动



从‘可用的转换操作’池中拖动所需的转换活动，并将其放置在‘分配的转换操作’池中，点击提交按钮。

图 5.22. 分配转换操作



接着回到流程编辑主窗口，点击画布下的保存按钮，保存所有其他的修改。

添加下列活动、转换和转换操作以完成“路径”：

记录需求直到“批准”

- 起始点：活动“记录需求” (A1)
- 可能的转换：“审批” (T1)
  - 如果这个活动的条件满足，该工单将移动到活动“审批” (A2)
  - 此外，还会执行下列的转换操作：
    - “将工单移入‘管理’队列” (TA1)
    - “变更工单负责人为‘经理’ ” (TA2)

活动：“记录需求” (A1)是流程工单预定义步骤，以确认转换：“审批” (T1)的可能性。如果适用，工单将移动到下一个活动：“审批” (A2)，并且执行转换操作“移动到‘管理’队列” (TA1)和“变更工单负责人为‘经理’ (TA2)。在活动“审批” (A2)中，活动对话框“审批拒绝” (AD2) 和“通过审批” (AD3) 可用。

审批

- 起始点：活动“审批” (A2)
- 可能的转换：
  - “审批拒绝” (T2)
    - 如果匹配这个，流程工单就移动到活动“流程结束” (A5)。
  - 此外，还会执行下列的转换操作：
    - “移动流程工单到‘员工’队列” (TA3)
    - “变更工单负责人为‘员工’” (TA4)

- “工单失败关闭” (TA8)
- “通过审批” (T3)
  - 如果匹配这个，流程工单就移动到活动“订购” (A3)。
  - 此外，还会执行下面的转换操作：
    - “将工单转入‘采购’队列” (TA5)

我们可以看到，定义流程工单的当前活动有一个或多个转换的可能性，转换有一个目标活动（和可能的一个或多个转换操作）。

#### 订单

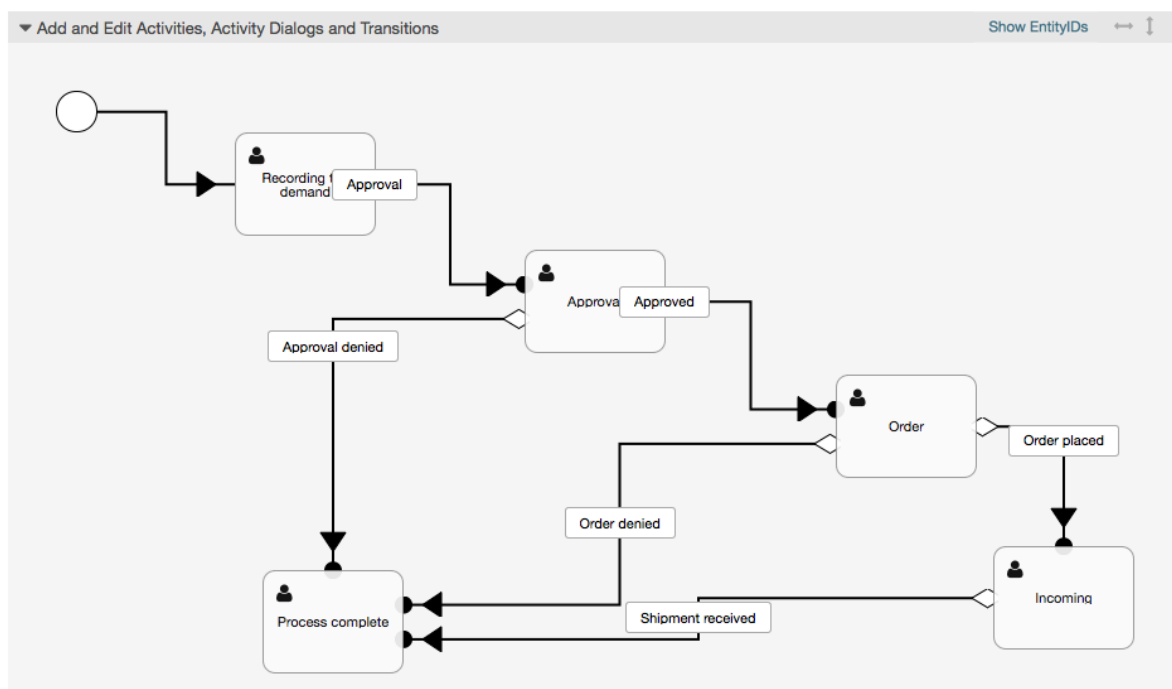
- 起始点：活动“订购” (A3)
- 可能的转换：
  - “采购拒绝” (T4)
    - 如果匹配这个，流程工单就移动到活动“流程结束” (A5)。
  - 此外，还会执行下列的转换操作：
    - “移动流程工单到‘员工’队列” (TA3)
    - “设置工单负责人为‘员工’” (TA4)
    - “工单失败关闭” (TA8)
  - “生成订单” (T5)
    - 如果匹配这个，流程工单就移动到活动“即将到达”
    - 此外，还会执行下面的转换操作：
      - “移动工单至‘邮局’队列” (TA6)

#### 即将到达

- 起始点：活动“即将到达” (A4)
- 可能的转换：
  - “货品收到” (T6)
    - 如果匹配这个，流程工单就移动到活动“流程结束” (A5)。
  - 此外，还会执行下列的转换操作：
    - “移动流程工单到‘员工’队列” (TA3)
    - “设置工单负责人为‘员工’” (TA4)
    - “工单成功关闭” (TA7)

书籍采购流程完整路径看上去就像这样：

图 5.23. 书籍采购完整的流程路径



在完成流程路径后，请点击画布正文的“保存”按钮，然后再点击“同步所有流程”按钮。这将收集所有的流程信息到数据库并创建一个缓存文件（用Perl语言）。这个缓存文件就是实际的流程配置，系统将用它来创建或处理流程工单。

任何流程变更(通过GUI界面)都需要重新同步缓存文件，才可以确保将变更映射到系统。

也可以从一个YAML文件导入整个流程，但是在导入前，仍然需要创建每个流程需要的所有动态字段、用户、队列等等。

注意：如果流程需要使用ACL，这些ACL也需要手动设置好。

下面是书籍采购流程示例的完整YAML文件内容：

```

---
Activities:
  A1:
    ActivityDialogs:
      - AD1
    ChangeTime: 2012-11-23 14:49:22
    Config:
      ActivityDialog:
        1: AD1
    CreateTime: 2012-11-23 11:49:38
    EntityID: A1
    ID: 151
    Name: 记录需求
  A2:
    ActivityDialogs:
      - AD2
      - AD3
    ChangeTime: 2012-12-13 00:55:12
    Config:
      ActivityDialog:
        1: AD2
        2: AD3
    CreateTime: 2012-11-23 11:50:11
    EntityID: A2
    ID: 152
    Name: 审批
  A3:

```

```
ActivityDialogs:
- AD4
- AD5
ChangeTime: 2012-11-23 18:12:14
Config:
  ActivityDialog:
    1: AD4
    2: AD5
CreateTime: 2012-11-23 11:50:35
EntityID: A3
ID: 153
Name: 采购
A4:
ActivityDialogs:
- AD6
ChangeTime: 2012-11-23 18:12:35
Config:
  ActivityDialog:
    1: AD6
CreateTime: 2012-11-23 11:51:00
EntityID: A4
ID: 154
Name: 即将到达
A5:
ActivityDialogs: []
ChangeTime: 2012-11-23 11:51:33
Config: {}
CreateTime: 2012-11-23 11:51:33
EntityID: A5
ID: 155
Name: 流程结束
ActivityDialogs:
AD1:
ChangeTime: 2012-12-06 02:16:21
Config:
  DescriptionLong: ''
  DescriptionShort: 记录需求
  FieldOrder:
    - DynamicField_Author
    - DynamicField_ISBN
    - DynamicField_Title
    - DynamicField_Status
  Fields:
    DynamicField_Author:
      DefaultValue: ''
      DescriptionLong: ''
      DescriptionShort: ''
      Display: 1
    DynamicField_ISBN:
      DefaultValue: ''
      DescriptionLong: ''
      DescriptionShort: ''
      Display: 1
    DynamicField_Status:
      DefaultValue: ''
      DescriptionLong: ''
      DescriptionShort: ''
      Display: 1
    DynamicField_Title:
      DefaultValue: ''
      DescriptionLong: ''
      DescriptionShort: ''
      Display: 1
  Interface:
    - AgentInterface
  Permission: ''
  RequiredLock: 0
  SubmitAdviceText: ''
  SubmitButtonText: ''
CreateTime: 2012-11-23 14:34:43
EntityID: AD1
ID: 154
```



```
Name: 记录需求
AD2:
ChangeTime: 2012-11-23 14:57:41
Config:
  DescriptionLong: ''
  DescriptionShort: 审批拒绝
  FieldOrder:
    - Article
    - DynamicField_Status
  Fields:
    Article:
      Config:
        ArticleType: note-internal
        DefaultValue: ''
        DescriptionLong: ''
        DescriptionShort: ''
        Display: 1
      DynamicField_Status:
        DefaultValue: ''
        DescriptionLong: ''
        DescriptionShort: ''
        Display: 1
    Interface:
      - AgentInterface
  Permission: ''
  RequiredLock: 0
  SubmitAdviceText: ''
  SubmitButtonText: 拒绝请求
CreateTime: 2012-11-23 14:36:39
EntityID: AD2
ID: 155
Name: Approval denied
AD3:
ChangeTime: 2012-12-14 03:14:23
Config:
  DescriptionLong: ''
  DescriptionShort: 通过审批
  FieldOrder:
    - DynamicField_Status
  Fields:
    DynamicField_Status:
      DefaultValue: ''
      DescriptionLong: ''
      DescriptionShort: ''
      Display: 1
    Interface:
      - AgentInterface
  Permission: ''
  RequiredLock: 0
  SubmitAdviceText: ''
  SubmitButtonText: 同意请求
CreateTime: 2012-11-23 14:37:35
EntityID: AD3
ID: 156
Name: 通过审批
AD4:
ChangeTime: 2012-11-23 14:58:52
Config:
  DescriptionLong: ''
  DescriptionShort: 采购拒绝
  FieldOrder:
    - Article
    - DynamicField_Status
  Fields:
    Article:
      Config:
        ArticleType: note-internal
        DefaultValue: ''
        DescriptionLong: ''
        DescriptionShort: ''
        Display: 1
      DynamicField_Status:
```

```
    DefaultValue: ''
    DescriptionLong: ''
    DescriptionShort: ''
    Display: 1
Interface:
  - AgentInterface
Permission: ''
RequiredLock: 0
SubmitAdviceText: ''
SubmitButtonText: 拒绝采购
CreateTime: 2012-11-23 14:38:48
EntityID: AD4
ID: 157
Name: 采购拒绝
AD5:
ChangeTime: 2012-12-06 02:20:12
Config:
  DescriptionLong: ''
  DescriptionShort: 生成订单
  FieldOrder:
    - DynamicField_DeliveryDate
    - DynamicField_Price
    - DynamicField_Supplier
    - DynamicField_Status
  Fields:
    DynamicField_DeliveryDate:
      DefaultValue: ''
      DescriptionLong: ''
      DescriptionShort: ''
      Display: 1
    DynamicField_Price:
      DefaultValue: ''
      DescriptionLong: ''
      DescriptionShort: ''
      Display: 1
    DynamicField_Status:
      DefaultValue: ''
      DescriptionLong: ''
      DescriptionShort: ''
      Display: 1
    DynamicField_Supplier:
      DefaultValue: ''
      DescriptionLong: ''
      DescriptionShort: ''
      Display: 1
Interface:
  - AgentInterface
Permission: ''
RequiredLock: 0
SubmitAdviceText: ''
SubmitButtonText: Place Order
CreateTime: 2012-11-23 14:41:28
EntityID: AD5
ID: 158
Name: 生成订单
AD6:
ChangeTime: 2012-11-23 14:42:43
Config:
  DescriptionLong: ''
  DescriptionShort: 货品到达
  FieldOrder:
    - DynamicField_DateOfReceipt
    - DynamicField_Status
  Fields:
    DynamicField_DateOfReceipt:
      DefaultValue: ''
      DescriptionLong: ''
      DescriptionShort: ''
      Display: 1
    DynamicField_Status:
      DefaultValue: ''
      DescriptionLong: ''
```

```
    DescriptionShort: ''
    Display: 1
    Interface:
      - AgentInterface
    Permission: ''
    RequiredLock: 0
    SubmitAdviceText: ''
    SubmitButtonText: ''
    CreateTime: 2012-11-23 14:42:43
    EntityID: AD6
    ID: 159
    Name: 货品到达
Process:
  Activities:
    - A1
    - A2
    - A3
    - A4
    - A5
  ChangeTime: 2012-12-06 02:31:59
  Config:
    Description: 采购一本书的流程
    Path:
      A1:
        T1:
          ActivityEntityID: A2
          TransitionAction:
            - TA2
            - TA1
        A2:
          T2:
            ActivityEntityID: A5
            TransitionAction:
              - TA3
              - TA4
              - TA8
          T3:
            ActivityEntityID: A3
            TransitionAction:
              - TA5
        A3:
          T4:
            ActivityEntityID: A5
            TransitionAction:
              - TA3
              - TA4
              - TA8
          T5:
            ActivityEntityID: A4
            TransitionAction:
              - TA6
        A4:
          T6:
            ActivityEntityID: A5
            TransitionAction:
              - TA3
              - TA4
              - TA7
      A5: {}
    StartActivity: A1
    StartActivityDialog: AD1
  CreateTime: 2012-11-23 11:45:12
  EntityID: P1
  ID: 94
  Layout:
    A1:
      left: 172
      top: 63
    A2:
      left: 402
      top: 156
    A3:
```

```
left: 649
top: 255
A4:
  left: 774
  top: 391
A5:
  left: 194
  top: 410
Name: 书籍采购
State: Active
StateEntityID: S1
TransitionActions:
  - TA1
  - TA2
  - TA3
  - TA4
  - TA8
  - TA5
  - TA3
  - TA4
  - TA8
  - TA6
  - TA3
  - TA4
  - TA7
Transitions:
  - T1
  - T2
  - T3
  - T4
  - T5
  - T6
TransitionActions:
TA1:
  ChangeTime: 2012-11-23 16:01:37
  Config:
    Queue: Management
    Module: Kernel::System::ProcessManagement::TransitionAction::TicketQueueSet
  CreateTime: 2012-11-23 15:50:59
  EntityID: TA1
  ID: 61
  Name: 将工单转入'管理'队列"
TA2:
  ChangeTime: 2012-11-23 16:02:12
  Config:
    Responsible: 经理
    Module: Kernel::System::ProcessManagement::TransitionAction::TicketResponsibleSet
  CreateTime: 2012-11-23 15:58:22
  EntityID: TA2
  ID: 62
  Name: 变更工单负责人为'经理'
TA3:
  ChangeTime: 2012-11-24 14:27:02
  Config:
    Queue: 员工
    Module: Kernel::System::ProcessManagement::TransitionAction::TicketQueueSet
  CreateTime: 2012-11-23 16:02:54
  EntityID: TA3
  ID: 63
  Name: 将工单转入'经理'队列"
TA4:
  ChangeTime: 2012-11-23 16:04:06
  Config:
    Responsible: 员工
    Module: Kernel::System::ProcessManagement::TransitionAction::TicketResponsibleSet
  CreateTime: 2012-11-23 16:04:06
  EntityID: TA4
  ID: 64
```

```
Name: 变更工单负责人为'员工'
TA5:
ChangeTime: 2012-12-06 02:18:34
Config:
  Config:
    Queue: 采购
    Module: Kernel::System::ProcessManagement::TransitionAction::TicketQueueSet
  CreateTime: 2012-11-23 16:04:54
  EntityID: TA5
  ID: 65
  Name: 将工单转入'采购'队列"
TA6:
ChangeTime: 2012-12-06 02:18:48
Config:
  Config:
    Queue: 邮局
    Module: Kernel::System::ProcessManagement::TransitionAction::TicketQueueSet
  CreateTime: 2012-11-23 16:06:20
  EntityID: TA6
  ID: 66
  Name: 将工单转入'邮局'队列"
TA7:
ChangeTime: 2012-12-06 02:29:55
Config:
  Config:
    State: 成功关闭
    Module: Kernel::System::ProcessManagement::TransitionAction::TicketStateSet
  CreateTime: 2012-12-06 02:29:27
  EntityID: TA7
  ID: 67
  Name: Close ticket successfully
TA8:
ChangeTime: 2012-12-06 02:31:12
Config:
  Config:
    State: closed unsuccessful
    Module: Kernel::System::ProcessManagement::TransitionAction::TicketStateSet
  CreateTime: 2012-12-06 02:31:12
  EntityID: TA8
  ID: 68
  Name: 工单失败关闭
Transitions:
T1:
ChangeTime: 2012-11-23 15:12:20
Config:
  Condition:
    1:
      Fields:
        DynamicField_Status:
          Match: Approval
          Type: String
      Type: and
    ConditionLinking: and
  CreateTime: 2012-11-23 11:53:52
  EntityID: T1
  ID: 94
  Name: Approval
T2:
ChangeTime: 2012-11-23 15:12:50
Config:
  Condition:
    1:
      Fields:
        DynamicField_Status:
          Match: Approval denied
          Type: String
      Type: and
    ConditionLinking: and
  CreateTime: 2012-11-23 11:54:26
  EntityID: T2
  ID: 95
  Name: Approval denied
```

```
T3:
ChangeTime: 2012-11-23 15:13:29
Config:
Condition:
  1:
    Fields:
      DynamicField_Status:
        Match: 通过审批
        Type: String
    Type: and
  ConditionLinking: and
CreateTime: 2012-11-23 11:54:54
EntityID: T3
ID: 96
Name: 通过审批
T4:
ChangeTime: 2012-11-23 15:14:08
Config:
Condition:
  1:
    Fields:
      DynamicField_Status:
        Match: 采购拒绝
        Type: String
    Type: and
  ConditionLinking: and
CreateTime: 2012-11-23 11:55:25
EntityID: T4
ID: 97
Name: Order denied
T5:
ChangeTime: 2012-11-23 18:30:33
Config:
Condition:
  1:
    Fields:
      DynamicField_Status:
        Match: 生成订单
        Type: String
    Type: and
  ConditionLinking: and
CreateTime: 2012-11-23 11:56:15
EntityID: T5
ID: 98
Name: 生成订单
T6:
ChangeTime: 2012-11-23 15:15:30
Config:
Condition:
  1:
    Fields:
      DynamicField_Status:
        Match: 货品到达
        Type: String
    Type: and
  ConditionLinking: and
CreateTime: 2012-11-23 11:56:48
EntityID: T6
ID: 99
Name: 货品到达
```

## 2.4. 流程配置参考

### 2.4.1. 流程

一个流程就是工作流/流程的路径。路径上的路标可以是活动或转换（后面再讨论转换）。

### 2.4.1.1. 流程配置

流程配置可以放在文件Kernel/Config.pm中，但强烈推荐创建一个新文件如Kernel/Config/Files/MyProcess.pm。注意图形界面生成的文件为Kernel/Config/File/ZZZProcessManagement，请一定不要使用这个文件名，否则在同步流程时会被覆盖。让我们来看一个示例流程配置（从流程缓存文件）：

```
$Self->{'Process'} = {
  'P1' => {
    Name           => 'Book order',
    CreateTime     => '16-02-2012 13:37:00',
    CreateBy      => '1',
    ChangeTime    => '17-02-2012 13:37:00',
    ChangeBy      => '1',
    State         => 'Active',
    StartActivity => 'A1',
    StartActivityDialog => 'AD1',
    Path => {
      'A1' => {
        'T1' => {
          ActivityEntityID => 'A2',
        },
      },
      'A2' => {
        'T2' => {
          ActivityEntityID => 'A3',
        },
      },
    },
  },
  'P2' => {
    Name           => 'IT order',
    CreateTime     => '26-02-2012 13:37:00',
    CreateBy      => '1',
    ChangeTime    => '27-02-2012 13:37:00',
    ChangeBy      => '1',
    State         => 'Active',
    StartActivity => 'A2',
    StartActivityDialog => 'AD2',
    Path => {
      'A2' => {
        'T3' => {
          ActivityEntityID => 'A4',
        },
      },
    },
  },
};
```

### 2.4.1.2. 名称

流程的名称，服务人员在创建流程工单时可以选择的流程名。

### 2.4.1.3. 创建时间

流程创建的时间。

### 2.4.1.4. 创建人

创建流程的用户UID。

### 2.4.1.5. 修改时间

流程被修改的时间。

### 2.4.1.6. 修改人

最后修改流程的用户UID。

### 2.4.1.7. 状态

定义的流程状态。可用值为：

- ‘激活’-可以在新的流程工单中使用的所有流程。
- ‘逐渐消失’-不能在新工单中使用，但是现有工单仍然在使用的流程。
- ‘未激活’-流程没有激活，不能在新工单或现有工单中使用。

### 2.4.1.8. 起始活动

当创建一个新的流程工单时，必须定义一个‘起始活动’。一旦创建了工单，这个活动就被设置并作为第一个转换检查的基础。

### 2.4.1.9. 起始活动对话框

对新的流程工单，必须定义一个‘起始活动对话框’。这将在创建一个流程工单时显示（在选定流程后）。在这个时间点，工单还不存在，工单会在‘起始活动对话框’提交之后创建。

### 2.4.1.10. 路径

路径包含了当前流程的活动及活动间可能的转换。还有在转换时发生的转换操作。这个方式控制了流程工单发生的路径。例如：

```
'A1' => {
  'T1' => {
    ActivityEntityID => 'A2',
  },
  'T2' => {
    ActivityEntityID => 'A3',
  },
  'T3' => {
    ActivityEntityID => 'A4',
    TransitionAction => ['TA1', 'TA2'],
  },
},
```

如果一个流程工单在活动 'A1'，它有三个可能的路径来获得另一活动。在转换'T1'到'T3'，定义了流程工单必须满足移动（转换）到另一活动的条件。

如果在这个案例中流程工单的所有值和转换'T2'所需的动态字段都是正确的，这个工单就会从活动'A1'移动到'A3'。在提交了一个活动对话框之后，或者工单有了其它任何修改，工单会从当前活动检查可能的转换。如果有多个可能的转换，就使用第一个（基于TransitionID的数字排序）。

此外，还可以在路径配置中给转换分配转换操作。这些转换操作就是在成功转换后执行的模块。在示例流程中它们用数组表单方式分配，我们会在后面讨论这些细节。

## 2.4.2. 活动

一个活动包含一个或多个活动对话框并模拟流程中的一个‘步骤’。当前活动的所有活动对话框都会在工单详情窗口中显示，直到一个转换条件满足后才能使用它们。

### 2.4.2.1. 活动配置

让我们看一个活动配置样例：



```

$self->{'Process::Activity'} =
{
  'A1' => {
    Name      => 'Activity 1 optional',
    CreateTime => '16-02-2012 13:37:00',
    CreateBy  => '1',
    ChangeTime => '17-02-2012 13:37:00',
    ChangeBy  => '1',
    ActivityDialog => {
      1 => 'AD1',
    },
  },
  'A2' => {
    Name      => 'Activity 2 optional',
    CreateTime => '16-02-2012 13:37:00',
    CreateBy  => '1',
    ChangeTime => '17-02-2012 13:37:00',
    ChangeBy  => '1',
    ActivityDialog => {
      1 => 'AD5',
      2 => 'AD6',
      3 => 'AD1',
    },
  },
},
};

```

### 2.4.2.2. 名称

活动的名称。

### 2.4.2.3. 创建时间

活动创建的时间。

### 2.4.2.4. 创建人

创建该活动用户UID。

### 2.4.2.5. 修改时间

最后修改的时间。

### 2.4.2.6. 修改人

最后修改该流程的用户UID。

### 2.4.2.7. 活动对话框

活动对话框包含了这个活动的可用活动对话框列表。当前活动的所有活动对话框都会显示在工单详情窗口中。它们显示的顺序在配置中设置，在这里'AD5'显示在'AD6'和'AD1'的前面。

## 2.4.3. 活动对话框

一个活动对话框是一个特别的窗口，能在不同的活动中使用。

### 2.4.3.1. 活动对话框配置

让我们看一个活动配置样例：

```

$self->{'Process::ActivityDialog'} = {
  'AD1' => {
    Name      => 'ActivityDialog 1 optional',

```

```

DescriptionShort => 'Basic info',
DescriptionLong => 'Please insert the necessary basic information for IT orders',
CreateTime      => '28-02-2012 13:37:00',
CreateBy       => '1',
ChangeTime     => '29-02-2012 13:37:00',
ChangeBy      => '1',
Fields => {
  PriorityID => {
    DescriptionShort => 'Priority ID',
    DescriptionLong  => 'Enter the priority here',
    Display          => 2,
  },
},
FieldOrder    => [ 'PriorityID' ],
SubmitAdviceText => 'Note: If you submit the form...',
SubmitButtonText => 'Send request',
},
'AD2' => {
  Name        => 'ActivityDialog 2 optional',
  DescriptionShort => 'Basic info',
  DescriptionLong  => 'Please insert the necessary basic information for Book
orders',
  CreateTime     => '28-02-2012 13:37:00',
  CreateBy      => '1',
  ChangeTime    => '29-02-2012 13:37:00',
  ChangeBy     => '1',
  Fields => {
    StateID => {
      DescriptionShort => 'State ID',
      DescriptionLong  => 'Enter the state here',
      Display          => 2,
      DefaultValue    => '2',
    },
    Queue => {
      DescriptionShort => 'Queue ID',
      DescriptionLong  => 'Enter the queue here',
      Display          => 2,
      DefaultValue    => 'Raw',
    },
    Title => {
      DescriptionShort => 'Title',
      DescriptionLong  => 'Enter the title here',
      Display          => 1,
      DefaultValue    => 'Default Title',
    },
    DynamicField_Anzahl => {
      DescriptionShort => 'Amount',
      DescriptionLong  => 'Enter the amount here',
      Display          => 2,
      DefaultValue    => '4',
    },
  },
  FieldOrder    => [ 'DynamicField_Anzahl', 'StateID', 'Queue', 'Title' ],
  SubmitAdviceText => 'Note: If you submit the form...',
  SubmitButtonText => 'Send request',
},
};

```

### 2.4.3.2. 名称

活动对话框名称

### 2.4.3.3. 创建时间

活动创建的时间。

### 2.4.3.4. 创建人

创建该活动对话框的用户UID。

### 2.4.3.5. 修改时间

最后修改的时间。

### 2.4.3.6. 修改人

最后修改这个活动对话框的用户UID。

### 2.4.3.7. 字段

包含所有能在活动对话框中显示的字段。目前能使用下列字段：

```
Title
State
StateID
Priority
PriorityID
Lock
LockID
Queue
QueueID
Customer
CustomerID
CustomerNo
CustomerUserID
Owner
OwnerID
Type
TypeID
SLA
SLAID
Service
ServiceID
Responsible
ResponsibleID
PendingTime
DynamicField_$FieldName # for all dynamic fields
```

单个字段配置的样例：

```
StateID => {
  DescriptionShort => 'State ID',
  DescriptionLong  => 'Enter the state here',
  Display          => 2,
  DefaultValue    => '2',
},
```

“信件”字段是一个特殊的例子。如果它出现在一个‘字段’配置中，活动对话框将包含一个完整的富文本编辑器，拥有主题字段并能处理附件。输入的文本会作为信件添加到工单并通过邮件发送。让我们看看信件字段配置的一个例子：

```
Article => {
  DescriptionShort => 'Please insert your comment here.',
  DescriptionLong  => '',
  Display         => 1,
  Config          => {
    ArticleType => 'note-internal',
    LabelSubject => '',
    LabelBody   => '',
  },
},
```

让我们看看这个字段的配置选项：

#### 2.4.3.7.1. 简短描述

(可选) 显示在字段标题下方的简短描述。

#### 2.4.3.7.2. 详细描述

(可选) 鼠标移到到字段上时显示的详细描述信息，比如怎样填写这个字段的建议。

#### 2.4.3.7.3. 显示

控制这个字段是否显示和/或必填，可能的值有：

- '0'：字段不可见。这在自动设置字段值时有用。这种情况下会保存配置的默认值。
- '1'：字段可见，但是可选的。
- '2'：字段可见，且是必填的。下列的字段只能为不可见或必填：

```
QueueID
Queue
State
StateID
Lock
LockID
Priority
PriorityID
Type
TypeID
```

这个字段配置为可选的，并且用户没有提交值，当用户提交活动对话框时将使用默认值保存。

#### 2.4.3.7.4. 默认值

对于有'ID'的字段（如QueueID、OwnerID），默认值是这个字段在数据库中的ID。对于其它没有'ID'的字段（如队列、所有者），默认值必须包含这个字段自己的值。例如：

```
Queue => {
  DescriptionShort => 'Queue',
  DescriptionLong  => 'Enter the queue here',
  Display          => 2,
  DefaultValue     => 'Raw',
},
```

#### 2.4.3.8. 字段顺序

在这里配置字段显示的顺序。重要：不可见字段也必须在这里配置，因为只有配置了的字段才会考虑保存，没有配置的字段的不会被保存。

#### 2.4.3.9. 提交建议文本

(可选) 显示在提交按钮右上方的文本，提供附加的帮助或建议信息。

#### 2.4.3.10. 提交按钮文本

(可选) 提交按钮的可定制文本。

### 2.4.4. 转换

一个转换决定（基于配置的条件）使用流程中的哪一个路径，例如：一个流程工单移动到哪个活动对话框。

### 2.4.4.1. 转换配置

让我们看一个转换配置的样例：

```
$Self->{'Process::Transition'} = {
  'T1' => {
    Name => 'Transition 1',
    CreateTime => '14-03-2012 13:37:00', # optional
    CreateBy => '1', # optional
    ChangeTime => '15-03-2012 13:37:00', # optional
    ChangeBy => '15-03-2012 13:37:00', # optional
    Condition => {
      Cond1 => {
        Fields => {
          StateID => {
            Type => 'String',
            Match => '1',
          },
        },
      },
    },
  },
  'T2' => {
    Name => 'Transition 2 optional',
    CreateTime => 'DATE', # optional
    CreateBy => 'USERID', # optional
    ChangeTime => 'DATE', # optional
    ChangeBy => 'USERID', # optional
    Condition => {
      Cond1 => {
        Queue => 'Raw',
        DynamicField_Farbe => '2',
        DynamicField_Anzahl => '1',
      },
    },
  },
},
};
```

### 2.4.4.2. 名称

转换的名称。

### 2.4.4.3. 创建时间

创建转换的时间。

### 2.4.4.4. 创建人

创建转换的用户UID。

### 2.4.4.5. 修改时间

最后修改转换的时间。

### 2.4.4.6. 修改人

最后修改转换的用户UID。

### 2.4.4.7. 条件

包含这个转换生效所需的所有条件。例如：

```
Condition => {
  Type => 'and',
```

```

Cond1 => {
  Type => 'and',
  Fields => {
    StateID => {
      Type => 'String',
      Match => '1',
    },
    DynamicField_Marke => {
      Type => 'String',
      Match => 'VW',
    },
  },
},
Cond2 => {
  Type => 'and',
  Fields => {
    Queue => {
      Type => 'String',
      Match => 'Raw',
    },
  },
},
},
},

```

让我们看看这个条件配置的详细信息。

#### 2.4.4.7.1. 类型（条件间）

指定不同的条件元素互相连接的方式，可能的值有：

- ‘与’：这是默认类型。要使转换生效必须满足所有的条件。
- ‘或’：至少需要满足一个条件。
- ‘异或’：必须恰好满足一个条件，不能满足多个条件。

#### 2.4.4.7.2. Cond1

这是示例条件的名称。可以随便选择条件名称。条件按排好的顺序评估。

#### 2.4.4.7.3. 类型（条件内字段）

指定这个条件内单个字段的测试结果互相连接的方式，可能的值有：

- ‘与’：这是默认类型。这个条件内的所有字段测试结果必须都匹配。
- ‘或’：至少需要匹配一个字段。
- ‘异或’：必须恰好匹配一个字段的测试结果，不能匹配多个。

#### 2.4.4.7.4. 字段

指定要测试其值的特殊字段。在我们的例子中：

```

Fields => {
  StateID => {
    Type => 'String',
    Match => '1',
  },
},

```

#### 2.4.4.7.5. StateID

字段名的例子。可以使用下列工单字段：

```

Title
State

```

```

StateID
Priority
PriorityID
Lock
LockID
Queue
QueueID
Customer
CustomerID
CustomerNo
CustomerUserID
Owner
OwnerID
Type
TypeID
SLA
SLAID
Service
ServiceID
Responsible
ResponsibleID
DynamicField_{$FieldName} # for all DynamicFields

```

当测试一个含有‘ID’（如SLAID）的字段时，这个字段的数据库ID将用于测试，而其它字段则使用实际值来测试。

#### 2.4.4.7.6. 类型

确定字段测试的类型，可能的值有：

- ‘String字符串’：比较字段值和在与‘Match’中指定的字符串，如果完全相同就是匹配的。
- ‘Hash哈希’：比较字段值的哈希值和在与‘Match’中指定的哈希值。所有的哈希值必须一致才是匹配的。
- ‘Array数组’：比较字段值的数组和在‘Match’中指定的数组。两个数据的清单值必须一致才是匹配的。
- ‘Regex正则表达式’：使用一个正则表达式来测试字段值，重点注意的是‘Match’包含`qr{}`作为基本条件。实际的正则表达式是在大括号中间。
- ‘Module模块’：允许你使用一个Perl模块来进行条件检查。如果模块返回1，检查是匹配的。你可以在文件`Kernel/System/ProcessManagement/TransitionValidation/ValidateDemo.pm`中找到一个示例模块。

### 2.4.5. 转换操作

转换操作是成功应用转换后触发的操作（当一个流程工单从一个活动移动到另一活动时）。这些转换操作用来对工单执行不同的变更，如变更工单的队列或所有者，你还可以创建你自己的转换操作来执行其它的复杂变更。

#### 2.4.5.1. 转换操作配置

让我们看一个转换配置的样例：

```

$self->{'Process::TransitionAction'} = {
    'TA1' => {
        Name => 'Queue Move',
        Module => 'Kernel::System::ProcessManagement::TransitionAction::TicketQueueSet',
        Config => {
            Queue => 'Junk',
            UserID => 123,
        },
    },
};

```

### 2.4.5.2. 名称

转换操作的名称。

### 2.4.5.3. 模块

指定使用的Perl模块。

### 2.4.5.4. 配置

这个参数包含模块所需的所有设置。它的内容取决于使用的特定的转换操作模块。请参阅单独的模块文档以了解详细信息。在我们的示例中，只是必须指定队列。不过我们还需发送UserID参数。转换操作将使用给定的UserID来模拟用户执行。

转换操作在“配置”参数中UserID的用法可在所有的转换操作中使用（从OTRS 3.2.4版本之后）。在本示例中特别重要的是触发转换操作的用户（也就是UserID为123的用户）有没有移动工单到队列'junk'的权限。

### 2.4.5.5. 重复使用转换操作模块

要重复使用转换操作模块，只需在配置中指定多个转换操作。例如：

```
$Self->{'Process::TransitionAction'} = {
  'TA1' => {
    Name => 'Queue Move Junk',
    Module => 'Kernel::System::ProcessManagement::TransitionAction::TicketQueueSet',
    Config => {
      Queue => 'Junk',
    },
  },
  'TA2' => {
    Name => 'Queue Move Raw',
    Module => 'Kernel::System::ProcessManagement::TransitionAction::TicketQueueSet',
    Config => {
      Queue => 'Raw',
    },
  },
};
```

这里使用了同一个模块来移到一个流程工单到队列'Raw'，还有移到队列'junk'。一个特定转换使用的转换操作由流程配置中的'路径'设置来确定。

### 2.4.5.6. 可用的转换操作

OTRS自带了多个可在你的流程中使用的转换操作。在这里你可以找到它们的文档以及需要如何配置它们。

#### 2.4.5.6.1. DynamicFieldSet动态字段设置

在一个流程工单中设置一个或多个动态字段，示例：

```
$Self->{'Process::TransitionAction'} = {
  'TA1' => {
    Name => 'Set DynamicField MasterSlave to Master and Approved to 1',
    Module => 'Kernel::System::ProcessManagement::TransitionAction::DynamicFieldSet',
    Config => {
      MasterSlave => 'Master',
      Approved => '1',
    },
  },
};
```



```
};
```

‘Name名称’指定配置的转换操作的名称。

‘MasterSlave主从’和‘Approved通过审批’就像动态字段名称一样指定，这个转换操作设置它们的值为（‘Master’和‘1’）。

#### 2.4.5.6.2. TicketArticleCreate工单信件创建

创建一个信件，并用于创建备注或邮件回复。

```
$Self->{'Process::TransitionAction'} = {
  'TA1' => {
    Name => 'Article Create Note Internal',
    Module => 'Kernel::System::ProcessManagement::TransitionAction::TicketArticleCreate',
    Config => {
      ArticleType => 'note-internal', #
      note-external (备注-外部) | phone (电话) | fax (传真) | sms (短信) | ... #
      # 不包括任何电子邮件类型
      SenderType => 'agent', #
      agent (服务人员) | system (系统) | customer (客户)
      ContentType => 'text/plain; charset=ISO-8859-15', # 或
      # 可选的字符集和MIME类型 (如'text/html; charset=UTF-8')
      Subject => 'some short description', # 必
      填项
      Body => 'the message text', # 必
      填项
      HistoryType => 'OwnerUpdate',
      # EmailCustomer (客户邮件) | Move (移动) | AddNote (添加备注) | PriorityUpdate (更新优先级) |
      # WebRequestCustomer (客户网页请求) | ...
      HistoryComment => 'Some free text!',
      From => 'Some Agent <email@example.com>', # 不
      是必填项但有用
      To => 'Some Customer A <customer-a@example.com>', # 不
      是必填项但有用
      Cc => 'Some Customer B <customer-b@example.com>', # 不
      是必填项但有用
      ReplyTo => 'Some Customer B <customer-b@example.com>', # 不
      是必填项
      InReplyTo => '<asdasdasd.12@example.com>', # 不
      是必填项但有用
      References => '<asdasdasd.1@example.com> <asdasdasd.12@example.com>', # 不
      是必填项但有用
      NoAgentNotify => 0, # 如
      果不想发送服务人员通知
      AutoResponseType => 'auto reply',
      # auto reject (自动拒绝) | auto follow up (自动跟进) | auto reply/new ticket (自动回复/新工单) | auto
      # remove (自动移除)
      ForceNotificationToUserID => [ 1, 43, 56 ], # 如
      果想要强制给一些人员发送通知
      ExcludeNotificationToUserID => [ 43, 56 ],
      # 如果想要在发送通知时完全排除一些人员,
      # 清单中的人员也会从信件的收件人中移除,
      # 比ForceNotificationToUserID优先级高
      ExcludeMuteNotificationToUserID => [ 43, 56 ],
      # 与ExcludeNotificationToUserID相同, 但是只有发送通知时移除, 服务人员仍然会显示在信件
      的收件人中。
    },
  },
};
```

‘Name名称’指定这个转换操作的名称。可以随意选择，但名称最好能体现转换操作的用途。

‘ArticleType信件类型’定义创建的信件的类型。可能的值有：电话、传真、短信、WEB请求、备注-内部、备注-外部和备注-报告。

'SenderType发件人类型'定义信件发件人的类型。可能的值有：服务人员、系统、客户。

'ContentType内容类型'定义信件的内容类型。可能的值有：'text/plain; charset=ISO-8859-15纯文本'或其它有效的字符集和MIME类型。

'Subject主题'定义信件主题，必填。

'Body正文'定义信件的内容，必填。

'HistoryType历史类型'定义历史条目的类型。可能的值有：添加备注、归档标志更新、退回、客户更新、服务人员邮件、客户邮件、升级响应时间通知之前、升级响应时间通知开始、升级响应时间结束、升级解决时间通知之前、升级解决时间通知开始、升级解决时间结束、升级更新时间之前、升级更新时间开始、升级更新时间结束、跟进、转发、锁定、循环保护、合并、杂项、转换、新工单、所有者更新、服务人员电话、客户电话、优先级更新、移除、负责人更新、发送服务人员通知、发送答复、发送自动跟进、发送自动拒绝、发送自动回复、发送客户通知、服务更新、设置挂起时间、SLA更新、状态更新、关注、系统请求、工单动态字段更新、添加工单链接、删除工单链接、工时管理、类型更新、解锁、取消关注、客户WEB请求。

'HistoryComment历史注释'定义历史条目的内容。

'From发件人'、'To收件人'、'Cc抄送'和'ReplyTo回复'使用上面指定格式的邮件地址。

'InReplyTo作为答复'和'References参考'使用邮件消息ID。

'NoAgentNotify无服务人员通知'-如果设置为1，不会发送服务人员的邮件通知。

'AutoResponseType自动响应类型'可以使用下列的值：自动跟进、自动拒绝、自动移除、自动回复、自动回复/新工单。

'ForceNotificationToUserID强制发送通知给UserID'、'ExcludeNotificationToUserID为UserID排除通知'、'ExcludeMuteNotificationToUserID为UserID排除静音通知'可以使用的UserID列表，给这些UserID的用户要么始终发送通知、不发送通知，要么加入发送清单但并不发送通知邮件。

### 2.4.5.6.3. 工单创建

创建一个有信件的工单，这个新工单能够链接到流程工单。例如：

```
$Self->{'Process::TransitionAction'} = {
  'TA1' => {
    Name => '创建工单',
    Module => 'Kernel::System::ProcessManagement::TransitionAction::TicketCreate',
    Config => {

      # 工单必填项：
      Title => '工单标题',
      Queue => 'Raw队列', # 或队列ID QueueID => 123,
      Lock => '未锁定',
      Priority => '3-正常', # 或优先级ID PriorityID => 2,
      State => '新建', # 或状态ID StateID => 5,
      CustomerID => '123465',
      CustomerUser => 'customer@example.com',
      OwnerID => 'someuserlogin', # 或所有者ID OwnerID => 123,

      # 工单可选项：
      TN => $TicketObject->TicketCreateNumber(), # 可选项
      Type => '故障', # 或类型ID TypeID => 1, 不是必填项
      Service => '服务A', # 或服务ID ServiceID => 1, 不是必填项
      SLA => 'SLA A', # 或 SLAID => 1, 不是必填项
      ResponsibleID => 123, # 不是必填项
      ArchiveFlag => 'y', # (y|n) 不是必填项
      PendingTime => '2011-12-23 23:05:00', # 可选项 (用于挂起状态)
      PendingTimeDiff => 123, # 可选项 (用于挂起状态)

      # 信件必填项:
      ArticleType => 'note-internal (备注-内部)', # note-
      external (备注-外部) |phone (电话) |fax (传真) |sms (短信) |...
```

```

# 不包括任何电子邮件
类型
    SenderType => 'agent', # agent (服务人员) |
system (系统) |customer (客户)
    ContentType => 'text/plain; charset=ISO-8859-15', # 或可选的字符集和MIME
类型 (如'text/html; charset=UTF-8')
    Subject => '一些简短描述', # 必填项
    Body => '消息正文', # 必填项
    HistoryType => 'OwnerUpdate (更新所有者)', #
    EmailCustomer (客户邮件) |Move (移动) |AddNote (添加备注) |PriorityUpdate (更新优先级) |
WebRequestCustomer (客户网页请求) |...
    HistoryComment => 'Some free text!',

#信件可选项:
    From => 'Some Agent <email@example.com>', # 不是必填项但有用
    To => 'Some Customer A <customer-a@example.com>', # 不是必填项但有用
    Cc => 'Some Customer B <customer-b@example.com>', # 不是必填项但有用
    ReplyTo => 'Some Customer B <customer-b@example.com>', # 不是必填项
    MessageID => '<asdasdasd.123@example.com>', # 不是必填项但有用
    InReplyTo => '<asdasdasd.12@example.com>', # 不是必填项但有用
    References => '<asdasdasd.1@example.com> <asdasdasd.12@example.com>', # 不
是必填项但有用
    NoAgentNotify => 0, # 如果不想发送服务人员
通知
    AutoResponseType => 'auto reply (自动回复)' # auto reject (自动拒绝) |
auto follow up (自动跟进) |auto reply/new ticket (自动回复/新工单) |auto remove (自动移除)

    ForceNotificationToUserID => [ 1, 43, 56 ], # 如果想要强制给一些人
员发送通知
    ExcludeNotificationToUserID => [ 43,56 ], # 如果想要在发送通知时
完全排除一些人员,
# 清单中的人员也会从信
件的收件人中移除,
# 比
ForceNotificationToUserID优先级高
    ExcludeMuteNotificationToUserID => [ 43,56 ], # 与
ExcludeNotificationToUserID相同,但是只有发送通知时移除,服务人员仍然会显示在信件的收件人中。
    TimeUnit => 123

# 其它:
    DynamicField_NameX => $Value,
    LinkAs => $LinkType, # 普通、父、子等等 (相
对于原始工单)
    UserID => 123, # 可选,用于覆盖登录用
户的UserID
    },
  },
};

```

'Name名称'指定这个转换操作的名称。可以随意选择，但名称最好能体现转换操作的用途。

'Title标题'工单的标题。

'Queue队列'或'QueueID'指定用于新工单的队列名称或ID。

'Lock锁定'或'LockID'设置工单的锁定状态。

'Priority优先级'或'PriorityID'在新工单中指定优先级的名称或ID。

'State状态'或'StateID'在新工单中指定状态的名称或ID。

'CustomerID客户ID'，为新工单设置客户ID。

'CustomerUser客户联系人'，分配给工单的客户登录名。

'Owner (所有者)'或'OwnerID (所有者ID)'，指定新工单所有者的服务人员的登录名或ID。

'TN'，新工单的客户编号。

'Type类型'或'TypeID'，指定用于新工单的工单类型名称或ID。

'Service服务'或'ServiceID'，指定用于新工单的服务名称或ID。

'SLA'或'SLAID'，指定用于新工单的SLA名称或ID。

'ResponsibleID负责人ID'，新工单负责人的服务人员ID。

'PendingTime挂起时间'，预先定义的当工单状态属于一种挂起状态类型时工单的挂起时间。

'PendingTimeDiff挂起时间差'，当工单状态属于一种挂起状态类型时，给工单挂起时间设置一个动态时间（从当前时间按秒表示）。

'ArticleType信件类型'定义创建的信件的类型。可能的值有：电话、传真、短信、WEB请求、备注-内部、备注-外部和备注-报告。

'SenderType发件人类型'定义信件的发件人类型，可能的值有：服务人员、系统、客户。

'ContentType内容类型'定义信件的内容类型。可能的值有：'text/plain; charset=ISO-8859-15纯文本'或其它有效的字符集和MIME类型。

'Subject主题'定义信件主题，必填。

'Body正文'定义信件的内容，必填。

'HistoryType历史类型'定义历史条目的类型。可能的值有：添加备注、归档标志更新、退回、客户更新、服务人员邮件、客户邮件、升级响应时间通知之前、升级响应时间通知开始、升级响应时间结束、升级解决时间通知之前、升级解决时间通知开始、升级解决时间结束、升级更新时间之前、升级更新时间开始、升级更新时间结束、跟进、转发、锁定、循环保护、合并、杂项、转换、新工单、所有者更新、服务人员电话、客户电话、优先级更新、移除、负责人更新、发送服务人员通知、发送答复、发送自动跟进、发送自动拒绝、发送自动回复、发送客户通知、服务更新、设置挂起时间、SLA更新、状态更新、关注、系统请求、工单动态字段更新、添加工单链接、删除工单链接、工时管理、类型更新、解锁、取消关注、客户WEB请求。

'HistoryComment历史注释'定义历史条目的内容。

'From发件人'、'To收件人'、'Cc抄送'和'ReplyTo回复'使用上面指定格式的邮件地址。

'InReplyTo作为答复'和'References参考'使用邮件消息ID。

'NoAgentNotify无服务人员通知'-如果设置为1，不会发送服务人员的邮件通知。

'AutoResponseType自动响应类型'可以使用下列的值：自动跟进、自动拒绝、自动移除、自动回复、自动回复/新工单。

'ForceNotificationToUserID强制发送通知给UserID'、'ExcludeNotificationToUserID为UserID排除通知'、'ExcludeMuteNotificationToUserID为UserID排除静音通知'可以使用的UserID列表，给这些UserID的用户要么始终发送通知、不发送通知，要么加入发送清单但并不发送通知邮件。

'时间单位'花费在当前工单信件上的时间，按秒、分钟或小时等表示。

'DynamicField\_NameX'这里DynamicField\_是必需的前缀，NameX是在新工单（在工单级而不是信件级）设置的动态字段的名称。

'LinkAs链接为'用来定义新工单与原始工单的关系，从新工单角度看，有普通（链接）、父（链接）、子（链接）等等。

#### 2.4.5.6.4. TicketCustomerSet工单客户设置

设置一个流程工单的客户。例如：

\_\_\_\_\_

```
$Self->{'Process::TransitionAction'} = {  
  'TA1' => {  
    Name => 'Customer Set Customer to test',  
    Module => 'Kernel::System::Process::TransitionAction::TicketCustomerSet',  
    Config => {  
      No => 'test',  
      User => 'client-user-123',  
      # or in other words  
      # CustomerID => 'client123',  
      # CustomerUserID => 'client-user-123',  
    },  
  },  
};
```

‘Name名称’指定配置的转换操作的名称。

‘No编号’或‘CustomerID客户ID’设置客户ID。

‘User用户’或‘CustomerUserID客户联系人ID’设置客户的（登录）用户名。

#### 2.4.5.6.5. TicketLockSet工单锁定设置

变更一个流程工单的锁定状态。例如：

```
$Self->{'Process::TransitionAction'} = {  
  'TA1' => {  
    Name => 'Set Lock to lock',  
    Module => 'Kernel::System::ProcessManagement::TransitionAction::TicketLockSet',  
    Config => {  
      Lock => 'lock',  
      # or  
      LockID => 2,  
    },  
  },  
};
```

‘Name名称’指定配置的转换操作的名称。

‘Lock锁定’定义流程新的锁定状态。

‘LockID’定义新锁定状态的内部ID号。

#### 2.4.5.6.6. TicketOwnerSet工单所有者设置

变更一个流程工单的所有者。例如：

```
$Self->{'Process::TransitionAction'} = {  
  'TA1' => {  
    Name => 'Owner Set root@localhost',  
    Module => 'Kernel::System::ProcessManagement::TransitionAction::TicketOwnerSet',  
    Config => {  
      Owner => 'root@localhost',  
      # or  
      OwnerID => 1,  
    },  
  },  
};
```

‘Name名称’指定配置的转换操作的名称。

‘Owner所有者’指定新所有者的登录名。

‘OwnerID’指定新所有者的内部ID号。

### 2.4.5.6.7. TicketQueueSet工单队列设置

移动工单到目标队列。例如：

```
$Self->{'Process::TransitionAction'} = {  
  'TA1' => {  
    Name => 'Queue Move Raw',  
    Module => 'Kernel::System::ProcessManagement::TransitionAction::TicketQueueSet',  
    Config => {  
      Queue => 'Raw',  
      # or  
      # QueueID => '2',  
    },  
  },  
};
```

‘Name名称’指定配置的转换操作的名称。

‘Queue队列’指定目标队列的名称。

‘QueueID队列ID’指定目标队列的内部ID号。

### 2.4.5.6.8. TicketResponsibleSet工单负责人设置

变更一个流程工单的负责人。例如：

```
$Self->{'Process::TransitionAction'} = {  
  'TA1' => {  
    Name => 'Responsible Set root@localhost',  
    Module =>  
    'Kernel::System::ProcessManagement::TransitionAction::TicketResponsibleSet',  
    Config => {  
      Responsible => 'root@localhost',  
      # or  
      ResponsibleID => 1,  
    },  
  },  
};
```

‘Name名称’指定配置的转换操作的名称。

‘Responsible负责人’指定新负责人的登录名。

‘ResponsibleID负责人ID’指定新负责人的内部ID。

### 2.4.5.6.9. TicketServiceSet工单服务设置

给一个流程工单分配一个服务。工单需要有一个客户，服务必须分配给那个客户。例如：

```
$Self->{'Process::TransitionAction'} = {  
  'TA1' => {  
    Name => 'Set MyService service',  
    Module => 'Kernel::System::ProcessManagement::TransitionAction::TicketServiceSet',  
    Config => {  
      Service => 'MyService',  
      # or  
      ServiceID => 123,  
    },  
  },  
};
```

‘Name名称’指定配置的转换操作的名称。

‘Service服务’定义流程工单的新服务。需要服务的全名（如 GrandFatherService::FatherService::SonService）。

‘ServiceID服务ID’定义新服务的内部ID号。

#### 2.4.5.6.10. TicketSLASet工单SLA设置

给一个流程工单分配一个服务级别协议。工单需要有一个服务，SLA（服务级别协议）必须分配给那个服务。例如：

```
$Self->{'Process::TransitionAction'} = {
  'TA1' => {
    Name => 'Set MySLA SLA',
    Module => 'Kernel::System::ProcessManagement::TransitionAction::TicketSLASet',
    Config => {
      SLA => 'MySLA',
      # or
      SLAID => 123,
    },
  },
};
```

‘Name名称’指定配置的转换操作的名称。

‘SLA’定义流程工单新的服务级别协议。

‘SLAID’定义新SLA的内部ID号。

#### 2.4.5.6.11. TicketStateSet工单状态设置

变更一个流程工单的状态。例如：

```
$Self->{'Process::TransitionAction'} = {
  'TA1' => {
    Name => 'Set State to open',
    Module => 'Kernel::System::ProcessManagement::TransitionAction::TicketStateSet',
    Config => {
      State => 'open',
      # or
      StateID => 4,

      PendingTimeDiff => 123,
    },
  },
};
```

‘Name名称’指定配置的转换操作的名称。

‘State状态’定义流程工单新的状态。

‘StateID’定义新状态的内部ID号。

‘PendingTimeDiff挂起时间差’仅用于挂起类型的状态，定义相对时间差秒数（相对于转换操作的执行时间）以便设置工单挂起时间，例如3600意味着挂起时间是在转换操作执行后的1小时。

#### 2.4.5.6.12. TicketTitleSet工单标题设置

设置一个流程工单的标题。例如：

```
$Self->{'Process::TransitionAction'} = {
  'TA1' => {
```

```

Name => 'Set Ticket Title to Ticket-title',
Module => 'Kernel::System::ProcessManagement::TransitionAction::TicketTitleSet',
Config => {
  Title => 'Ticket-title',
},
},
};

```

‘Name名称’指定配置的转换操作的名称。

‘Title标题’指定工单的新标题。

#### 2.4.5.6.13. TicketTypeSet工单类型设置

设置一个流程工单的工单类型。例如：

```

$self->{'Process::TransitionAction'} = {
  'TA1' => {
    Name => 'Set Ticket Type to default',
    Module => 'Kernel::System::ProcessManagement::TransitionAction::TicketTypeSet',
    Config => {
      Type => 'default',
      # or
      # TypeID => '1',
    },
  },
};

```

‘Name名称’指定配置的转换操作的名称。

‘Type类型’指定工单类型的名称。

‘TypeID’指定工单类型的内部ID。

### 2.4.6. 访问控制列表（ACL）

在ACL的帮助下，你可以在流程工单中限制可选择的值。另请参阅ACL参考以获取全部工单ACL语法的描述。

#### 2.4.6.1. ACL 配置

只可以在文件Kernel/Config.pm中定义ACL。例如：

```

$self->{TicketAcl}->{'001-ACL-ProcessProperties'} = {
  Properties => {
    Process => {
      ProcessEntityID => ['P1'],
      ActivityEntityID => ['A1'],
      ActivityDialogEntityID => ['AD1'],
    }
  },
  Possible => {
    ActivityDialog => ['AD1', 'AD3'],
  },
  PossibleNot => {
    ActivityDialog => ['AD3'],
  },
};

```

#### 2.4.6.2. 001-ACL-ProcessProperties

ACL规则的名称。ACL规则的更多的信息，请查阅ACL手册。



### 2.4.6.3. 流程

这部分用于检查是否必须应用一个ACL。如果指定了值，就是应用了ACL规则，可以使用下列值：

#### 2.4.6.3.1. ProcessEntityID 流程实体ID

一个流程的ID，如果工单分配到这个流程，则匹配成功。

#### 2.4.6.3.2. ActivityEntityID 活动实体ID

流程工单目前分配的活动的ID。

#### 2.4.6.3.3. ActivityDialogEntityID 活动对话框实体ID

一个流程工单目前打开的活动对话框的ID。

### 2.4.6.4. 可能的/不可能的 活动对话框

在这里你可以指定活动对话框ID清单。这个清单限制了在工单详情窗口提供给用户的可用活动对话框。

'Possible 可能的'列出允许的活动对话框。上面示例的设置配置的活动对话框清单中仅允许'AD1'和'AD3'。

'PossibleNot 不可能的'列出不允许的活动对话框。在上面的示例中，这个设置从配置的活动对话框清单中移除'AD3'。

如果同时指定了'Possible可能的'和'PossibleNot 不可能的'，配置的活动对话框将首先用'Possible可能的'过滤，在我们的示例中仅留下'AD1'和'AD3'。然后应用'PossibleNot 不可能的'，过滤掉'AD3'，所以最终只有'AD1'保留下来，显示为用户能够使用的可用活动对话框。

如果匹配了多个ACL规则，则计算所有匹配规则的交集来确定可能的活动对话框。例如：

配置的活动对话框：'AD1'、'AD2'、'AD3'、'AD4'、'AD5'、'AD6'、'AD7'。

```

$self->{TicketAcl}->{'001-ACL-Status'} = {
  Properties => {
    Ticket => {
      Status => 'new',
    }
  },
  Possible => {
    ActivityDialog => ['AD1', 'AD2', 'AD3', 'AD6', 'AD7'],
  },
};
$self->{TicketAcl}->{'002-ACL-Queue'} = {
  Properties => {
    Ticket => {
      Queue => ['Raw']
    }
  },
  Possible => {
    ActivityDialog => ['AD2', 'AD3', 'AD4', 'AD7'],
  },
};
$self->{TicketAcl}->{'003-ACL-Priority'} = {
  Properties => {
    Ticket => {
      Priority => ['3 normal']
    }
  },
  PossibleNot => {
    ActivityDialog => ['AD3', 'AD4'],
  },
};

```

如果一个流程工单的状态为‘new新建’、在‘Raw’队列且优先级为‘3 - 正常’，则会匹配所有的3条ACL规则。

第一条规则减少活动对话框，从‘AD1’、‘AD2’、‘AD3’、‘AD4’、‘AD5’、‘AD6’、‘AD7’变成了‘AD1’、‘AD2’、‘AD3’、‘AD6’、‘AD7’，禁用了‘AD4’和‘AD5’。

第二条规则会进一步减少剩余的活动对话框。在我们的示例中，仅存‘AD2’、‘AD3’、‘AD7’。

现在第三条规则通过‘PossibleNot 不可能的’会进一步减少这个清单。从清单中移除‘AD3’。没有移除‘AD4’，因为它并没有在先前的清单中。最终，‘AD2’和‘AD7’保留为用户能够使用的可用活动对话框。

还可以限制能够显示在“新流程工单”窗口的流程。这个功能与限制活动对话框类似，除了一点：ACL只能基于用户。

查看下面的示例：

```
$Self->{TicketAcl}->{'200-ACL-Process'} = {
  # match properties
  Properties => {
    User => {
      UserID => [2, 3],
    },
  },
  Possible => {
    Process => ['P1', 'P2', 'P3'],
  },
  PossibleNot => {
    Process => ['P4'],
  },
};
```

```
$Self->{TicketAcl}->{'201-ACL-Process'} = {
  # match properties
  Properties => {
    User => {
      Group_rw => [ 'MyGroup' ],
    },
  },
  Possible => {
    Process => ['P1', 'P2', 'P3'],
  },
  PossibleNot => {
    Process => ['P4'],
  },
};
```

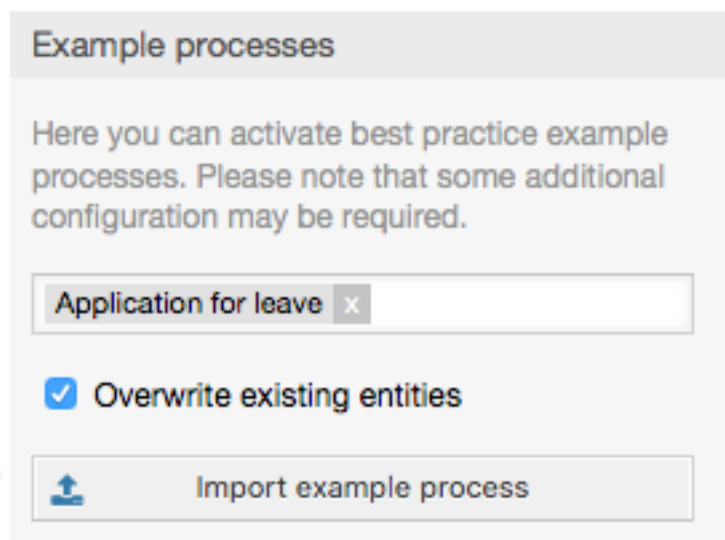
```
$Self->{TicketAcl}->{'202-ACL-Process'} = {
  # match properties
  Properties => {
    User => {
      Role => [ 'MyRole' ],
    },
  },
  Possible => {
    Process => ['P1', 'P2', 'P3'],
  },
  PossibleNot => {
    Process => ['P4'],
  },
};
```

## 2.5. 导入样例流程

### 2.5.1. 导入

On the 流程管理 窗口可以找到 流程样例小部件，这里可以找到流程样例的最佳实践。当前只有 请假申请流程可用，但可以在 **OTRS Business Solution™** 版本中找到更多的样例流程。

图 5.24. 导入样例流程小部件



在下拉菜单中选择流程，并点击导入示例流程按钮。流程导入后，不要忘记部署变更。

## 3. 创建你自己的主题

你可以创建你自己的主题，从而按你喜欢的布局使用ORTS的WEB界面。要创建自己的主题，你应该按你的需要定制输出模块。

输出模板的语法和结构方面的更多信息请参阅开发人员手册<http://otrs.github.io/doc>，特别是 [templates模板](#) 这一章。

作为一个示例，执行下列的步骤来创建一个名为“Company”的新主题：

1. 创建一个名为Kernel/Output/HTML/Templates/Company的目录，并从目录Kernel/Output/HTML/Templates/Standard中复制你要修改的所有文件到这个新目录。

### 重要

只需复制你确实要修改的文件即可。OTRS会自动从Standard主题中获取缺失的文件。这也使得后续的版本升级更容易。

2. 修改目录Kernel/Output/HTML/Templates/Company中的文件，定制你需要的布局。
3. 要激活这个新主题，先在系统配置Frontend::Themes中添加。

现在这个新主题就能使用了。你可以从 个人偏好设置页面选择新主题。

### 警告

不要直接修改OTRS自带的主题文件，因为这些修改在OTRS升级后会丢失。只需按上述步骤创建你自己的主题就行了。

## 4. OTRS界面的本地化

OTRS框架本地化的程序，创建一个新语言翻译的步骤，与翻译定制的程序一样，都可以在开发人员手册 <http://otrs.github.io/doc>的“语言翻译”章节找到。



## 第 6 章 性能优化

下面是OTRS安装（包括配置）、编码、内存使用及其它方面的性能增强技术的清单。

### 1. OTRS

提升OTRS性能有如下一些方法。

#### 1.1. TicketIndexModule 工单索引模块

工单队列视图的索引有两个后端模块：

`Kernel::System::Ticket::IndexAccelerator::RuntimeDB`

This is the default option, and will generate each queue view on the fly from the ticket table. You will not have performance trouble until you have about 60,000 open tickets in your system.

`Kernel::System::Ticket::IndexAccelerator::StaticDB`

The most powerful module, should be used when you have above 80,000 open tickets. It uses an extra `ticket_index` table, which will be populated with keywords based on ticket data. Use `bin/otrs.Console.pl Maint::Ticket::QueueIndexRebuild` for generating an initial index after switching backends.

You can change the used `IndexAccelerator` module via `SysConfig`.

#### 1.2. SearchIndexModule

This module helps to extend your articles full-text search (From, To, Cc, Subject and Body search). There are two backend modules for the search index:

`Kernel::System::Ticket::ArticleSearchIndex::RuntimeDB`

This is the default option, and will do full-text searches on live data (it works fine for up to 50,000 tickets).

`Kernel::System::Ticket::ArticleSearchIndex::StaticDB`

This module will strip all articles and build an index after article creation, increasing performance of full-text searches up to 50%.

You can change the used `SearchIndexModule` via `SysConfig`.

To create an initial index, use `bin/otrs.Console.pl Maint::Ticket::FulltextIndexRebuild`.

For `StaticDB` module, there are some options available for fine tuning:

`Ticket::SearchIndex::Attribute`

The attribute `"WordCountMax"` defines the maximum of words which will be processed to build up the index. For example only the first 1000 words of an article body are stored in the article search index. The attributes `"WordLengthMin"` and `"WordLengthMax"` are used as word length boundaries. Only words with a length between these two values are stored in the article search index.

## Ticket::SearchIndex::Filters

There are three default filters defined:

- The first filter strips out special chars like: , & < > ? " ! \* | ; [ ] ( ) + \$ ^=
- The second filter strips out words which begin or ends with one of following chars: ' : .
- The third filter strips out words which do not contain a word-character: a-z, A-Z, 0-9, \_

## Ticket::SearchIndex::StopWords

There are so-called stop-words defined for some languages. These stop-words will be skipped while creating the search index.

## 1.3. TicketStorageModule 工单存储模块

存储工单/信件有两个不同后端模块：

### Kernel::System::Ticket::ArticleStorageDB

This default module will store attachments in the database.

#### 注意

Don't use it with large setups.

Pro: If your web server isn't running under 'otrs' user, use this module to avoid file permission problems.

Con: It is not advisable to store attachments in your database. Take care that your database is able to store large objects. I.e.: configure MySQL with `set-variable = max_allowed_packet=8M` to store 8 MB objects (the default is 2M).

### Kernel::System::Ticket::ArticleStorageFS

Use this module to store attachments on the local file system.

#### 注意

Recommended for large setups.

赞成：它很快！

Con: Your web server should run under the 'otrs' user. Also, if you have multiple front-end servers, you must make sure the filesystem is shared between the servers. Place it on an NFS share or preferably a SAN or similar solution.

#### 注意

You can switch from one back-end to the other on the fly. You can switch the backend in the SysConfig, and then run the command line utility `bin/otrs.Console.pl Admin::Article::StorageSwitch` to put the articles from the database onto the filesystem or the other way around. You can use the `--target` option to specify the target backend. Please note that the entire process can take considerable time to run, depending on the number of articles you have and the available CPU power and/or network capacity.

```
shell> bin/otrs.Console.pl Admin::Article::StorageSwitch --target ArticleStorageFS
```

脚本: 切换存储后端, 从数据库切换到文件系统。

如果你想在数据库中保持旧的附件, 可以激活系统配置选项Ticket::StorageModule::CheckAllBackends, 以确保OTRS仍然能够找到它们。

## 1.4. Archiving Tickets

As OTRS can be used as an audit-proof system, deleting closed tickets may not be a good idea. Therefore we implemented a feature that allows you to archive tickets.

匹配某个条件的工单可以标记为“已归档”。这些工单在使用常规的工单搜索或运行一个自动任务时无法访问。系统本身不再需要处理大量的工单, 而只考虑“最近”的工单就可以了。这在大型系统中能带来巨大的性能提升。

To use the archive feature simply follow these steps:

### 1. Activate the archive system in SysConfig

在系统管理页面中, 进入系统管理后选择Ticket (工单) 组, 在Core::Ticket中找到选项Ticket::ArchiveSystem, 默认设置为“否”。修改这个选项为“是”并保存。

### 2. 定义一个自动任务

在系统管理页面, 选择'GenericAgent自动任务'并添加一个新任务。

#### a. 任务设置

为归档任务提供一个名称, 并选择合适的选项来计划这个任务。

#### b. 工单过滤

工单过滤器就是搜索匹配选定条件的工单。要只归档前几个月关闭的工单, 使用过滤器也许是一个好主意。

#### c. 工单操作

在这部分设置标签为“归档选中的工单”字段为“archive tickets归档工单”。

#### d. 保存任务

在页面的结尾可以找到保存任务的按钮。

#### e. 影响的工单

系统在执行这个自动任务时会显示所有要归档的工单。

### 3. 工单搜索

当你搜索工单时, 系统默认搜索未归档的工单。如果你要同时搜索已归档的工单, 仅需在定义搜索条件时添加“归档搜索”即可。

## 1.5. 缓存

OTRS caches a lot of temporary data in /opt/otrs/var/tmp. Please make sure that this uses a high performance file system/storage. If you have enough RAM, you can also try to put this directory on a ramdisk like this:

```
shell> /opt/otrs/bin/otrs.Console.pl Maint::Session::DeleteAll
shell> /opt/otrs/bin/otrs.Console.pl Maint::Cache::Delete
shell> sudo mount -o size=16G -t tmpfs none /opt/otrs/var/tmp
```

# 可在文件/etc/fstab中添加永久挂载点

## 注意

Please note that this will be a non-permanent storage that will be lost on server reboot. All your sessions (if you store them in the filesystem) and your cache data will be lost.

There is also a centralized memcached based cache backend available for purchase from OTRS Group.

## 2. 数据库

你使用的数据库不同，数据库性能问题也不一样。学习你采用的数据库的相关文档，或与你的数据库管理员一起检查。

### 2.1. MySQL

如果你使用默认的MySQL表类型MyISAM，并且删除了大型表的一部分或对有变长行的表（有VARCHAR、BLOB、TEXT列的表）做了大量修改，你必须用“optimize”命令对表的数据文件做碎片整理。

如果mysqld进程占用了大量的CPU时间，你就应该尝试执行optimize，来优化ticket（工单）、ticket\_history（工单历史）和article（信件）表（参阅下面的脚本）。

```
shell> mysql -u user -p database
mysql> optimize table ticket;
mysql> optimize table ticket_history;
mysql> optimize table article;
```

脚本：优化数据库表。

### 2.2. PostgreSQL

PostgreSQL最好通过修改数据目录下的postgresql.conf文件来调优。关于如何调优的建议，请参考下列文章：

- <http://www.revsys.com/writings/postgresql-performance.html>
- <http://varlena.com/GeneralBits/Tidbits/perf.html>
- [http://varlena.com/GeneralBits/Tidbits/annotated\\_conf\\_e.html](http://varlena.com/GeneralBits/Tidbits/annotated_conf_e.html)

If performance is still not satisfactory, we suggest that you join the PostgreSQL Performance mailing list ( <http://www.postgresql.org/community/lists/> ), and ask questions there. The folks on the PostgreSQL list are very friendly and can probably help.

## 3. WEB服务器

Of course you should use mod\_perl 2.0 ( <http://perl.apache.org/> ). It's much faster (~\*100) than pure CGI, but it needs more RAM.



## 3.1. 预建立的数据库连接

You can have the database connections pre-established on startup of the web server. This saves time (see `README.webserver`).

## 3.2. 预装载的模块 - `startup.pl`

Use the startup script `scripts/apache2-perl-startup.pl` for preloaded/precompiled Perl modules on your `mod_perl` webserver to be faster, with a smaller memory footprint (see `README.webserver`).

## 3.3. 当磁盘文件更新时重载Perl模块

By default `Apache::Reload` is used in `scripts/apache2-httpd.include.conf`. Disable it and you will get 8% more speed. But remember to restart the web server if you install any modules via the OTRS Package Manager, or any values in your `SysConfig` or in `Kernel/Config.pm`.

### 重要

This would also mean you can't use the OTRS Package Manager via the web interface, you will need to use the console commands: `bin/otrs.Console.pl Admin::Package::*`.

## 3.4. 选择正确的策略

If you have a larger installation, e.g. over 1,000 new tickets per day and over 40 agents, it is a good idea to read the chapters on Performance of the `mod_perl` User's Guide ( <http://perl.apache.org/docs/2.0/user/index.html> ).

## 3.5. `mod_gzip/mod_deflate`

If your bandwidth is small, use `mod_deflate` for Apache2. If you have an html page with 45k, `mod_gzip/mod_deflate` compresses it to about 7k. The drawback is that this increases the load on the server side.

# 附录 A. 其他资源

## otrs.com

OTRS网站地址为：[www.otrs.com](http://www.otrs.com), 网站上有源代码、文档和OTRS新闻等。你还可以在这里找到来自OTRS创建者-OTRS集团 关于专业服务和OTRS系统管理员培训研讨会的信息。

### 邮件列表

表 A.1. 邮件列表

名称 & 地址	描述
<a href="mailto:announce@otrs.org">announce@otrs.org</a>	低流量列表，英语，用来公告OTRS新版本和安全问题。
<a href="mailto:otrs@otrs.org">otrs@otrs.org</a>	中等流量列表，英语，你可以在这里找到所有排序了的相关问题和产品支持。
<a href="mailto:otrs-de@otrs.org">otrs-de@otrs.org</a>	中等流量列表，德语，你可以在这里找到所有排序了的相关问题和产品支持。
<a href="mailto:dev@otrs.org">dev@otrs.org</a>	中等流量列表，英语，开发人员在这里各种设计和实现问题。

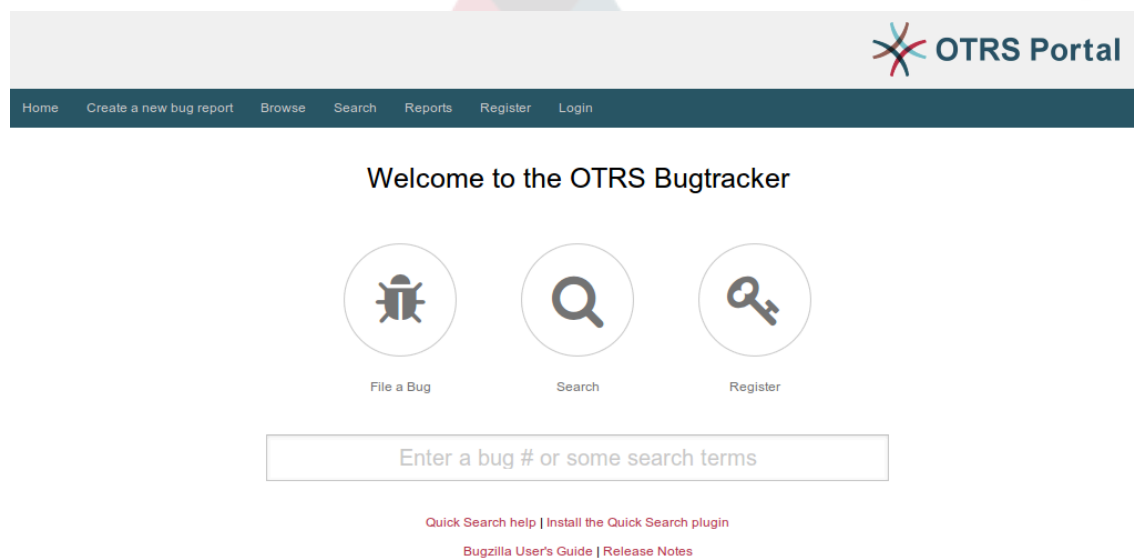
### 翻译

你可以通过[Transifex](#)帮助翻译OTRS到你的语言。

### Bug 跟踪

请访问<http://bugs.otrs.org>来报告软件缺陷（如下图）。请注意一个BUG和一个配置问题之间的不同。配置问题是你设置一个系统时遇到的问题，或者有关使用OTRS的一般问题。BUG报告仅用于OTRS或其它开源的OTRS模块自身的源代码问题。对配置问题你可以使用[OTRS集团可用的商业支持](#)，或公共邮件列表。

图 A.1. Bug 跟踪工具



# 附录 B. 配置选项参考手册

## 1. CloudService

### CloudService → CloudService::Admin::ModuleRegistration

#### CloudService::Admin::Module###100-SupportDataCollector

Cloud service admin module registration for the transport layer.

默认值:

```
$Self->{'CloudService::Admin::Module'}->{'100-SupportDataCollector'} = {
  'ConfigDialog' => 'AdminCloudServiceSupportDataCollector',
  'Description' => 'Configure sending of support data to OTRS Group for improved
support.',
  'Icon' => 'fa fa-compass',
  'Name' => 'Support data collector'
};
```

#### CloudService::Admin::Module###200-SMS

Cloud service admin module registration for the transport layer.

默认值:

```
$Self->{'CloudService::Admin::Module'}->{'200-SMS'} = {
  'ConfigDialog' => 'AdminCloudServiceSMS',
  'Description' => 'This will allow the system to send text messages via SMS.',
  'Icon' => 'fa fa-mobile',
  'IsOTRSBusiness' => '1',
  'Name' => 'SMS'
};
```

### CloudService → Core

#### CloudServices::Disabled

Disables the communication between this system and OTRS Group servers that provides cloud services. If active, some functionality will be lost such as system registration, support data sending, upgrading to and use of OTRS Business Solution™, OTRS Verify™, OTRS News and product News dashboard widgets, among others.

这个设置默认没有激活。

默认值:

```
$Self->{'CloudServices::Disabled'} = '0';
```

### CloudService → Frontend::Agent::ModuleNotify

#### Frontend::NotifyModule###100-CloudServicesDisabled

Defines the module to display a notification if cloud services are disabled.

不能使这个设置失效。

默认值:

```
$Self->{'Frontend::NotifyModule'}->{'100-CloudServicesDisabled'} = {
  'Group' => 'admin',
```

```
'Module' => 'Kernel::Output::HTML::Notification::AgentCloudServicesDisabled'  
};
```

## 2. Daemon

### Daemon → Core::Daemon::ModuleRegistration

#### DaemonModules###SchedulerGenericAgentTaskManager

The daemon registration for the scheduler generic agent task manager.

不能使这个设置失效。

默认值:

```
$Self->{'DaemonModules'}->{'SchedulerGenericAgentTaskManager'} = {  
  'Module' => 'Kernel::System::Daemon::DaemonModules::SchedulerGenericAgentTaskManager'  
};
```

#### DaemonModules###SchedulerCronTaskManager

The daemon registration for the scheduler cron task manager.

不能使这个设置失效。

默认值:

```
$Self->{'DaemonModules'}->{'SchedulerCronTaskManager'} = {  
  'Module' => 'Kernel::System::Daemon::DaemonModules::SchedulerCronTaskManager'  
};
```

#### DaemonModules###SchedulerFutureTaskManager

The daemon registration for the scheduler future task manager.

不能使这个设置失效。

默认值:

```
$Self->{'DaemonModules'}->{'SchedulerFutureTaskManager'} = {  
  'Module' => 'Kernel::System::Daemon::DaemonModules::SchedulerFutureTaskManager'  
};
```

#### DaemonModules###SchedulerTaskWorker

The daemon registration for the scheduler task worker.

不能使这个设置失效。

默认值:

```
$Self->{'DaemonModules'}->{'SchedulerTaskWorker'} = {  
  'Module' => 'Kernel::System::Daemon::DaemonModules::SchedulerTaskWorker'  
};
```

### Daemon → Core::Log

#### Daemon::Log::DaysToKeep

Defines the number of days to keep the daemon log files.

这个设置默认没有激活。

默认值:

```
$Self->{'Daemon::Log::DaysToKeep'} = '1';
```

### **Daemon::Log::STDOUT**

If enabled the daemon will redirect the standard output stream to a log file.

不能使这个设置失效。

默认值:

```
$Self->{'Daemon::Log::STDOUT'} = '0';
```

### **Daemon::Log::STDERR**

If enabled the daemon will redirect the standard error stream to a log file.

不能使这个设置失效。

默认值:

```
$Self->{'Daemon::Log::STDERR'} = '1';
```

## **Daemon → Core::Web**

### **Loader::Agent::CommonCSS###001-Daemon**

List of CSS files to always be loaded for the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Loader::Agent::CommonCSS'}->{'001-Daemon'} = [  
  'Core.Agent.DaemonInfo.css'  
];
```

### **Loader::Agent::CommonJS###001-Daemon**

List of JS files to always be loaded for the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Loader::Agent::CommonJS'}->{'001-Daemon'} = [  
  'Core.Agent.DaemonInfo.js'  
];
```

## **Daemon → Daemon**

### **Daemon::PID::Path**

If enabled the daemon will use this directory to create its PID files. Note: Please stop the daemon before any change and use this setting only if <\$OTRSHome>/var/run/ can not be used.

这个设置默认没有激活。

默认值:

```
$Self->{'Daemon::PID::Path'} = '/opt/otrs/var/run/';
```

## Daemon → Daemon::SchedulerCronTaskManager::Task

### Daemon::SchedulerCronTaskManager::Task###CoreCacheCleanup

Delete expired cache from core modules.

默认值:

```
$Self->{'Daemon::SchedulerCronTaskManager::Task'}->{'CoreCacheCleanup'} = {  
  'Function' => 'CleanUp',  
  'MaximumParallelInstances' => '1',  
  'Module' => 'Kernel::System::Cache',  
  'Params' => [  
    'Expired',  
    '1'  
  ],  
  'Schedule' => '20 0 * * 0',  
  'TaskName' => 'CoreCacheCleanup'  
};
```

### Daemon::SchedulerCronTaskManager::Task###LoaderCacheDelete

Delete expired loader cache weekly (Sunday mornings).

默认值:

```
$Self->{'Daemon::SchedulerCronTaskManager::Task'}->{'LoaderCacheDelete'} = {  
  'Function' => 'CacheDelete',  
  'MaximumParallelInstances' => '1',  
  'Module' => 'Kernel::System::Loader',  
  'Params' => [],  
  'Schedule' => '30 0 * * 0',  
  'TaskName' => 'LoaderCacheDelete'  
};
```

### Daemon::SchedulerCronTaskManager::Task###FetchMail

Fetch emails via fetchmail.

这个设置默认没有激活。

默认值:

```
$Self->{'Daemon::SchedulerCronTaskManager::Task'}->{'FetchMail'} = {  
  'Function' => 'Fetch',  
  'MaximumParallelInstances' => '1',  
  'Module' => 'Kernel::System::FetchMail',  
  'Params' => [],  
  'Schedule' => '* /5 * * * *',  
  'TaskName' => 'FetchMail'  
};
```

### Daemon::SchedulerCronTaskManager::Task###FetchMailSSL

Fetch emails via fetchmail (using SSL).

这个设置默认没有激活。

默认值:

```
$Self->{'Daemon::SchedulerCronTaskManager::Task'}->{'FetchMailSSL'} = {  
  'Function' => 'Fetch',  
  'MaximumParallelInstances' => '1',  
  'Module' => 'Kernel::System::FetchMail',  
  'Params' => [  
    'SSL',  
    '1'  
  ],  
  'Schedule' => '* /5 * * * *',  
};
```

```
'TaskName' => 'FetchMailSSL'
};
```

### Daemon::SchedulerCronTaskManager::Task###GenerateDashboardStats

Generate dashboard statistics.

默认值:

```
$Self->{'Daemon::SchedulerCronTaskManager::Task'}->{'GenerateDashboardStats'} = {
  'Function' => 'Execute',
  'MaximumParallelInstances' => '1',
  'Module' => 'Kernel::System::Console::Command::Maint::Stats::Dashboard::Generate',
  'Params' => [],
  'Schedule' => '5 * * * *',
  'TaskName' => 'GenerateDashboardStats'
};
```

### Daemon::SchedulerCronTaskManager::Task###EscalationCheck

Triggers ticket escalation events and notification events for escalation.

默认值:

```
$Self->{'Daemon::SchedulerCronTaskManager::Task'}->{'EscalationCheck'} = {
  'Function' => 'Execute',
  'MaximumParallelInstances' => '1',
  'Module' => 'Kernel::System::Console::Command::Maint::Ticket::EscalationCheck',
  'Params' => [],
  'Schedule' => '* / 5 * * * *',
  'TaskName' => 'EscalationCheck'
};
```

### Daemon::SchedulerCronTaskManager::Task###TicketPendingCheck

Process pending tickets.

默认值:

```
$Self->{'Daemon::SchedulerCronTaskManager::Task'}->{'TicketPendingCheck'} = {
  'Function' => 'Execute',
  'MaximumParallelInstances' => '1',
  'Module' => 'Kernel::System::Console::Command::Maint::Ticket::PendingCheck',
  'Params' => [],
  'Schedule' => '45 */ 2 * * * *',
  'TaskName' => 'TicketPendingCheck'
};
```

### Daemon::SchedulerCronTaskManager::Task###SpoolMailsReprocess

Reprocess mails from spool directory that could not be imported in the first place.

默认值:

```
$Self->{'Daemon::SchedulerCronTaskManager::Task'}->{'SpoolMailsReprocess'} = {
  'Function' => 'Execute',
  'MaximumParallelInstances' => '1',
  'Module' =>
  'Kernel::System::Console::Command::Maint::PostMaster::SpoolMailsReprocess',
  'Params' => [],
  'Schedule' => '10 0 * * *',
  'TaskName' => 'SpoolMailsReprocess'
};
```

### Daemon::SchedulerCronTaskManager::Task###MailAccountFetch

Fetch incoming emails from configured mail accounts.

默认值:

```
$Self->{'Daemon::SchedulerCronTaskManager::Task'}->{'MailAccountFetch'} = {
  'Function' => 'Execute',
  'MaximumParallelInstances' => '1',
  'Module' => 'Kernel::System::Console::Command::Maint::PostMaster::MailAccountFetch',
  'Params' => [],
  'Schedule' => '*/*10 * * * *',
  'TaskName' => 'MailAccountFetch'
};
```

### Daemon::SchedulerCronTaskManager::Task###TicketAcceleratorRebuild

Rebuild the ticket index for AgentTicketQueue.

默认值:

```
$Self->{'Daemon::SchedulerCronTaskManager::Task'}->{'TicketAcceleratorRebuild'} = {
  'Function' => 'TicketAcceleratorRebuild',
  'MaximumParallelInstances' => '1',
  'Module' => 'Kernel::System::Ticket',
  'Params' => [],
  'Schedule' => '01 01 * * *',
  'TaskName' => 'TicketAcceleratorRebuild'
};
```

### Daemon::SchedulerCronTaskManager::Task###SessionDeleteExpired

Delete expired sessions.

默认值:

```
$Self->{'Daemon::SchedulerCronTaskManager::Task'}->{'SessionDeleteExpired'} = {
  'Function' => 'Execute',
  'MaximumParallelInstances' => '1',
  'Module' => 'Kernel::System::Console::Command::Maint::Session::DeleteExpired',
  'Params' => [],
  'Schedule' => '55 */2 * * *',
  'TaskName' => 'SessionDeleteExpired'
};
```

### Daemon::SchedulerCronTaskManager::Task###TicketUnlockTimeout

Unlock tickets that are past their unlock timeout.

默认值:

```
$Self->{'Daemon::SchedulerCronTaskManager::Task'}->{'TicketUnlockTimeout'} = {
  'Function' => 'Execute',
  'MaximumParallelInstances' => '1',
  'Module' => 'Kernel::System::Console::Command::Maint::Ticket::UnlockTimeout',
  'Params' => [],
  'Schedule' => '35 * * * *',
  'TaskName' => 'TicketUnlockTimeout'
};
```

### Daemon::SchedulerCronTaskManager::Task###RenewCustomerSMIMECertificates

Renew existing SMIME certificates from customer backend. Note: SMIME and SMIME::FetchFromCustomer needs to be active in SysConfig and customer backend needs to be configured to fetch UserSMIMECertificate attribute.

默认值:

```
$Self->{'Daemon::SchedulerCronTaskManager::Task'}->{'RenewCustomerSMIMECertificates'} =
{
  'Function' => 'Execute',
  'MaximumParallelInstances' => '1',
  'Module' =>
  'Kernel::System::Console::Command::Maint::SMIME::CustomerCertificate::Renew',
```



```
'Params' => [],  
'Schedule' => '02 02 * * *',  
'TaskName' => 'RenewCustomerSMIMECertificates'  
};
```

### **Daemon::SchedulerCronTaskManager::Task###Custom1**

Executes a custom command or module. Note: if module is used, function is required.

这个设置默认没有激活。

默认值:

```
$Self->{'Daemon::SchedulerCronTaskManager::Task'}->{'Custom1'} = {  
  'Function' => '',  
  'MaximumParallelInstances' => '1',  
  'Module' => '',  
  'Params' => [],  
  'Schedule' => '* * * * *',  
  'TaskName' => 'Custom1'  
};
```

### **Daemon::SchedulerCronTaskManager::Task###Custom2**

Executes a custom command or module. Note: if module is used, function is required.

这个设置默认没有激活。

默认值:

```
$Self->{'Daemon::SchedulerCronTaskManager::Task'}->{'Custom2'} = {  
  'Function' => '',  
  'MaximumParallelInstances' => '1',  
  'Module' => '',  
  'Params' => [],  
  'Schedule' => '* * * * *',  
  'TaskName' => 'Custom2'  
};
```

### **Daemon::SchedulerCronTaskManager::Task###Custom3**

Executes a custom command or module. Note: if module is used, function is required.

这个设置默认没有激活。

默认值:

```
$Self->{'Daemon::SchedulerCronTaskManager::Task'}->{'Custom3'} = {  
  'Function' => '',  
  'MaximumParallelInstances' => '1',  
  'Module' => '',  
  'Params' => [],  
  'Schedule' => '* * * * *',  
  'TaskName' => 'Custom3'  
};
```

### **Daemon::SchedulerCronTaskManager::Task###Custom4**

Executes a custom command or module. Note: if module is used, function is required.

这个设置默认没有激活。

默认值:

```
$Self->{'Daemon::SchedulerCronTaskManager::Task'}->{'Custom4'} = {  
  'Function' => '',  
  'MaximumParallelInstances' => '1',  
  'Module' => '',
```

```
'Params' => [],  
'Schedule' => '* * * * *',  
'TaskName' => 'Custom4'  
};
```

### **Daemon::SchedulerCronTaskManager::Task###Custom5**

Executes a custom command or module. Note: if module is used, function is required.

这个设置默认没有激活。

默认值:

```
$Self->{'Daemon::SchedulerCronTaskManager::Task'}->{'Custom5'} = {  
  'Function' => '',  
  'MaximumParallelInstances' => '1',  
  'Module' => '',  
  'Params' => [],  
  'Schedule' => '* * * * *',  
  'TaskName' => 'Custom5'  
};
```

### **Daemon::SchedulerCronTaskManager::Task###Custom6**

Executes a custom command or module. Note: if module is used, function is required.

这个设置默认没有激活。

默认值:

```
$Self->{'Daemon::SchedulerCronTaskManager::Task'}->{'Custom6'} = {  
  'Function' => '',  
  'MaximumParallelInstances' => '1',  
  'Module' => '',  
  'Params' => [],  
  'Schedule' => '* * * * *',  
  'TaskName' => 'Custom6'  
};
```

### **Daemon::SchedulerCronTaskManager::Task###Custom7**

Executes a custom command or module. Note: if module is used, function is required.

这个设置默认没有激活。

默认值:

```
$Self->{'Daemon::SchedulerCronTaskManager::Task'}->{'Custom7'} = {  
  'Function' => '',  
  'MaximumParallelInstances' => '1',  
  'Module' => '',  
  'Params' => [],  
  'Schedule' => '* * * * *',  
  'TaskName' => 'Custom7'  
};
```

### **Daemon::SchedulerCronTaskManager::Task###Custom8**

Executes a custom command or module. Note: if module is used, function is required.

这个设置默认没有激活。

默认值:

```
$Self->{'Daemon::SchedulerCronTaskManager::Task'}->{'Custom8'} = {  
  'Function' => '',  
  'MaximumParallelInstances' => '1',  
  'Module' => '',
```

```
'Params' => [],
'Schedule' => '* * * * *',
'TaskName' => 'Custom8'
};
```

### Daemon::SchedulerCronTaskManager::Task###Custom9

Executes a custom command or module. Note: if module is used, function is required.

这个设置默认没有激活。

默认值:

```
$Self->{'Daemon::SchedulerCronTaskManager::Task'}->{'Custom9'} = {
  'Function' => '',
  'MaximumParallelInstances' => '1',
  'Module' => '',
  'Params' => [],
  'Schedule' => '* * * * *',
  'TaskName' => 'Custom9'
};
```

### Daemon::SchedulerCronTaskManager::Task###GenericAgentFile1

Run file based generic agent jobs (Note: module name needs to be specified in -configuration-module param e.g. "Kernel::System::GenericAgent").

这个设置默认没有激活。

默认值:

```
$Self->{'Daemon::SchedulerCronTaskManager::Task'}->{'GenericAgentFile1'} = {
  'Function' => 'Execute',
  'MaximumParallelInstances' => '1',
  'Module' => 'Kernel::System::Console::Command::Maint::GenericAgent::Run',
  'Params' => [
    '--configuration-module',
    '<ModuleName>'
  ],
  'Schedule' => '*/20 * * * *',
  'TaskName' => 'GenericAgentFile1'
};
```

### Daemon::SchedulerCronTaskManager::Task###GenericAgentFile2

Run file based generic agent jobs (Note: module name needs to be specified in -configuration-module param e.g. "Kernel::System::GenericAgent").

这个设置默认没有激活。

默认值:

```
$Self->{'Daemon::SchedulerCronTaskManager::Task'}->{'GenericAgentFile2'} = {
  'Function' => 'Execute',
  'MaximumParallelInstances' => '1',
  'Module' => 'Kernel::System::Console::Command::Maint::GenericAgent::Run',
  'Params' => [
    '--configuration-module',
    '<ModuleName>'
  ],
  'Schedule' => '*/20 * * * *',
  'TaskName' => 'GenericAgentFile2'
};
```

### Daemon::SchedulerCronTaskManager::Task###GenericAgentFile3

Run file based generic agent jobs (Note: module name needs to be specified in -configuration-module param e.g. "Kernel::System::GenericAgent").

这个设置默认没有激活。

默认值:

```
$Self->{'Daemon::SchedulerCronTaskManager::Task'}->{'GenericAgentFile3'} = {  
  'Function' => 'Execute',  
  'MaximumParallelInstances' => '1',  
  'Module' => 'Kernel::System::Console::Command::Maint::GenericAgent::Run',  
  'Params' => [  
    '--configuration-module',  
    '<ModuleName>'  
  ],  
  'Schedule' => '* / 20 * * * *',  
  'TaskName' => 'GenericAgentFile3'  
};
```

#### **Daemon::SchedulerCronTaskManager::Task###GenericAgentFile4**

Run file based generic agent jobs (Note: module name needs to be specified in -configuration-module param e.g. "Kernel::System::GenericAgent").

这个设置默认没有激活。

默认值:

```
$Self->{'Daemon::SchedulerCronTaskManager::Task'}->{'GenericAgentFile4'} = {  
  'Function' => 'Execute',  
  'MaximumParallelInstances' => '1',  
  'Module' => 'Kernel::System::Console::Command::Maint::GenericAgent::Run',  
  'Params' => [  
    '--configuration-module',  
    '<ModuleName>'  
  ],  
  'Schedule' => '* / 20 * * * *',  
  'TaskName' => 'GenericAgentFile4'  
};
```

#### **Daemon::SchedulerCronTaskManager::Task###GenericAgentFile5**

Run file based generic agent jobs (Note: module name needs to be specified in -configuration-module param e.g. "Kernel::System::GenericAgent").

这个设置默认没有激活。

默认值:

```
$Self->{'Daemon::SchedulerCronTaskManager::Task'}->{'GenericAgentFile5'} = {  
  'Function' => 'Execute',  
  'MaximumParallelInstances' => '1',  
  'Module' => 'Kernel::System::Console::Command::Maint::GenericAgent::Run',  
  'Params' => [  
    '--configuration-module',  
    '<ModuleName>'  
  ],  
  'Schedule' => '* / 20 * * * *',  
  'TaskName' => 'GenericAgentFile5'  
};
```

#### **Daemon::SchedulerCronTaskManager::Task###RegistrationUpdateSend**

Sends registration information to OTRS group.

这个设置不能修改。

默认值:

```
$Self->{'Daemon::SchedulerCronTaskManager::Task'}->{'RegistrationUpdateSend'} = {  
  'Function' => 'Execute',
```

```
'MaximumParallelInstances' => '1',
'Module' => 'Kernel::System::Console::Command::Maint::Registration::UpdateSend',
'Params' => [],
'Schedule' => '30 * * * *',
'TaskName' => 'RegistrationUpdateSend'
};
```

### Daemon::SchedulerCronTaskManager::Task###SupportDataCollectAsynchronous

Collect support data for asynchronous plug-in modules.

这个设置不能修改。

默认值:

```
$Self->{'Daemon::SchedulerCronTaskManager::Task'}->{'SupportDataCollectAsynchronous'} =
{
  'Function' => 'Execute',
  'MaximumParallelInstances' => '1',
  'Module' =>
  'Kernel::System::Console::Command::Maint::SupportData::CollectAsynchronous',
  'Params' => [],
  'Schedule' => '1 * * * *',
  'TaskName' => 'SupportDataCollectAsynchronous'
};
```

### Daemon::SchedulerCronTaskManager::Task###OTRSBusinessEntitlementCheck

Checks the entitlement status of OTRS Business Solution™.

这个设置不能修改。

默认值:

```
$Self->{'Daemon::SchedulerCronTaskManager::Task'}->{'OTRSBusinessEntitlementCheck'} =
{
  'Function' => 'Execute',
  'MaximumParallelInstances' => '1',
  'Module' => 'Kernel::System::Console::Command::Maint::OTRSBusiness::EntitlementCheck',
  'Params' => [],
  'Schedule' => '25,45 */1 * * *',
  'TaskName' => 'OTRSBusinessEntitlementCheck'
};
```

### Daemon::SchedulerCronTaskManager::Task###OTRSBusinessAvailabilityCheck

Checks the availability of OTRS Business Solution™ for this system.

这个设置不能修改。

默认值:

```
$Self->{'Daemon::SchedulerCronTaskManager::Task'}->{'OTRSBusinessAvailabilityCheck'} =
{
  'Function' => 'Execute',
  'MaximumParallelInstances' => '1',
  'Module' =>
  'Kernel::System::Console::Command::Maint::OTRSBusiness::AvailabilityCheck',
  'Params' => [],
  'Schedule' => '15,35,55 */1 * * *',
  'TaskName' => 'OTRSBusinessAvailabilityCheck'
};
```

## Daemon → Daemon::SchedulerGenericAgentTaskManager

### Daemon::SchedulerGenericAgentTaskManager::TicketLimit

Defines the maximum number of affected tickets per job.

不能使这个设置失效。

默认值:

```
$Self->{'Daemon::SchedulerGenericAgentTaskManager::TicketLimit'} = '4000';
```

### **Daemon::SchedulerGenericAgentTaskManager::SleepTime**

Defines a sleep time in microseconds between tickets while they are been processed by a job.

不能使这个设置失效。

默认值:

```
$Self->{'Daemon::SchedulerGenericAgentTaskManager::SleepTime'} = '0';
```

### **Daemon → Daemon::SchedulerGenericInterfaceTaskManager**

#### **Daemon::SchedulerGenericInterfaceTaskManager::FutureTaskTimeDiff**

Defines the default the number of seconds (from current time) to re-schedule a generic interface failed task.

不能使这个设置失效。

默认值:

```
$Self->{'Daemon::SchedulerGenericInterfaceTaskManager::FutureTaskTimeDiff'} = '300';
```

### **Daemon → Daemon::SchedulerTaskWorker**

#### **Daemon::SchedulerTaskWorker::MaximumWorkers**

Defines the maximum number of tasks to be executed as the same time.

不能使这个设置失效。

默认值:

```
$Self->{'Daemon::SchedulerTaskWorker::MaximumWorkers'} = '5';
```

#### **Daemon::SchedulerTaskWorker::NotificationRecipientEmail**

Specifies the email addresses to get notification messages from scheduler tasks.

不能使这个设置失效。

默认值:

```
$Self->{'Daemon::SchedulerTaskWorker::NotificationRecipientEmail'} = 'root@localhost';
```

### **Daemon → Frontend::Admin::ModuleRegistration**

#### **Frontend::Module###AgentDaemonInfo**

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AgentDaemonInfo'} = {  
  'Description' => 'Admin',
```

```
'Group' => [
  'admin'
],
'Title' => 'Shows information on how to start OTRS Daemon'
};
```

## Daemon → Frontend::Agent::ModuleNotify

### Frontend::NotifyModule###800-Daemon-Check

Defines the module to display a notification in the agent interface if the OTRS Daemon is not running.

默认值:

```
$Self->{'Frontend::NotifyModule'}->{'800-Daemon-Check'} = {
  'Module' => 'Kernel::Output::HTML::Notification::DaemonCheck'
};
```

## 3. DynamicFields

### DynamicFields → DynamicFields::Driver::Registration

#### DynamicFields::Driver###Text

DynamicField backend registration.

默认值:

```
$Self->{'DynamicFields::Driver'}->{'Text'} = {
  'ConfigDialog' => 'AdminDynamicFieldText',
  'DisplayName' => 'Text',
  'Module' => 'Kernel::System::DynamicField::Driver::Text'
};
```

#### DynamicFields::Driver###TextArea

DynamicField backend registration.

默认值:

```
$Self->{'DynamicFields::Driver'}->{'TextArea'} = {
  'ConfigDialog' => 'AdminDynamicFieldText',
  'DisplayName' => 'Textarea',
  'Module' => 'Kernel::System::DynamicField::Driver::TextArea'
};
```

#### DynamicFields::Driver###Checkbox

DynamicField backend registration.

默认值:

```
$Self->{'DynamicFields::Driver'}->{'Checkbox'} = {
  'ConfigDialog' => 'AdminDynamicFieldCheckbox',
  'DisplayName' => 'Checkbox',
  'Module' => 'Kernel::System::DynamicField::Driver::Checkbox'
};
```

#### DynamicFields::Driver###Dropdown

DynamicField backend registration.

默认值:

```
$Self->{'DynamicFields::Driver'}->{'Dropdown'} = {  
  'ConfigDialog' => 'AdminDynamicFieldDropdown',  
  'DisplayName' => 'Dropdown',  
  'Module' => 'Kernel::System::DynamicField::Driver::Dropdown'  
};
```

### DynamicFields::Driver###DateTime

DynamicField backend registration.

默认值:

```
$Self->{'DynamicFields::Driver'}->{'DateTime'} = {  
  'ConfigDialog' => 'AdminDynamicFieldDateTime',  
  'DisplayName' => 'Date / Time',  
  'Module' => 'Kernel::System::DynamicField::Driver::DateTime'  
};
```

### DynamicFields::Driver###Date

DynamicField backend registration.

默认值:

```
$Self->{'DynamicFields::Driver'}->{'Date'} = {  
  'ConfigDialog' => 'AdminDynamicFieldDateTime',  
  'DisplayName' => 'Date',  
  'Module' => 'Kernel::System::DynamicField::Driver::Date'  
};
```

### DynamicFields::Driver###Multiselect

DynamicField backend registration.

默认值:

```
$Self->{'DynamicFields::Driver'}->{'Multiselect'} = {  
  'ConfigDialog' => 'AdminDynamicFieldMultiselect',  
  'DisplayName' => 'Multiselect',  
  'ItemSeparator' => ', ',  
  'Module' => 'Kernel::System::DynamicField::Driver::Multiselect'  
};
```

## DynamicFields → DynamicFields::ObjectType::Registration

### DynamicFields::ObjectType###Article

DynamicField object registration.

默认值:

```
$Self->{'DynamicFields::ObjectType'}->{'Article'} = {  
  'DisplayName' => 'Article',  
  'Module' => 'Kernel::System::DynamicField::ObjectType::Article',  
  'Prio' => '110'  
};
```

### DynamicFields::ObjectType###Ticket

DynamicField object registration.

默认值:

```
$Self->{'DynamicFields::ObjectType'}->{'Ticket'} = {  
  'DisplayName' => 'Ticket',  
  'Module' => 'Kernel::System::DynamicField::ObjectType::Ticket',  
  'Prio' => '100'  
};
```



```
};
```

## DynamicFields → Frontend::Admin::ModuleRegistration

### Frontend::Module###AdminDynamicField

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminDynamicField'} = {  
  'Description' => 'This module is part of the admin area of OTRS.',  
  'Group' => [  
    'admin'  
  ],  
  'Loader' => {  
    'CSS' => [  
      'Core.Agent.Admin.DynamicField.css'  
    ],  
    'JavaScript' => [  
      'Core.Agent.Admin.DynamicField.js'  
    ]  
  },  
  'NavBarModule' => {  
    'Block' => 'Ticket',  
    'Description' => 'Create and manage dynamic fields.',  
    'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',  
    'Name' => 'Dynamic Fields',  
    'Prio' => '1000'  
  },  
  'NavBarName' => 'Admin',  
  'Title' => 'Dynamic Fields GUI'  
};
```

### Frontend::Module###AdminDynamicFieldText

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminDynamicFieldText'} = {  
  'Description' => 'This module is part of the admin area of OTRS.',  
  'Group' => [  
    'admin'  
  ],  
  'Loader' => {  
    'JavaScript' => [  
      'Core.Agent.Admin.DynamicField.js',  
      'Core.Agent.Admin.DynamicFieldText.js'  
    ]  
  },  
  'Title' => 'Dynamic Fields Text Backend GUI'  
};
```

### Frontend::Module###AdminDynamicFieldCheckbox

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminDynamicFieldCheckbox'} = {  
  'Description' => 'This module is part of the admin area of OTRS.',  
  'Group' => [  
    'admin'  
  ],  
  'Loader' => {  
    'JavaScript' => [  
      'Core.Agent.Admin.DynamicField.js'  
    ]  
  }  
};
```

```

},
'Title' => 'Dynamic Fields Checkbox Backend GUI'
};

```

### Frontend::Module###AdminDynamicFieldDropdown

Frontend module registration for the agent interface.

默认值:

```

$self->{'Frontend::Module'}->{'AdminDynamicFieldDropdown'} = {
  'Description' => 'This module is part of the admin area of OTRS.',
  'Group' => [
    'admin'
  ],
  'Loader' => {
    'CSS' => [
      'Core.Agent.Admin.DynamicField.css'
    ],
    'JavaScript' => [
      'Core.Agent.Admin.DynamicField.js',
      'Core.Agent.Admin.DynamicFieldDropdown.js'
    ]
  },
  'Title' => 'Dynamic Fields Drop-down Backend GUI'
};

```

### Frontend::Module###AdminDynamicFieldDateTime

Frontend module registration for the agent interface.

默认值:

```

$self->{'Frontend::Module'}->{'AdminDynamicFieldDateTime'} = {
  'Description' => 'This module is part of the admin area of OTRS.',
  'Group' => [
    'admin'
  ],
  'Loader' => {
    'CSS' => [
      'Core.Agent.Admin.DynamicField.css'
    ],
    'JavaScript' => [
      'Core.Agent.Admin.DynamicField.js',
      'Core.Agent.Admin.DynamicFieldDateTime.js'
    ]
  },
  'Title' => 'Dynamic Fields Date Time Backend GUI'
};

```

### Frontend::Module###AdminDynamicFieldMultiselect

Frontend module registration for the agent interface.

默认值:

```

$self->{'Frontend::Module'}->{'AdminDynamicFieldMultiselect'} = {
  'Description' => 'This module is part of the admin area of OTRS.',
  'Group' => [
    'admin'
  ],
  'Loader' => {
    'CSS' => [
      'Core.Agent.Admin.DynamicField.css'
    ],
    'JavaScript' => [
      'Core.Agent.Admin.DynamicField.js',
      'Core.Agent.Admin.DynamicFieldMultiselect.js'
    ]
  },
};

```

```
'Title' => 'Dynamic Fields Multiselect Backend GUI'
};
```

## DynamicFields → Frontend::Agent::Preferences

### PreferencesGroups###DynamicField

Defines the config parameters of this item, to be shown in the preferences view.

这个设置默认没有激活。

默认值:

```
$Self->{'PreferencesGroups'}->{'DynamicField'} = {
  'Active' => '1',
  'Block' => 'Input',
  'Column' => 'Other Settings',
  'Data' => "[% Env('UserDynamicField_NameX') %]",
  'Desc' => 'Add a default name for Dynamic Field.',
  'Key' => 'Default value for NameX',
  'Label' => 'NameX',
  'Module' => 'Kernel::Output::HTML::Preferences::Generic',
  'PrefKey' => 'UserDynamicField_NameX',
  'Prio' => '7000'
};
```

### PreferencesGroups###DynamicFieldsOverviewPageShown

Parameters for the pages (in which the dynamic fields are shown) of the dynamic fields overview.

默认值:

```
$Self->{'PreferencesGroups'}->{'DynamicFieldsOverviewPageShown'} = {
  'Active' => '0',
  'Column' => 'Other Settings',
  'Data' => {
    '10' => '10',
    '15' => '15',
    '20' => '20',
    '25' => '25',
    '30' => '30',
    '35' => '35'
  },
  'DataSelected' => '25',
  'Key' => 'Dynamic fields limit per page for Dynamic Fields Overview',
  'Label' => 'Dynamic Fields Overview Limit',
  'Module' => 'Kernel::Output::HTML::Preferences::Generic',
  'PrefKey' => 'AdminDynamicFieldsOverviewPageShown',
  'Prio' => '8000'
};
```

## 4. Framework

### Framework → Core

#### SecureMode

Disables the web installer (<http://yourhost.example.com/otrs/installer.pl>), to prevent the system from being hijacked. If set to "No", the system can be reinstalled and the current basic configuration will be used to pre-populate the questions within the installer script. If not active, it also disables the GenericAgent, PackageManager and SQL Box.

不能使这个设置失效。

默认值:

```
$Self->{'SecureMode'} = '0';
```

### Frontend::DebugMode

Enables or disables the debug mode over frontend interface.

默认值:

```
$Self->{'Frontend::DebugMode'} = '0';
```

### Frontend::AjaxDebug

Delivers extended debugging information in the frontend in case any AJAX errors occur, if enabled.

不能使这个设置失效。

默认值:

```
$Self->{'Frontend::AjaxDebug'} = '0';
```

### Frontend::TemplateCache

Enables or disables the caching for templates. **WARNING: Do NOT disable template caching for production environments for it will cause a massive performance drop! This setting should only be disabled for debugging reasons!**

不能使这个设置失效。

默认值:

```
$Self->{'Frontend::TemplateCache'} = '1';
```

### ConfigLevel

Sets the configuration level of the administrator. Depending on the config level, some sysconfig options will be not shown. The config levels are in ascending order: Expert, Advanced, Beginner. The higher the config level is (e.g. Beginner is the highest), the less likely is it that the user can accidentally configure the system in a way that it is not usable any more.

不能使这个设置失效。

默认值:

```
$Self->{'ConfigLevel'} = '100';
```

### ConfigImportAllowed

Controls if the admin is allowed to import a saved system configuration in SysConfig.

不能使这个设置失效。

默认值:

```
$Self->{'ConfigImportAllowed'} = '1';
```

### ProductName

Defines the name of the application, shown in the web interface, tabs and title bar of the web browser.

不能使这个设置失效。

默认值:

```
$Self->{'ProductName'} = 'OTRS 5s';
```

## SystemID

Defines the system identifier. Every ticket number and http session string contains this ID. This ensures that only tickets which belong to your system will be processed as follow-ups (useful when communicating between two instances of OTRS).

不能使这个设置失效。

默认值:

```
$Self->{'SystemID'} = '10';
```

## FQDN

Defines the fully qualified domain name of the system. This setting is used as a variable, OTRS\_CONFIG\_FQDN which is found in all forms of messaging used by the application, to build links to the tickets within your system.

不能使这个设置失效。

默认值:

```
$Self->{'FQDN'} = 'yourhost.example.com';
```

## SupportDataCollector::HTTPHostname

Defines the HTTP hostname for the support data collection with the public module 'PublicSupportDataCollector' (e.g. used from the OTRS Daemon).

这个设置默认没有激活。

默认值:

```
$Self->{'SupportDataCollector::HTTPHostname'} = '';
```

## SupportDataCollector::WebUserAgent::Timeout

Defines the timeout (in seconds, minimum is 20 seconds) for the support data collection with the public module 'PublicSupportDataCollector' (e.g. used from the OTRS Daemon).

这个设置默认没有激活。

默认值:

```
$Self->{'SupportDataCollector::WebUserAgent::Timeout'} = '20';
```

## NodeID

Defines the cluster node identifier. This is only used in cluster configurations where there is more than one OTRS frontend system. Note: only values from 1 to 99 are allowed.

这个设置默认没有激活。

默认值:

```
$Self->{'NodeID'} = '1';
```

## HttpType

Defines the type of protocol, used by the web server, to serve the application. If https protocol will be used instead of plain http, it must be specified here. Since this has no affect on the web server's settings or behavior, it will not change the method of access to the application and, if it is wrong, it will not prevent you from logging into the application. This setting is only used as a variable, OTRS\_CONFIG\_HttpType which is found in all forms of messaging used by the application, to build links to the tickets within your system.

不能使这个设置失效。

默认值:

```
$Self->{'HttpType'} = 'http';
```

## ScriptAlias

Sets the prefix to the scripts folder on the server, as configured on the web server. This setting is used as a variable, OTRS\_CONFIG\_ScriptAlias which is found in all forms of messaging used by the application, to build links to the tickets within the system.

不能使这个设置失效。

默认值:

```
$Self->{'ScriptAlias'} = 'otrs/';
```

## AdminEmail

Defines the system administrator's email address. It will be displayed in the error screens of the application.

不能使这个设置失效。

默认值:

```
$Self->{'AdminEmail'} = 'admin@example.com';
```

## Organization

Company name which will be included in outgoing emails as an X-Header.

不能使这个设置失效。

默认值:

```
$Self->{'Organization'} = 'Example Company';
```

## DefaultLanguage

Defines the default front-end language. All the possible values are determined by the available language files on the system (see the next setting).

不能使这个设置失效。

默认值:

```
$Self->{'DefaultLanguage'} = 'en';
```

## DefaultUsedLanguages

Defines all the languages that are available to the application. Specify only English names of languages here.

不能使这个设置失效。

默认值:

```
$Self->{'DefaultUsedLanguages'} = {
  'ar_SA' => 'Arabic (Saudi Arabia)',
  'bg' => 'Bulgarian',
  'ca' => 'Catalan',
  'cs' => 'Czech',
  'da' => 'Danish',
  'de' => 'German',
  'el' => 'Greek',
  'en' => 'English (United States)',
  'en_CA' => 'English (Canada)',
  'en_GB' => 'English (United Kingdom)',
  'es' => 'Spanish',
  'es_CO' => 'Spanish (Colombia)',
  'es_MX' => 'Spanish (Mexico)',
  'et' => 'Estonian',
  'fa' => 'Persian',
  'fi' => 'Finnish',
  'fr' => 'French',
  'fr_CA' => 'French (Canada)',
  'gl' => 'Galician',
  'he' => 'Hebrew',
  'hi' => 'Hindi',
  'hr' => 'Croatian',
  'hu' => 'Hungarian',
  'id' => 'Indonesian',
  'it' => 'Italian',
  'ja' => 'Japanese',
  'lt' => 'Lithuanian',
  'lv' => 'Latvian',
  'ms' => 'Malay',
  'nb_NO' => 'Norwegian',
  'nl' => 'Nederlands',
  'pl' => 'Polish',
  'pt' => 'Portuguese',
  'pt_BR' => 'Portuguese (Brasil)',
  'ru' => 'Russian',
  'sk_SK' => 'Slovak',
  'sl' => 'Slovenian',
  'sr_Cyrl' => 'Serbian Cyrillic',
  'sr_Latn' => 'Serbian Latin',
  'sv' => 'Swedish',
  'sw' => 'Swahili',
  'th_TH' => 'Thai',
  'tr' => 'Turkish',
  'uk' => 'Ukrainian',
  'vi_VN' => 'Vietnam',
  'zh_CN' => 'Chinese (Simplified)',
  'zh_TW' => 'Chinese (Traditional)'
};
```

### DefaultUsedLanguagesNative

Defines all the languages that are available to the application. Specify only native names of languages here.

不能使这个设置失效。

默认值:

```
$Self->{'DefaultUsedLanguagesNative'} = {
  'ar_SA' => 'العَرَبِيَّةُ',
  'bg' => 'Български',
  'ca' => 'Català',
  'cs' => 'Česky',
  'da' => 'Dansk',
  'de' => 'Deutsch',
  'el' => 'Ελληνικά',
```

```
'en' => 'English (United States)',
'en_CA' => 'English (Canada)',
'en_GB' => 'English (United Kingdom)',
'es' => 'Español',
'es_CO' => 'Español (Colombia)',
'es_MX' => 'Español (México)',
'et' => 'Eesti',
'fa' => 'فارسی',
'fi' => 'Suomi',
'fr' => 'Français',
'fr_CA' => 'Français (Canada)',
'gl' => 'Galego',
'he' => '#####',
'hi' => '#####',
'hr' => 'Hrvatski',
'hu' => 'Magyar',
'id' => 'Bahasa Indonesia',
'it' => 'Italiano',
'ja' => '日本語',
'lt' => 'Lietuvių kalba',
'lv' => 'Latvijas',
'ms' => 'Melayu',
'nb_NO' => 'Norsk bokmål',
'nl' => 'Nederlandse',
'pl' => 'Polski',
'pt' => 'Português',
'pt_BR' => 'Português Brasileiro',
'ru' => 'Русский',
'sk_SK' => 'Slovenčina',
'sl' => 'Slovenščina',
'sr_Cyrl' => 'Српски',
'sr_Latn' => 'Srpski',
'sv' => 'Svenska',
'sw' => 'Kiswahili',
'th_TH' => '#####',
'tr' => 'Türkçe',
'uk' => 'Українська',
'vi_VN' => 'Việt Nam',
'zh_CN' => '简体中文',
'zh_TW' => '正體中文'
};
```

## DefaultTheme

Defines the default front-end (HTML) theme to be used by the agents and customers. If you like, you can add your own theme. Please refer the administrator manual located at <http://otrs.github.io/doc/>.

不能使这个设置失效。

默认值:

```
$Self->{'DefaultTheme'} = 'Standard';
```

## DefaultTheme::HostBased

It is possible to configure different themes, for example to distinguish between agents and customers, to be used on a per-domain basis within the application. Using a regular expression (regex), you can configure a Key/Content pair to match a domain. The value in "Key" should match the domain, and the value in "Content" should be a valid theme on your system. Please see the example entries for the proper form of the regex.

这个设置默认没有激活。

默认值:

```
$Self->{'DefaultTheme::HostBased'} = {
  'host1\\.example\\.com' => 'SomeTheme1',
```





默认值:

```
$Self->{'CGILogPrefix'} = 'OTRS-CGI';
```

### DemoSystem

Runs the system in "Demo" mode. If set to "Yes", agents can change preferences, such as selection of language and theme via the agent web interface. These changes are only valid for the current session. It will not be possible for agents to change their passwords.

不能使这个设置失效。

默认值:

```
$Self->{'DemoSystem'} = '0';
```

### OutOfOfficeMessageTemplate

Defines out of office message template. Two string parameters (%s) available: end date and number of days left.

默认值:

```
$Self->{'OutOfOfficeMessageTemplate'} = '*** out of office until %s (%s d left) ***';
```

### SwitchToUser

Allows the administrators to login as other users, via the users administration panel.

不能使这个设置失效。

默认值:

```
$Self->{'SwitchToUser'} = '0';
```

### SwitchToCustomer

Allows the administrators to login as other customers, via the customer user administration panel.

不能使这个设置失效。

默认值:

```
$Self->{'SwitchToCustomer'} = '0';
```

### SwitchToCustomer::PermissionGroup

Specifies the group where the user needs rw permissions so that he can access the "SwitchToCustomer" feature.

不能使这个设置失效。

默认值:

```
$Self->{'SwitchToCustomer::PermissionGroup'} = 'admin';
```

### NotificationSenderName

Specifies the name that should be used by the application when sending notifications. The sender name is used to build the complete display name for the notification master (i.e. "OTRS Notifications" otrs@your.example.com).

不能使这个设置失效。

默认值:

```
$Self->{'NotificationSenderName'} = 'OTRS Notifications';
```

### NotificationSenderEmail

Specifies the email address that should be used by the application when sending notifications. The email address is used to build the complete display name for the notification master (i.e. "OTRS Notifications" otrs@your.example.com). You can use the OTRS\_CONFIG\_FQDN variable as set in your configuration, or choose another email address.

不能使这个设置失效。

默认值:

```
$Self->{'NotificationSenderEmail'} = 'otrs@<OTRS_CONFIG_FQDN>';
```

### System::Customer::Permission

Defines the standard permissions available for customers within the application. If more permissions are needed, you can enter them here. Permissions must be hard coded to be effective. Please ensure, when adding any of the afore mentioned permissions, that the "rw" permission remains the last entry.

不能使这个设置失效。

默认值:

```
$Self->{'System::Customer::Permission'} = [  
  'ro',  
  'rw'  
];
```

### LanguageDebug

Debugs the translation set. If this is set to "Yes" all strings (text) without translations are written to STDERR. This can be helpful when you are creating a new translation file. Otherwise, this option should remain set to "No".

不能使这个设置失效。

默认值:

```
$Self->{'LanguageDebug'} = '0';
```

### Secure::DisableBanner

If enabled, the OTRS version tag will be removed from the Webinterface, the HTTP headers and the X-Headers of outgoing mails. NOTE: If you change this option, please make sure to delete the cache.

不能使这个设置失效。

默认值:

```
$Self->{'Secure::DisableBanner'} = '0';
```

## Framework → Core::Cache

### Cache::Module

Selects the cache backend to use.

不能使这个设置失效。

默认值:

```
$Self->{'Cache::Module'} = 'Kernel::System::Cache::FileStorable';
```

### Cache::InMemory

Should the cache data be held in memory?

不能使这个设置失效。

默认值:

```
$Self->{'Cache::InMemory'} = '1';
```

### Cache::InBackend

Should the cache data be stored in the selected cache backend?

不能使这个设置失效。

默认值:

```
$Self->{'Cache::InBackend'} = '1';
```

### Cache::SubdirLevels

Specify how many sub directory levels to use when creating cache files. This should prevent too many cache files being in one directory.

不能使这个设置失效。

默认值:

```
$Self->{'Cache::SubdirLevels'} = '2';
```

## Framework → Core::CustomerCompany

### CustomerCompany::EventModulePost###100-UpdateCustomerUsers

Event module that updates customer users after an update of the Customer.

默认值:

```
$Self->{'CustomerCompany::EventModulePost'}->{'100-UpdateCustomerUsers'} = {
  'Event' => 'CustomerCompanyUpdate',
  'Module' => 'Kernel::System::CustomerCompany::Event::CustomerUserUpdate',
  'Transaction' => '0'
};
```

## Framework → Core::CustomerUser

### CustomerUser::EventModulePost###100-UpdateSearchProfiles

Event module that updates customer user search profiles if login changes.

默认值:

```
$Self->{'CustomerUser::EventModulePost'}->{'100-UpdateSearchProfiles'} = {
  'Event' => 'CustomerUserUpdate',
  'Module' => 'Kernel::System::CustomerUser::Event::SearchProfileUpdate',
  'Transaction' => '0'
};
```

## CustomerUser::EventModulePost###100-UpdateServiceMembership

Event module that updates customer user service membership if login changes.

默认值:

```
$Self->{'CustomerUser::EventModulePost'}->{'100-UpdateServiceMembership'} = {
  'Event' => 'CustomerUserUpdate',
  'Module' => 'Kernel::System::CustomerUser::Event::ServiceMemberUpdate',
  'Transaction' => '0'
};
```

## Framework → Core::Fetchmail

### Fetchmail::Bin

Defines the fall-back path to open fetchmail binary. Note: The name of the binary needs to be 'fetchmail', if it is different please use a symbolic link.

这个设置默认没有激活。

默认值:

```
$Self->{'Fetchmail::Bin'} = '/usr/bin/fetchmail';
```

## Framework → Core::LinkObject

### LinkObject::ViewMode

Determines the way the linked objects are displayed in each zoom mask.

不能使这个设置失效。

默认值:

```
$Self->{'LinkObject::ViewMode'} = 'Simple';
```

### LinkObject::Type###Normal

Defines the link type 'Normal'. If the source name and the target name contain the same value, the resulting link is a non-directional one; otherwise, the result is a directional link.

不能使这个设置失效。

默认值:

```
$Self->{'LinkObject::Type'}->{'Normal'} = {
  'SourceName' => 'Normal',
  'TargetName' => 'Normal'
};
```

### LinkObject::Type###ParentChild

Defines the link type 'ParentChild'. If the source name and the target name contain the same value, the resulting link is a non-directional one; otherwise, the result is a directional link.

不能使这个设置失效。

默认值:

```
$Self->{'LinkObject::Type'}->{'ParentChild'} = {
  'SourceName' => 'Parent',
  'TargetName' => 'Child'
};
```

## LinkObject::TypeGroup###0001

Defines the link type groups. The link types of the same group cancel one another. Example: If ticket A is linked per a 'Normal' link with ticket B, then these tickets could not be additionally linked with link of a 'ParentChild' relationship.

默认值:

```
$Self->{'LinkObject::TypeGroup'}->{'0001'} = [  
  'Normal',  
  'ParentChild'  
];
```

## Framework → Core::Log

### LogModule

Defines the log module for the system. "File" writes all messages in a given logfile, "SysLog" uses the syslog daemon of the system, e.g. syslogd.

不能使这个设置失效。

默认值:

```
$Self->{'LogModule'} = 'Kernel::System::Log::SysLog';
```

### LogModule::SysLog::Facility

If "SysLog" was selected for LogModule, a special log facility can be specified.

不能使这个设置失效。

默认值:

```
$Self->{'LogModule::SysLog::Facility'} = 'user';
```

### LogModule::SysLog::LogSock

If "SysLog" was selected for LogModule, a special log sock can be specified (on solaris you may need to use 'stream').

不能使这个设置失效。

默认值:

```
$Self->{'LogModule::SysLog::LogSock'} = 'unix';
```

### LogModule::SysLog::Charset

If "SysLog" was selected for LogModule, the charset that should be used for logging can be specified.

不能使这个设置失效。

默认值:

```
$Self->{'LogModule::SysLog::Charset'} = 'utf-8';
```

### LogModule::LogFile

If "file" was selected for LogModule, a logfile must be specified. If the file doesn't exist, it will be created by the system.

不能使这个设置失效。

默认值:

```
$Self->{'LogModule::LogFile'} = '/tmp/otrs.log';
```

### LogModule::LogFile::Date

Adds a suffix with the actual year and month to the OTRS log file. A logfile for every month will be created.

不能使这个设置失效。

默认值:

```
$Self->{'LogModule::LogFile::Date'} = '0';
```

### MinimumLogLevel

Set the minimum log level. If you select 'error', just errors are logged. With 'debug' you get all logging messages. The order of log levels is: 'debug', 'info', 'notice' and 'error'.

不能使这个设置失效。

默认值:

```
$Self->{'MinimumLogLevel'} = 'error';
```

## Framework → Core::MIME-Viewer

### MIME-Viewer###application/excel

Specifies the path to the converter that allows the view of Microsoft Excel files, in the web interface.

这个设置默认没有激活。

默认值:

```
$Self->{'MIME-Viewer'}->{'application/excel'} = 'xlhtml';
```

### MIME-Viewer###application/msword

Specifies the path to the converter that allows the view of Microsoft Word files, in the web interface.

这个设置默认没有激活。

默认值:

```
$Self->{'MIME-Viewer'}->{'application/msword'} = 'wvWare';
```

### MIME-Viewer###application/pdf

Specifies the path to the converter that allows the view of PDF documents, in the web interface.

这个设置默认没有激活。

默认值:

```
$Self->{'MIME-Viewer'}->{'application/pdf'} = 'pdftohtml -stdout -i';
```

### MIME-Viewer###text/xml

Specifies the path to the converter that allows the view of XML files, in the web interface.

这个设置默认没有激活。

默认值:

```
$Self->{'MIME-Viewer'}->{'text/xml'} = '<OTRS_CONFIG_Home>/scripts/tools/xml2html.pl';
```

## Framework → Core::MirrorDB

### Core::MirrorDB::DSN

OTRS can use one or more readonly mirror databases for expensive operations like fulltext search or statistics generation. Here you can specify the DSN for the first mirror database.

这个设置默认没有激活。

默认值:

```
$Self->{'Core::MirrorDB::DSN'} = 'DBI:mysql:database=mirrordb;host=mirrordbhost';
```

### Core::MirrorDB::User

Specify the username to authenticate for the first mirror database.

这个设置默认没有激活。

默认值:

```
$Self->{'Core::MirrorDB::User'} = 'some_user';
```

### Core::MirrorDB::Password

Specify the password to authenticate for the first mirror database.

这个设置默认没有激活。

默认值:

```
$Self->{'Core::MirrorDB::Password'} = 'some_password';
```

### Core::MirrorDB::AdditionalMirrors###1

Configure any additional readonly mirror databases that you want to use.

这个设置默认没有激活。

默认值:

```
$Self->{'Core::MirrorDB::AdditionalMirrors'}->{'1'} = {  
  'DSN' => 'DBI:mysql:database=mirrordb;host=mirrordbhost',  
  'Password' => 'some_password',  
  'User' => 'some_user'  
};
```

### Core::MirrorDB::AdditionalMirrors###2

Configure any additional readonly mirror databases that you want to use.

这个设置默认没有激活。

默认值:

```
$Self->{'Core::MirrorDB::AdditionalMirrors'}->{'2'} = {  
  'DSN' => 'DBI:mysql:database=mirrordb;host=mirrordbhost',  
  'Password' => 'some_password',  
  'User' => 'some_user'  
};
```



```
};
```

### Core::MirrorDB::AdditionalMirrors###3

Configure any additional readonly mirror databases that you want to use.

这个设置默认没有激活。

默认值:

```
$Self->{'Core::MirrorDB::AdditionalMirrors'}->{'3'} = {  
  'DSN' => 'DBI:mysql:database=mirrordb;host=mirrordbhost',  
  'Password' => 'some_password',  
  'User' => 'some_user'  
};
```

### Core::MirrorDB::AdditionalMirrors###4

Configure any additional readonly mirror databases that you want to use.

这个设置默认没有激活。

默认值:

```
$Self->{'Core::MirrorDB::AdditionalMirrors'}->{'4'} = {  
  'DSN' => 'DBI:mysql:database=mirrordb;host=mirrordbhost',  
  'Password' => 'some_password',  
  'User' => 'some_user'  
};
```

### Core::MirrorDB::AdditionalMirrors###5

Configure any additional readonly mirror databases that you want to use.

这个设置默认没有激活。

默认值:

```
$Self->{'Core::MirrorDB::AdditionalMirrors'}->{'5'} = {  
  'DSN' => 'DBI:mysql:database=mirrordb;host=mirrordbhost',  
  'Password' => 'some_password',  
  'User' => 'some_user'  
};
```

## Framework → Core::OTRSBusiness

### OTRSBusiness::ReleaseChannel

Specify the channel to be used to fetch OTRS Business Solution™ updates. Warning: Development releases might not be complete, your system might experience unrecoverable errors and on extreme cases could become unresponsive!

不能使这个设置失效。

默认值:

```
$Self->{'OTRSBusiness::ReleaseChannel'} = '1';
```

## Framework → Core::PDF

### PDF::LogoFile

Specifies the path of the file for the logo in the page header (gif|jpg|png, 700 x 100 pixel).

不能使这个设置失效。

默认值:

```
$Self->{'PDF::LogoFile'} = '<OTRS_CONFIG_Home>/var/logo-otrs.png';
```

### **PDF::PageSize**

Defines the standard size of PDF pages.

不能使这个设置失效。

默认值:

```
$Self->{'PDF::PageSize'} = 'a4';
```

### **PDF::MaxPages**

Defines the maximum number of pages per PDF file.

不能使这个设置失效。

默认值:

```
$Self->{'PDF::MaxPages'} = '100';
```

### **PDF::TTFontFile###Proportional**

Defines the path and TTF-File to handle proportional font in PDF documents.

不能使这个设置失效。

默认值:

```
$Self->{'PDF::TTFontFile'}->{'Proportional'} = 'DejaVuSans.ttf';
```

### **PDF::TTFontFile###ProportionalBold**

Defines the path and TTF-File to handle bold proportional font in PDF documents.

不能使这个设置失效。

默认值:

```
$Self->{'PDF::TTFontFile'}->{'ProportionalBold'} = 'DejaVuSans-Bold.ttf';
```

### **PDF::TTFontFile###ProportionalItalic**

Defines the path and TTF-File to handle italic proportional font in PDF documents.

不能使这个设置失效。

默认值:

```
$Self->{'PDF::TTFontFile'}->{'ProportionalItalic'} = 'DejaVuSans-Oblique.ttf';
```

### **PDF::TTFontFile###ProportionalBoldItalic**

Defines the path and TTF-File to handle bold italic proportional font in PDF documents.

不能使这个设置失效。

默认值:

```
$Self->{'PDF::TTFontFile'}->{'ProportionalBoldItalic'} = 'DejaVuSans-BoldOblique.ttf';
```

### **PDF::TTFontFile###Monospaced**

Defines the path and TTF-File to handle monospaced font in PDF documents.

不能使这个设置失效。

默认值:

```
$Self->{'PDF::TTFontFile'}->{'Monospaced'} = 'DejaVuSansMono.ttf';
```

### **PDF::TTFontFile###MonospacedBold**

Defines the path and TTF-File to handle bold monospaced font in PDF documents.

不能使这个设置失效。

默认值:

```
$Self->{'PDF::TTFontFile'}->{'MonospacedBold'} = 'DejaVuSansMono-Bold.ttf';
```

### **PDF::TTFontFile###MonospacedItalic**

Defines the path and TTF-File to handle italic monospaced font in PDF documents.

不能使这个设置失效。

默认值:

```
$Self->{'PDF::TTFontFile'}->{'MonospacedItalic'} = 'DejaVuSansMono-Oblique.ttf';
```

### **PDF::TTFontFile###MonospacedBoldItalic**

Defines the path and TTF-File to handle bold italic monospaced font in PDF documents.

不能使这个设置失效。

默认值:

```
$Self->{'PDF::TTFontFile'}->{'MonospacedBoldItalic'} = 'DejaVuSansMono-BoldOblique.ttf';
```

## **Framework → Core::Package**

### **Package::FileUpload**

Enables file upload in the package manager frontend.

不能使这个设置失效。

默认值:

```
$Self->{'Package::FileUpload'} = '1';
```

### **Package::RepositoryRoot**

Defines the location to get online repository list for additional packages. The first available result will be used.

默认值:

```
$Self->{'Package::RepositoryRoot'} = [  
'https://ftp.otrs.org/pub/otrs/misc/packages/repository.xml'  
];
```

### **Package::RepositoryList**

Defines the list of online repositories. Another installations can be used as repository, for example: Key="http://example.com/otrs/public.pl?Action=PublicRepository;File=" and Content="Some Name".

这个设置默认没有激活。

默认值:

```
$Self->{'Package::RepositoryList'} = {  
  'ftp://ftp.example.com/pub/otrs/misc/packages/' => '[Example] ftp://ftp.example.com/'  
};
```

### **Package::RepositoryAccessRegExp**

Defines the IP regular expression for accessing the local repository. You need to enable this to have access to your local repository and the `Package::RepositoryList` is required on the remote host.

这个设置默认没有激活。

默认值:

```
$Self->{'Package::RepositoryAccessRegExp'} = '127\\.0\\.0\\.1';
```

### **Package::Timeout**

Sets the timeout (in seconds) for package downloads. Overwrites "WebUserAgent::Timeout".

不能使这个设置失效。

默认值:

```
$Self->{'Package::Timeout'} = '120';
```

### **Package::Proxy**

Fetches packages via proxy. Overwrites "WebUserAgent::Proxy".

这个设置默认没有激活。

默认值:

```
$Self->{'Package::Proxy'} = 'http://proxy.sn.no:8001/';
```

### **Package::AllowLocalModifications**

If this setting is active, local modifications will not be highlighted as errors in the package manager and support data collector.

这个设置默认没有激活。

默认值:

```
$Self->{'Package::AllowLocalModifications'} = '0';
```

### **Package::ShowFeatureAddons**

Toggles display of OTRS FeatureAddons list in PackageManager.

默认值:

```
$Self->{'Package::ShowFeatureAddons'} = '1';
```

### **Package::EventModulePost###99-SupportDataSend**

Package event module file a scheduler task for update registration.

默认值:

```
$Self->{'Package::EventModulePost'}->{'99-SupportDataSend'} = {  
  'Event' => '(PackageInstall|PackageReinstall|PackageUpgrade|PackageUninstall)',  
  'Module' => 'Kernel::System::Package::Event::SupportDataSend',  
  'Transaction' => '1'  
};
```

## Framework → Core::PerformanceLog

### PerformanceLog

Enables performance log (to log the page response time). It will affect the system performance. Frontend::Module###AdminPerformanceLog must be enabled.

默认值:

```
$Self->{'PerformanceLog'} = '0';
```

### PerformanceLog::File

Specifies the path of the file for the performance log.

不能使这个设置失效。

默认值:

```
$Self->{'PerformanceLog::File'} = '<OTRS_CONFIG_Home>/var/log/Performance.log';
```

### PerformanceLog::FileMax

Defines the maximum size (in MB) of the log file.

不能使这个设置失效。

默认值:

```
$Self->{'PerformanceLog::FileMax'} = '25';
```

## Framework → Core::ReferenceData

### ReferenceData::OwnCountryList

This setting allows you to override the built-in country list with your own list of countries. This is particularly handy if you just want to use a small select group of countries.

这个设置默认没有激活。

默认值:

```
$Self->{'ReferenceData::OwnCountryList'} = {  
  'AT' => 'Austria',  
  'CH' => 'Switzerland',  
  'DE' => 'Germany'  
};
```

## Framework → Core::SOAP

### SOAP::User

Defines the username to access the SOAP handle (bin/cgi-bin/rpc.pl).

这个设置默认没有激活。

默认值:

```
$Self->{'SOAP::User'} = 'some_user';
```

### **SOAP::Password**

Defines the password to access the SOAP handle (bin/cgi-bin/rpc.pl).

这个设置默认没有激活。

默认值:

```
$Self->{'SOAP::Password'} = 'some_pass';
```

### **SOAP::Keep-Alive**

Enable keep-alive connection header for SOAP responses.

不能使这个设置失效。

默认值:

```
$Self->{'SOAP::Keep-Alive'} = '0';
```

## **Framework → Core::Sendmail**

### **SendmailModule**

Defines the module to send emails. "Sendmail" directly uses the sendmail binary of your operating system. Any of the "SMTP" mechanisms use a specified (external) mailserver. "DoNotSendEmail" doesn't send emails and it is useful for test systems.

不能使这个设置失效。

默认值:

```
$Self->{'SendmailModule'} = 'Kernel::System::Email::Sendmail';
```

### **SendmailModule::CMD**

If "Sendmail" was selected as SendmailModule, the location of the sendmail binary and the needed options must be specified.

不能使这个设置失效。

默认值:

```
$Self->{'SendmailModule::CMD'} = '/usr/sbin/sendmail -i -f';
```

### **SendmailModule::Host**

If any of the "SMTP" mechanisms was selected as SendmailModule, the mailhost that sends out the mails must be specified.

不能使这个设置失效。

默认值:

```
$Self->{'SendmailModule::Host'} = 'mail.example.com';
```

### **SendmailModule::Port**

If any of the "SMTP" mechanisms was selected as SendmailModule, the port where your mailserver is listening for incoming connections must be specified.

这个设置默认没有激活。

默认值:

```
$Self->{'SendmailModule::Port'} = '25';
```

### **SendmailModule::AuthUser**

If any of the "SMTP" mechanisms was selected as SendmailModule, and authentication to the mail server is needed, an username must be specified.

这个设置默认没有激活。

默认值:

```
$Self->{'SendmailModule::AuthUser'} = 'MailserverLogin';
```

### **SendmailModule::AuthPassword**

If any of the "SMTP" mechanisms was selected as SendmailModule, and authentication to the mail server is needed, a password must be specified.

这个设置默认没有激活。

默认值:

```
$Self->{'SendmailModule::AuthPassword'} = 'MailserverPassword';
```

### **SendmailBcc**

Sends all outgoing email via bcc to the specified address. Please use this only for backup reasons.

默认值:

```
$Self->{'SendmailBcc'} = '';
```

### **SendmailEnvelopeFrom**

If set, this address is used as envelope sender in outgoing messages (not notifications - see below). If no address is specified, the envelope sender is equal to queue e-mail address.

这个设置默认没有激活。

默认值:

```
$Self->{'SendmailEnvelopeFrom'} = '';
```

### **SendmailNotificationEnvelopeFrom**

If set, this address is used as envelope sender header in outgoing notifications. If no address is specified, the envelope sender header is empty (unless SendmailNotificationEnvelopeFrom::FallbackToEmailFrom is set).

这个设置默认没有激活。

默认值:

```
$Self->{'SendmailNotificationEnvelopeFrom'} = '';
```

### **SendmailNotificationEnvelopeFrom::FallbackToEmailFrom**

If no SendmailNotificationEnvelopeFrom is specified, this setting makes it possible to use the email's from address instead of an empty envelope sender (required in certain mail server configurations).

这个设置默认没有激活。

默认值:

```
$Self->{'SendmailNotificationEnvelopeFrom::FallbackToEmailFrom'} = '0';
```

### SendmailEncodingForce

Forces encoding of outgoing emails (7bit|8bit|quoted-printable|base64).

这个设置默认没有激活。

默认值:

```
$Self->{'SendmailEncodingForce'} = 'base64';
```

## Framework → Core::Session

### SessionModule

Defines the module used to store the session data. With "DB" the frontend server can be splitted from the db server. "FS" is faster.

不能使这个设置失效。

默认值:

```
$Self->{'SessionModule'} = 'Kernel::System::AuthSession::DB';
```

### SessionName

Defines the name of the session key. E.g. Session, SessionID or OTRS.

不能使这个设置失效。

默认值:

```
$Self->{'SessionName'} = 'OTRSAgentInterface';
```

### CustomerPanelSessionName

Defines the name of the key for customer sessions.

不能使这个设置失效。

默认值:

```
$Self->{'CustomerPanelSessionName'} = 'OTRSCustomerInterface';
```

### SessionCheckRemoteIP

Turns on the remote ip address check. It should be set to "No" if the application is used, for example, via a proxy farm or a dialup connection, because the remote ip address is mostly different for the requests.

不能使这个设置失效。

默认值:

```
$Self->{'SessionCheckRemoteIP'} = '1';
```

### SessionDeleteIfNotRemoteID

Deletes a session if the session id is used with an invalid remote IP address.



不能使这个设置失效。

默认值:

```
$Self->{'SessionDeleteIfNotRemoteID'} = '1';
```

### SessionMaxTime

Defines the maximal valid time (in seconds) for a session id.

不能使这个设置失效。

默认值:

```
$Self->{'SessionMaxTime'} = '57600';
```

### SessionMaxIdleTime

Sets the inactivity time (in seconds) to pass before a session is killed and a user is logged out.

不能使这个设置失效。

默认值:

```
$Self->{'SessionMaxIdleTime'} = '7200';
```

### SessionActiveTime

DEPRECATED: This config setting will be removed in further versions of OTRS. Sets the time (in seconds) a user is marked as active (minimum active time is 300 seconds).

不能使这个设置失效。

默认值:

```
$Self->{'SessionActiveTime'} = '600';
```

### SessionDeleteIfTimeToOld

Deletes requested sessions if they have timed out.

不能使这个设置失效。

默认值:

```
$Self->{'SessionDeleteIfTimeToOld'} = '1';
```

### SessionUseCookie

Makes the session management use html cookies. If html cookies are disabled or if the client browser disabled html cookies, then the system will work as usual and append the session id to the links.

不能使这个设置失效。

默认值:

```
$Self->{'SessionUseCookie'} = '1';
```

### SessionUseCookieAfterBrowserClose

Stores cookies after the browser has been closed.

不能使这个设置失效。

默认值:

```
$Self->{'SessionUseCookieAfterBrowserClose'} = '0';
```

### SessionCSRFProtection

Protection against CSRF (Cross Site Request Forgery) exploits (for more info see [http://en.wikipedia.org/wiki/Cross-site\\_request\\_forgery](http://en.wikipedia.org/wiki/Cross-site_request_forgery)).

不能使这个设置失效。

默认值:

```
$Self->{'SessionCSRFProtection'} = '1';
```

### AgentSessionLimitPriorWarning

Sets the maximum number of active agents within the timespan defined in SessionMaxIdleTime before a prior warning will be visible for the logged in agents.

这个设置默认没有激活。

默认值:

```
$Self->{'AgentSessionLimitPriorWarning'} = '90';
```

### AgentSessionLimit

Sets the maximum number of active agents within the timespan defined in SessionMaxIdleTime.

默认值:

```
$Self->{'AgentSessionLimit'} = '100';
```

### AgentSessionPerUserLimit

Sets the maximum number of active sessions per agent within the timespan defined in SessionMaxIdleTime.

默认值:

```
$Self->{'AgentSessionPerUserLimit'} = '20';
```

### CustomerSessionLimit

Sets the maximum number of active customers within the timespan defined in SessionMaxIdleTime.

默认值:

```
$Self->{'CustomerSessionLimit'} = '100';
```

### CustomerSessionPerUserLimit

Sets the maximum number of active sessions per customers within the timespan defined in SessionMaxIdleTime.

默认值:

```
$Self->{'CustomerSessionPerUserLimit'} = '20';
```

### SessionDir

If "FS" was selected for SessionModule, a directory where the session data will be stored must be specified.

不能使这个设置失效。

默认值:

```
$Self->{'SessionDir'} = '<OTRS_CONFIG_Home>/var/sessions';
```

## SessionTable

If "DB" was selected for SessionModule, a table in database where session data will be stored must be specified.

不能使这个设置失效。

默认值:

```
$Self->{'SessionTable'} = 'sessions';
```

## Framework → Core::SpellChecker

### SpellChecker

Enables spell checker support.

不能使这个设置失效。

默认值:

```
$Self->{'SpellChecker'} = '0';
```

### SpellCheckerBin

Install ispell or aspell on the system, if you want to use a spell checker. Please specify the path to the aspell or ispell binary on your operating system.

不能使这个设置失效。

默认值:

```
$Self->{'SpellCheckerBin'} = '/usr/bin/ispell';
```

### SpellCheckerDictDefault

Defines the default spell checker dictionary.

不能使这个设置失效。

默认值:

```
$Self->{'SpellCheckerDictDefault'} = 'english';
```

### SpellCheckerIgnore

Defines a default list of words, that are ignored by the spell checker.

不能使这个设置失效。

默认值:

```
$Self->{'SpellCheckerIgnore'} = [  
  'www',  
  'webmail',  
  'https',  
  'http',  
  'html',  
  'rfc'  
];
```

## Framework → Core::Stats

### Stats::StatsHook

Sets the stats hook.

不能使这个设置失效。

默认值:

```
$Self->{'Stats::StatsHook'} = 'Stat#';
```

### Stats::StatsStartNumber

Start number for statistics counting. Every new stat increments this number.

不能使这个设置失效。

默认值:

```
$Self->{'Stats::StatsStartNumber'} = '10000';
```

### Stats::MaxXaxisAttributes

Defines the default maximum number of X-axis attributes for the time scale.

这个设置默认没有激活。

默认值:

```
$Self->{'Stats::MaxXaxisAttributes'} = '1000';
```

## Framework → Core::Time

### TimeInputFormat

Defines the date input format used in forms (option or input fields).

不能使这个设置失效。

默认值:

```
$Self->{'TimeInputFormat'} = 'Option';
```

### TimeShowAlwaysLong

Shows time in long format (days, hours, minutes), if set to "Yes"; or in short format (days, hours), if set to "No".

不能使这个设置失效。

默认值:

```
$Self->{'TimeShowAlwaysLong'} = '0';
```

### TimeZone

Sets the system time zone (required a system with UTC as system time). Otherwise this is a diff time to the local time.

这个设置默认没有激活。

默认值:

```
$Self->{'TimeZone'} = '+0';
```

## TimeZoneUser

Sets the user time zone per user (required a system with UTC as system time and UTC under TimeZone). Otherwise this is a diff time to the local time.

默认值:

```
$Self->{'TimeZoneUser'} = '0';
```

## TimeZoneUserBrowserAutoOffset

Sets the user time zone per user based on java script / browser time zone offset feature at login time.

默认值:

```
$Self->{'TimeZoneUserBrowserAutoOffset'} = '1';
```

## MaximumCalendarNumber

Maximum Number of a calendar shown in a dropdown.

这个设置默认没有激活。

默认值:

```
$Self->{'MaximumCalendarNumber'} = '50';
```

## CalendarWeekDayStart

Define the start day of the week for the date picker.

不能使这个设置失效。

默认值:

```
$Self->{'CalendarWeekDayStart'} = '1';
```

## TimeVacationDays

Adds the permanent vacation days. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).

不能使这个设置失效。

默认值:

```
$Self->{'TimeVacationDays'} = {  
  '1' => {  
    '1' => 'New Year\'s Day'  
  },  
  '12' => {  
    '24' => 'Christmas Eve',  
    '25' => 'First Christmas Day',  
    '26' => 'Second Christmas Day',  
    '31' => 'New Year\'s Eve'  
  },  
  '5' => {  
    '1' => 'International Workers\' Day'  
  }  
};
```

## TimeVacationDaysOneTime

Adds the one time vacation days. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).

不能使这个设置失效。

默认值:

```
$Self->{'TimeVacationDaysOneTime'} = {  
  '2004' => {  
    '1' => {  
      '1' => 'test'  
    }  
  }  
};
```

## TimeWorkingHours

Defines the hours and week days to count the working time.

不能使这个设置失效。

默认值:

```
$Self->{'TimeWorkingHours'} = {  
  'Fri' => [  
    '8',  
    '9',  
    '10',  
    '11',  
    '12',  
    '13',  
    '14',  
    '15',  
    '16',  
    '17',  
    '18',  
    '19',  
    '20'  
  ],  
  'Mon' => [  
    '8',  
    '9',  
    '10',  
    '11',  
    '12',  
    '13',  
    '14',  
    '15',  
    '16',  
    '17',  
    '18',  
    '19',  
    '20'  
  ],  
  'Sat' => [],  
  'Sun' => [],  
  'Thu' => [  
    '8',  
    '9',  
    '10',  
    '11',  
    '12',  
    '13',  
    '14',  
    '15',  
    '16',  
    '17',  
    '18',  
    '19',  
    '20'  
  ],  
  'Tue' => [  
    '8',  
    '9',
```

```
'10',
'11',
'12',
'13',
'14',
'15',
'16',
'17',
'18',
'19',
'20'
],
'Wed' => [
'8',
'9',
'10',
'11',
'12',
'13',
'14',
'15',
'16',
'17',
'18',
'19',
'20'
]
];
```

### TimeShowCompleteDescription

Shows time use complete description (days, hours, minutes), if set to "Yes"; or just first letter (d, h, m), if set to "No".

不能使这个设置失效。

默认值:

```
$Self->{'TimeShowCompleteDescription'} = '0';
```

## Framework → Core::Time::Calendar1

### TimeZone::Calendar1Name

Defines the name of the indicated calendar.

不能使这个设置失效。

默认值:

```
$Self->{'TimeZone::Calendar1Name'} = 'Calendar Name 1';
```

### TimeZone::Calendar1

Defines the time zone of the indicated calendar, which can be assigned later to a specific queue.

不能使这个设置失效。

默认值:

```
$Self->{'TimeZone::Calendar1'} = '0';
```

### CalendarWeekDayStart::Calendar1

Define the start day of the week for the date picker for the indicated calendar.

不能使这个设置失效。

默认值:

```
$Self->{'CalendarWeekDayStart::Calendar1'} = '1';
```

### TimeVacationDays::Calendar1

Adds the permanent vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).

不能使这个设置失效。

默认值:

```
$Self->{'TimeVacationDays::Calendar1'} = {  
  '1' => {  
    '1' => 'New Year\'s Day'  
  },  
  '12' => {  
    '24' => 'Christmas Eve',  
    '25' => 'First Christmas Day',  
    '26' => 'Second Christmas Day',  
    '31' => 'New Year\'s Eve'  
  },  
  '5' => {  
    '1' => 'International Workers\' Day'  
  }  
};
```

### TimeVacationDaysOneTime::Calendar1

Adds the one time vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).

不能使这个设置失效。

默认值:

```
$Self->{'TimeVacationDaysOneTime::Calendar1'} = {  
  '2004' => {  
    '1' => {  
      '1' => 'test'  
    }  
  }  
};
```

### TimeWorkingHours::Calendar1

Defines the hours and week days of the indicated calendar, to count the working time.

不能使这个设置失效。

默认值:

```
$Self->{'TimeWorkingHours::Calendar1'} = {  
  'Fri' => [  
    '8',  
    '9',  
    '10',  
    '11',  
    '12',  
    '13',  
    '14',  
    '15',  
    '16',  
    '17',  
    '18',  
    '19',  
  ]  
};
```



```
'20'  
],  
'Mon' => [  
  '8',  
  '9',  
  '10',  
  '11',  
  '12',  
  '13',  
  '14',  
  '15',  
  '16',  
  '17',  
  '18',  
  '19',  
  '20'  
],  
'Sat' => [],  
'Sun' => [],  
'Thu' => [  
  '8',  
  '9',  
  '10',  
  '11',  
  '12',  
  '13',  
  '14',  
  '15',  
  '16',  
  '17',  
  '18',  
  '19',  
  '20'  
],  
'Tue' => [  
  '8',  
  '9',  
  '10',  
  '11',  
  '12',  
  '13',  
  '14',  
  '15',  
  '16',  
  '17',  
  '18',  
  '19',  
  '20'  
],  
'Wed' => [  
  '8',  
  '9',  
  '10',  
  '11',  
  '12',  
  '13',  
  '14',  
  '15',  
  '16',  
  '17',  
  '18',  
  '19',  
  '20'  
]  
];
```

## Framework → Core::Time::Calendar2

### TimeZone::Calendar2Name

Defines the name of the indicated calendar.

不能使这个设置失效。

默认值:

```
$Self->{'TimeZone::Calendar2Name'} = 'Calendar Name 2';
```

### **TimeZone::Calendar2**

Defines the time zone of the indicated calendar, which can be assigned later to a specific queue.

不能使这个设置失效。

默认值:

```
$Self->{'TimeZone::Calendar2'} = '0';
```

### **CalendarWeekDayStart::Calendar2**

Define the start day of the week for the date picker for the indicated calendar.

不能使这个设置失效。

默认值:

```
$Self->{'CalendarWeekDayStart::Calendar2'} = '1';
```

### **TimeVacationDays::Calendar2**

Adds the permanent vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).

不能使这个设置失效。

默认值:

```
$Self->{'TimeVacationDays::Calendar2'} = {  
  '1' => {  
    '1' => 'New Year\'s Day'  
  },  
  '12' => {  
    '24' => 'Christmas Eve',  
    '25' => 'First Christmas Day',  
    '26' => 'Second Christmas Day',  
    '31' => 'New Year\'s Eve'  
  },  
  '5' => {  
    '1' => 'International Workers\' Day'  
  }  
};
```

### **TimeVacationDaysOneTime::Calendar2**

Adds the one time vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).

不能使这个设置失效。

默认值:

```
$Self->{'TimeVacationDaysOneTime::Calendar2'} = {  
  '2004' => {  
    '1' => {  
      '1' => 'test'  
    }  
  }  
}
```

```
};
```

## TimeWorkingHours::Calendar2

Defines the hours and week days of the indicated calendar, to count the working time.

不能使这个设置失效。

默认值:

```
$Self->{'TimeWorkingHours::Calendar2'} = {  
  'Fri' => [  
    '8',  
    '9',  
    '10',  
    '11',  
    '12',  
    '13',  
    '14',  
    '15',  
    '16',  
    '17',  
    '18',  
    '19',  
    '20'  
  ],  
  'Mon' => [  
    '8',  
    '9',  
    '10',  
    '11',  
    '12',  
    '13',  
    '14',  
    '15',  
    '16',  
    '17',  
    '18',  
    '19',  
    '20'  
  ],  
  'Sat' => [],  
  'Sun' => [],  
  'Thu' => [  
    '8',  
    '9',  
    '10',  
    '11',  
    '12',  
    '13',  
    '14',  
    '15',  
    '16',  
    '17',  
    '18',  
    '19',  
    '20'  
  ],  
  'Tue' => [  
    '8',  
    '9',  
    '10',  
    '11',  
    '12',  
    '13',  
    '14',  
    '15',  
    '16',  
    '17',  
    '18',  
    '19',  
    '20'  
  ]  
}
```

```

],
'Wed' => [
  '8',
  '9',
  '10',
  '11',
  '12',
  '13',
  '14',
  '15',
  '16',
  '17',
  '18',
  '19',
  '20'
]
};

```

## Framework → Core::Time::Calendar3

### TimeZone::Calendar3Name

Defines the name of the indicated calendar.

不能使这个设置失效。

默认值:

```
$Self->{'TimeZone::Calendar3Name'} = 'Calendar Name 3';
```

### TimeZone::Calendar3

Defines the time zone of the indicated calendar, which can be assigned later to a specific queue.

不能使这个设置失效。

默认值:

```
$Self->{'TimeZone::Calendar3'} = '0';
```

### CalendarWeekDayStart::Calendar3

Define the start day of the week for the date picker for the indicated calendar.

不能使这个设置失效。

默认值:

```
$Self->{'CalendarWeekDayStart::Calendar3'} = '1';
```

### TimeVacationDays::Calendar3

Adds the permanent vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).

不能使这个设置失效。

默认值:

```

$Self->{'TimeVacationDays::Calendar3'} = {
  '1' => {
    '1' => 'New Year\'s Day'
  },
  '12' => {
    '24' => 'Christmas Eve',
    '25' => 'First Christmas Day',
  }
};

```

```
'26' => 'Second Christmas Day',
'31' => 'New Year\'s Eve'
},
'5' => {
  '1' => 'International Workers\' Day'
}
};
```

### TimeVacationDaysOneTime::Calendar3

Adds the one time vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).

不能使这个设置失效。

默认值:

```
$Self->{'TimeVacationDaysOneTime::Calendar3'} = {
  '2004' => {
    '1' => {
      '1' => 'test'
    }
  }
};
```

### TimeWorkingHours::Calendar3

Defines the hours and week days of the indicated calendar, to count the working time.

不能使这个设置失效。

默认值:

```
$Self->{'TimeWorkingHours::Calendar3'} = {
  'Fri' => [
    '8',
    '9',
    '10',
    '11',
    '12',
    '13',
    '14',
    '15',
    '16',
    '17',
    '18',
    '19',
    '20'
  ],
  'Mon' => [
    '8',
    '9',
    '10',
    '11',
    '12',
    '13',
    '14',
    '15',
    '16',
    '17',
    '18',
    '19',
    '20'
  ],
  'Sat' => [],
  'Sun' => [],
  'Thu' => [
    '8',
    '9',
    '10',
```

```
'11',
'12',
'13',
'14',
'15',
'16',
'17',
'18',
'19',
'20'
],
'Tue' => [
'8',
'9',
'10',
'11',
'12',
'13',
'14',
'15',
'16',
'17',
'18',
'19',
'20'
],
'Wed' => [
'8',
'9',
'10',
'11',
'12',
'13',
'14',
'15',
'16',
'17',
'18',
'19',
'20'
]
];
```

## Framework → Core::Time::Calendar4

### TimeZone::Calendar4Name

Defines the name of the indicated calendar.

不能使这个设置失效。

默认值:

```
$Self->{'TimeZone::Calendar4Name'} = 'Calendar Name 4';
```

### TimeZone::Calendar4

Defines the time zone of the indicated calendar, which can be assigned later to a specific queue.

不能使这个设置失效。

默认值:

```
$Self->{'TimeZone::Calendar4'} = '0';
```

### CalendarWeekDayStart::Calendar4

Define the start day of the week for the date picker for the indicated calendar.

不能使这个设置失效。

默认值:

```
$Self->{'CalendarWeekDayStart::Calendar4'} = '1';
```

### TimeVacationDays::Calendar4

Adds the permanent vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).

不能使这个设置失效。

默认值:

```
$Self->{'TimeVacationDays::Calendar4'} = {
  '1' => {
    '1' => 'New Year\'s Day'
  },
  '12' => {
    '24' => 'Christmas Eve',
    '25' => 'First Christmas Day',
    '26' => 'Second Christmas Day',
    '31' => 'New Year\'s Eve'
  },
  '5' => {
    '1' => 'International Workers\' Day'
  }
};
```

### TimeVacationDaysOneTime::Calendar4

Adds the one time vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).

不能使这个设置失效。

默认值:

```
$Self->{'TimeVacationDaysOneTime::Calendar4'} = {
  '2004' => {
    '1' => {
      '1' => 'test'
    }
  }
};
```

### TimeWorkingHours::Calendar4

Defines the hours and week days of the indicated calendar, to count the working time.

不能使这个设置失效。

默认值:

```
$Self->{'TimeWorkingHours::Calendar4'} = {
  'Fri' => [
    '8',
    '9',
    '10',
    '11',
    '12',
    '13',
    '14',
    '15',
    '16',
    '17',
    '18',
    '19',
  ]
};
```

```
'20'  
],  
'Mon' => [  
  '8',  
  '9',  
  '10',  
  '11',  
  '12',  
  '13',  
  '14',  
  '15',  
  '16',  
  '17',  
  '18',  
  '19',  
  '20'  
],  
'Sat' => [],  
'Sun' => [],  
'Thu' => [  
  '8',  
  '9',  
  '10',  
  '11',  
  '12',  
  '13',  
  '14',  
  '15',  
  '16',  
  '17',  
  '18',  
  '19',  
  '20'  
],  
'Tue' => [  
  '8',  
  '9',  
  '10',  
  '11',  
  '12',  
  '13',  
  '14',  
  '15',  
  '16',  
  '17',  
  '18',  
  '19',  
  '20'  
],  
'Wed' => [  
  '8',  
  '9',  
  '10',  
  '11',  
  '12',  
  '13',  
  '14',  
  '15',  
  '16',  
  '17',  
  '18',  
  '19',  
  '20'  
]  
];
```

## Framework → Core::Time::Calendar5

### TimeZone::Calendar5Name

Defines the name of the indicated calendar.



不能使这个设置失效。

默认值:

```
$Self->{'TimeZone::Calendar5Name'} = 'Calendar Name 5';
```

### **TimeZone::Calendar5**

Defines the time zone of the indicated calendar, which can be assigned later to a specific queue.

不能使这个设置失效。

默认值:

```
$Self->{'TimeZone::Calendar5'} = '0';
```

### **CalendarWeekDayStart::Calendar5**

Define the start day of the week for the date picker for the indicated calendar.

不能使这个设置失效。

默认值:

```
$Self->{'CalendarWeekDayStart::Calendar5'} = '1';
```

### **TimeVacationDays::Calendar5**

Adds the permanent vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).

不能使这个设置失效。

默认值:

```
$Self->{'TimeVacationDays::Calendar5'} = {  
  '1' => {  
    '1' => 'New Year\'s Day'  
  },  
  '12' => {  
    '24' => 'Christmas Eve',  
    '25' => 'First Christmas Day',  
    '26' => 'Second Christmas Day',  
    '31' => 'New Year\'s Eve'  
  },  
  '5' => {  
    '1' => 'International Workers\' Day'  
  }  
};
```

### **TimeVacationDaysOneTime::Calendar5**

Adds the one time vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).

不能使这个设置失效。

默认值:

```
$Self->{'TimeVacationDaysOneTime::Calendar5'} = {  
  '2004' => {  
    '1' => {  
      '1' => 'test'  
    }  
  }  
};
```

```
};
```

## TimeWorkingHours::Calendar5

Defines the hours and week days of the indicated calendar, to count the working time.

不能使这个设置失效。

默认值:

```
$Self->{'TimeWorkingHours::Calendar5'} = {  
  'Fri' => [  
    '8',  
    '9',  
    '10',  
    '11',  
    '12',  
    '13',  
    '14',  
    '15',  
    '16',  
    '17',  
    '18',  
    '19',  
    '20'  
  ],  
  'Mon' => [  
    '8',  
    '9',  
    '10',  
    '11',  
    '12',  
    '13',  
    '14',  
    '15',  
    '16',  
    '17',  
    '18',  
    '19',  
    '20'  
  ],  
  'Sat' => [],  
  'Sun' => [],  
  'Thu' => [  
    '8',  
    '9',  
    '10',  
    '11',  
    '12',  
    '13',  
    '14',  
    '15',  
    '16',  
    '17',  
    '18',  
    '19',  
    '20'  
  ],  
  'Tue' => [  
    '8',  
    '9',  
    '10',  
    '11',  
    '12',  
    '13',  
    '14',  
    '15',  
    '16',  
    '17',  
    '18',  
    '19',  
    '20'  
  ]  
}
```

```

],
'Wed' => [
  '8',
  '9',
  '10',
  '11',
  '12',
  '13',
  '14',
  '15',
  '16',
  '17',
  '18',
  '19',
  '20'
]
};

```

## Framework → Core::Time::Calendar6

### TimeZone::Calendar6Name

Defines the name of the indicated calendar.

不能使这个设置失效。

默认值:

```
$Self->{'TimeZone::Calendar6Name'} = 'Calendar Name 6';
```

### TimeZone::Calendar6

Defines the time zone of the indicated calendar, which can be assigned later to a specific queue.

不能使这个设置失效。

默认值:

```
$Self->{'TimeZone::Calendar6'} = '0';
```

### CalendarWeekDayStart::Calendar6

Define the start day of the week for the date picker for the indicated calendar.

不能使这个设置失效。

默认值:

```
$Self->{'CalendarWeekDayStart::Calendar6'} = '1';
```

### TimeVacationDays::Calendar6

Adds the permanent vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).

不能使这个设置失效。

默认值:

```

$Self->{'TimeVacationDays::Calendar6'} = {
  '1' => {
    '1' => 'New Year\'s Day'
  },
  '12' => {
    '24' => 'Christmas Eve',
    '25' => 'First Christmas Day',
  }
};

```

```
'26' => 'Second Christmas Day',
'31' => 'New Year\'s Eve'
},
'5' => {
  '1' => 'International Workers\' Day'
}
};
```

### TimeVacationDaysOneTime::Calendar6

Adds the one time vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).

不能使这个设置失效。

默认值:

```
$Self->{'TimeVacationDaysOneTime::Calendar6'} = {
  '2004' => {
    '1' => {
      '1' => 'test'
    }
  }
};
```

### TimeWorkingHours::Calendar6

Defines the hours and week days of the indicated calendar, to count the working time.

不能使这个设置失效。

默认值:

```
$Self->{'TimeWorkingHours::Calendar6'} = {
  'Fri' => [
    '8',
    '9',
    '10',
    '11',
    '12',
    '13',
    '14',
    '15',
    '16',
    '17',
    '18',
    '19',
    '20'
  ],
  'Mon' => [
    '8',
    '9',
    '10',
    '11',
    '12',
    '13',
    '14',
    '15',
    '16',
    '17',
    '18',
    '19',
    '20'
  ],
  'Sat' => [],
  'Sun' => [],
  'Thu' => [
    '8',
    '9',
    '10',
```

```
'11',
'12',
'13',
'14',
'15',
'16',
'17',
'18',
'19',
'20'
],
'Tue' => [
'8',
'9',
'10',
'11',
'12',
'13',
'14',
'15',
'16',
'17',
'18',
'19',
'20'
],
'Wed' => [
'8',
'9',
'10',
'11',
'12',
'13',
'14',
'15',
'16',
'17',
'18',
'19',
'20'
]
];
```

## Framework → Core::Time::Calendar7

### TimeZone::Calendar7Name

Defines the name of the indicated calendar.

不能使这个设置失效。

默认值:

```
$Self->{'TimeZone::Calendar7Name'} = 'Calendar Name 7';
```

### TimeZone::Calendar7

Defines the time zone of the indicated calendar, which can be assigned later to a specific queue.

不能使这个设置失效。

默认值:

```
$Self->{'TimeZone::Calendar7'} = '0';
```

### CalendarWeekDayStart::Calendar7

Define the start day of the week for the date picker for the indicated calendar.

不能使这个设置失效。

默认值:

```
$Self->{'CalendarWeekDayStart::Calendar7'} = '1';
```

### TimeVacationDays::Calendar7

Adds the permanent vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).

不能使这个设置失效。

默认值:

```
$Self->{'TimeVacationDays::Calendar7'} = {  
  '1' => {  
    '1' => 'New Year\'s Day'  
  },  
  '12' => {  
    '24' => 'Christmas Eve',  
    '25' => 'First Christmas Day',  
    '26' => 'Second Christmas Day',  
    '31' => 'New Year\'s Eve'  
  },  
  '5' => {  
    '1' => 'International Workers\' Day'  
  }  
};
```

### TimeVacationDaysOneTime::Calendar7

Adds the one time vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).

不能使这个设置失效。

默认值:

```
$Self->{'TimeVacationDaysOneTime::Calendar7'} = {  
  '2004' => {  
    '1' => {  
      '1' => 'test'  
    }  
  }  
};
```

### TimeWorkingHours::Calendar7

Defines the hours and week days of the indicated calendar, to count the working time.

不能使这个设置失效。

默认值:

```
$Self->{'TimeWorkingHours::Calendar7'} = {  
  'Fri' => [  
    '8',  
    '9',  
    '10',  
    '11',  
    '12',  
    '13',  
    '14',  
    '15',  
    '16',  
    '17',  
    '18',  
    '19',  
  ]  
};
```

```
'20'  
],  
'Mon' => [  
  '8',  
  '9',  
  '10',  
  '11',  
  '12',  
  '13',  
  '14',  
  '15',  
  '16',  
  '17',  
  '18',  
  '19',  
  '20'  
],  
'Sat' => [],  
'Sun' => [],  
'Thu' => [  
  '8',  
  '9',  
  '10',  
  '11',  
  '12',  
  '13',  
  '14',  
  '15',  
  '16',  
  '17',  
  '18',  
  '19',  
  '20'  
],  
'Tue' => [  
  '8',  
  '9',  
  '10',  
  '11',  
  '12',  
  '13',  
  '14',  
  '15',  
  '16',  
  '17',  
  '18',  
  '19',  
  '20'  
],  
'Wed' => [  
  '8',  
  '9',  
  '10',  
  '11',  
  '12',  
  '13',  
  '14',  
  '15',  
  '16',  
  '17',  
  '18',  
  '19',  
  '20'  
]  
];
```

## Framework → Core::Time::Calendar8

### TimeZone::Calendar8Name

Defines the name of the indicated calendar.

不能使这个设置失效。

默认值:

```
$Self->{'TimeZone::Calendar8Name'} = 'Calendar Name 8';
```

### **TimeZone::Calendar8**

Defines the time zone of the indicated calendar, which can be assigned later to a specific queue.

不能使这个设置失效。

默认值:

```
$Self->{'TimeZone::Calendar8'} = '0';
```

### **CalendarWeekDayStart::Calendar8**

Define the start day of the week for the date picker for the indicated calendar.

不能使这个设置失效。

默认值:

```
$Self->{'CalendarWeekDayStart::Calendar8'} = '1';
```

### **TimeVacationDays::Calendar8**

Adds the permanent vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).

不能使这个设置失效。

默认值:

```
$Self->{'TimeVacationDays::Calendar8'} = {  
  '1' => {  
    '1' => 'New Year\'s Day'  
  },  
  '12' => {  
    '24' => 'Christmas Eve',  
    '25' => 'First Christmas Day',  
    '26' => 'Second Christmas Day',  
    '31' => 'New Year\'s Eve'  
  },  
  '5' => {  
    '1' => 'International Workers\' Day'  
  }  
};
```

### **TimeVacationDaysOneTime::Calendar8**

Adds the one time vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).

不能使这个设置失效。

默认值:

```
$Self->{'TimeVacationDaysOneTime::Calendar8'} = {  
  '2004' => {  
    '1' => {  
      '1' => 'test'  
    }  
  }  
};
```





```
'8',
'9',
'10',
'11',
'12',
'13',
'14',
'15',
'16',
'17',
'18',
'19',
'20'
]
};
```

## Framework → Core::Time::Calendar9

### TimeZone::Calendar9Name

Defines the name of the indicated calendar.

不能使这个设置失效。

默认值:

```
$Self->{'TimeZone::Calendar9Name'} = 'Calendar Name 9';
```

### TimeZone::Calendar9

Defines the time zone of the indicated calendar, which can be assigned later to a specific queue.

不能使这个设置失效。

默认值:

```
$Self->{'TimeZone::Calendar9'} = '0';
```

### CalendarWeekDayStart::Calendar9

Define the start day of the week for the date picker for the indicated calendar.

不能使这个设置失效。

默认值:

```
$Self->{'CalendarWeekDayStart::Calendar9'} = '1';
```

### TimeVacationDays::Calendar9

Adds the permanent vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).

不能使这个设置失效。

默认值:

```
$Self->{'TimeVacationDays::Calendar9'} = {
  '1' => {
    '1' => 'New Year\'s Day'
  },
  '12' => {
    '24' => 'Christmas Eve',
    '25' => 'First Christmas Day',
    '26' => 'Second Christmas Day',
    '31' => 'New Year\'s Eve'
  }
}
```

```

},
'5' => {
  '1' => 'International Workers\' Day'
}
};

```

### TimeVacationDaysOneTime::Calendar9

Adds the one time vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).

不能使这个设置失效。

默认值:

```

$self->{'TimeVacationDaysOneTime::Calendar9'} = {
  '2004' => {
    '1' => {
      '1' => 'test'
    }
  }
};

```

### TimeWorkingHours::Calendar9

Defines the hours and week days of the indicated calendar, to count the working time.

不能使这个设置失效。

默认值:

```

$self->{'TimeWorkingHours::Calendar9'} = {
  'Fri' => [
    '8',
    '9',
    '10',
    '11',
    '12',
    '13',
    '14',
    '15',
    '16',
    '17',
    '18',
    '19',
    '20'
  ],
  'Mon' => [
    '8',
    '9',
    '10',
    '11',
    '12',
    '13',
    '14',
    '15',
    '16',
    '17',
    '18',
    '19',
    '20'
  ],
  'Sat' => [],
  'Sun' => [],
  'Thu' => [
    '8',
    '9',
    '10',
    '11',
    '12',

```

```

    '13',
    '14',
    '15',
    '16',
    '17',
    '18',
    '19',
    '20'
  ],
  'Tue' => [
    '8',
    '9',
    '10',
    '11',
    '12',
    '13',
    '14',
    '15',
    '16',
    '17',
    '18',
    '19',
    '20'
  ],
  'Wed' => [
    '8',
    '9',
    '10',
    '11',
    '12',
    '13',
    '14',
    '15',
    '16',
    '17',
    '18',
    '19',
    '20'
  ]
]
};

```

## Framework → Core::Web

### Frontend::WebPath

Defines the URL base path of icons, CSS and Java Script.

不能使这个设置失效。

默认值:

```
$Self->{'Frontend::WebPath'} = '/otrs-web/';
```

### Frontend::ImagePath

Defines the URL image path of icons for navigation.

不能使这个设置失效。

默认值:

```
$Self->{'Frontend::ImagePath'} = '<OTRS_CONFIG_Frontend::WebPath>skins/Agent/default/img/';
```

### Frontend::CSSPath

Defines the URL CSS path.

不能使这个设置失效。

默认值:

```
$Self->{'Frontend::CSSPath'} = '<OTRS_CONFIG_Frontend::WebPath>css/';
```

### **Frontend::JavaScriptPath**

Defines the URL java script path.

不能使这个设置失效。

默认值:

```
$Self->{'Frontend::JavaScriptPath'} = '<OTRS_CONFIG_Frontend::WebPath>js/';
```

### **Frontend::RichText**

Uses richtext for viewing and editing: articles, salutations, signatures, standard templates, auto responses and notifications.

不能使这个设置失效。

默认值:

```
$Self->{'Frontend::RichText'} = '1';
```

### **Frontend::RichTextPath**

Defines the URL rich text editor path.

不能使这个设置失效。

默认值:

```
$Self->{'Frontend::RichTextPath'} = '<OTRS_CONFIG_Frontend::WebPath>js/thirdparty/  
ckeditor-4.5.6/';
```

### **Frontend::RichTextWidth**

Defines the width for the rich text editor component. Enter number (pixels) or percent value (relative).

不能使这个设置失效。

默认值:

```
$Self->{'Frontend::RichTextWidth'} = '620';
```

### **Frontend::RichTextHeight**

Defines the height for the rich text editor component. Enter number (pixels) or percent value (relative).

不能使这个设置失效。

默认值:

```
$Self->{'Frontend::RichTextHeight'} = '320';
```

### **Frontend::RichText::DefaultCSS**

Defines the default CSS used in rich text editors.

不能使这个设置失效。

默认值:

```
$Self->{'Frontend::RichText::DefaultCSS'} = 'font-family:Geneva,Helvetica,Arial,sans-serif; font-size: 12px;';
```

### **Frontend::RichText::EnhancedMode**

Defines if the enhanced mode should be used (enables use of table, replace, subscript, superscript, paste from word, etc.).

不能使这个设置失效。

默认值:

```
$Self->{'Frontend::RichText::EnhancedMode'} = '0';
```

### **Frontend::RichText::EnhancedMode::Customer**

Defines if the enhanced mode should be used (enables use of table, replace, subscript, superscript, paste from word, etc.) in customer interface.

不能使这个设置失效。

默认值:

```
$Self->{'Frontend::RichText::EnhancedMode::Customer'} = '0';
```

### **DisableMSIFrameSecurityRestricted**

Disable restricted security for IFrames in IE. May be required for SSO to work in IE.

默认值:

```
$Self->{'DisableMSIFrameSecurityRestricted'} = '0';
```

### **DisableLoginAutocomplete**

Disable autocomplete in the login screen.

默认值:

```
$Self->{'DisableLoginAutocomplete'} = '0';
```

### **DisableIFrameOriginRestricted**

Disable HTTP header "X-Frame-Options: SAMEORIGIN" to allow OTRS to be included as an IFrame in other websites. Disabling this HTTP header can be a security issue! Only disable it, if you know what you are doing!

默认值:

```
$Self->{'DisableIFrameOriginRestricted'} = '0';
```

### **DisableContentSecurityPolicy**

Disable HTTP header "Content-Security-Policy" to allow loading of external script contents. Disabling this HTTP header can be a security issue! Only disable it, if you know what you are doing!

默认值:

```
$Self->{'DisableContentSecurityPolicy'} = '0';
```

### **DefaultViewNewLine**

Automated line break in text messages after x number of chars.

不能使这个设置失效。

默认值:

```
$Self->{'DefaultViewNewLine'} = '90';
```

### DefaultViewLines

Sets the number of lines that are displayed in text messages (e.g. ticket lines in the QueueZoom).

不能使这个设置失效。

默认值:

```
$Self->{'DefaultViewLines'} = '6000';
```

### Frontend::AnimationEnabled

Turns on the animations used in the GUI. If you have problems with these animations (e.g. performance issues), you can turn them off here.

不能使这个设置失效。

默认值:

```
$Self->{'Frontend::AnimationEnabled'} = '1';
```

### Frontend::MenuDragDropEnabled

Turns on drag and drop for the main navigation.

不能使这个设置失效。

默认值:

```
$Self->{'Frontend::MenuDragDropEnabled'} = '1';
```

### AttachmentDownloadType

Allows choosing between showing the attachments of a ticket in the browser (inline) or just make them downloadable (attachment).

不能使这个设置失效。

默认值:

```
$Self->{'AttachmentDownloadType'} = 'attachment';
```

### WebMaxFileUpload

Defines the maximal size (in bytes) for file uploads via the browser. Warning: Setting this option to a value which is too low could cause many masks in your OTRS instance to stop working (probably any mask which takes input from the user).

不能使这个设置失效。

默认值:

```
$Self->{'WebMaxFileUpload'} = '24000000';
```

### WebUploadCacheModule

Selects the module to handle uploads via the web interface. "DB" stores all uploads in the database, "FS" uses the file system.

不能使这个设置失效。

默认值:

```
$Self->{'WebUploadCacheModule'} = 'Kernel::System::Web::UploadCache::DB';
```

### Frontend::Output::FilterText###AAAURL

Defines the filter that processes the text in the articles, in order to highlight URLs.

默认值:

```
$Self->{'Frontend::Output::FilterText'}->{'AAAURL'} = {
  'Module' => 'Kernel::Output::HTML::FilterText::URL',
  'Templates' => {
    'AgentTicketZoom' => '1'
  }
};
```

### Frontend::Themes

Activates the available themes on the system. Value 1 means active, 0 means inactive.

默认值:

```
$Self->{'Frontend::Themes'} = {
  'Lite' => '0',
  'Standard' => '1'
};
```

### Frontend::Output::FilterText###OutputFilterTextAutoLink

Defines a filter to process the text in the articles, in order to highlight predefined keywords.

这个设置默认没有激活。

默认值:

```
$Self->{'Frontend::Output::FilterText'}->{'OutputFilterTextAutoLink'} = {
  'Module' => 'Kernel::Output::HTML::FilterText::AutoLink',
  'Templates' => {
    'AgentTicketZoom' => '1'
  }
};
```

### Frontend::Output::OutputFilterTextAutoLink###CVE

Defines a filter for html output to add links behind CVE numbers. The element Image allows two input kinds. At once the name of an image (e.g. faq.png). In this case the OTRS image path will be used. The second possibility is to insert the link to the image.

这个设置默认没有激活。

默认值:

```
$Self->{'Frontend::Output::OutputFilterTextAutoLink'}->{'CVE'} = {
  'RegExp' => [
    '(CVE|CAN)\-(\d{3,4})\-(\d{2,})'
  ],
  'Templates' => {
    'AgentTicketZoom' => '1'
  },
  'URL1' => {
    'Description' => 'Mitre',
    'Image' => 'http://cve.mitre.org/favicon.ico',
    'Target' => '_blank',
    'URL' => 'http://cve.mitre.org/cgi-bin/cvename.cgi?name=<MATCH1>-<MATCH2>-<MATCH3>'
  }
};
```



```

},
'URL2' => {
  'Description' => 'Google',
  'Image' => 'http://www.google.de/favicon.ico',
  'Target' => '_blank',
  'URL' => 'http://google.com/search?q=<MATCH1>-<MATCH2>-<MATCH3>'
},
'URL3' => {
  'Description' => 'US-CERT NVD',
  'Image' => 'http://nvd.nist.gov/favicon.ico',
  'Target' => '_blank',
  'URL' => 'http://nvd.nist.gov/nvd.cfm?cvename=<MATCH1>-<MATCH2>-<MATCH3>'
}
};

```

### Frontend::Output::OutputFilterTextAutoLink###Bugtraq

Defines a filter for html output to add links behind bugtraq numbers. The element Image allows two input kinds. At once the name of an image (e.g. faq.png). In this case the OTRS image path will be used. The second possibility is to insert the link to the image.

这个设置默认没有激活。

默认值:

```

$self->{'Frontend::Output::OutputFilterTextAutoLink'}->{'Bugtraq'} = {
  'RegExp' => [
    'Bugtraq[\\s\\w\\t]*?ID[\\s\\w\\t]*?:[\\s\\w\\t]*?(\\d{2,8})',
    'Bugtraq[\\s\\w\\t]*?ID[\\s\\w\\t]*?(\\d{2,8})',
    'Bugtraq[\\s\\w\\t]*?:[\\s\\w\\t]*?(\\d{2,8})',
    'Bugtraq[\\s\\w\\t]*?(\\d{2,8})',
    'BID[\\s\\w\\t]*?:[\\s\\w\\t]*?(\\d{2,8})',
    'BID[\\s\\w\\t]*?(\\d{2,8})'
  ],
  'Templates' => {
    'AgentTicketZoom' => '1'
  },
  'URL1' => {
    'Description' => 'Security Focus',
    'Image' => 'http://www.securityfocus.com/favicon.ico',
    'Target' => '_blank',
    'URL' => 'http://www.securityfocus.com/bid/<MATCH1>/info'
  },
  'URL2' => {
    'Description' => 'Google',
    'Image' => 'http://www.google.de/favicon.ico',
    'Target' => '_blank',
    'URL' => 'http://google.com/search?q=<MATCH>'
  }
};

```

### Frontend::Output::OutputFilterTextAutoLink###MSBulletins

Defines a filter for html output to add links behind MSBulletin numbers. The element Image allows two input kinds. At once the name of an image (e.g. faq.png). In this case the OTRS image path will be used. The second possibility is to insert the link to the image.

这个设置默认没有激活。

默认值:

```

$self->{'Frontend::Output::OutputFilterTextAutoLink'}->{'MSBulletins'} = {
  'RegExp' => [
    'MS[^A-Za-z]{0,5}(\\d\\d)?.?(\\d{2,4})'
  ],
  'Templates' => {
    'AgentTicketZoom' => '1'
  }
};

```

```

},
'URL1' => {
  'Description' => 'Microsoft Technet',
  'Image' => 'http://www.microsoft.com/favicon.ico',
  'Target' => '_blank',
  'URL' => 'http://www.microsoft.com/technet/security/bulletin/MS<MATCH1>-
<MATCH2>.mspx'
},
'URL2' => {
  'Description' => 'Google',
  'Image' => 'http://www.google.de/favicon.ico',
  'Target' => '_blank',
  'URL' => 'http://google.com/search?q=MS<MATCH1>-<MATCH2>'
}
};

```

### Frontend::Output::OutputFilterTextAutoLink###Setting1

Define a filter for html output to add links behind a defined string. The element Image allows two input kinds. At once the name of an image (e.g. faq.png). In this case the OTRS image path will be used. The second possibility is to insert the link to the image.

这个设置默认没有激活。

默认值:

```

$self->{'Frontend::Output::OutputFilterTextAutoLink'}->{'Setting1'} = {
  'RegExp' => [
    'RegExp'
  ],
  'Templates' => {
    'AgentTicketZoom' => '1'
  },
  'URL1' => {
    'Description' => 'Description',
    'Image' => 'right-small.png',
    'Target' => '_blank',
    'URL' => 'URL'
  },
  'URL2' => {
    'Description' => 'Description',
    'Image' => 'Image',
    'Target' => '_blank',
    'URL' => 'URL'
  }
};

```

### Frontend::Output::OutputFilterTextAutoLink###Setting2

Defines a filter for html output to add links behind a defined string. The element Image allows two input kinds. At once the name of an image (e.g. faq.png). In this case the OTRS image path will be used. The second possibility is to insert the link to the image.

这个设置默认没有激活。

默认值:

```

$self->{'Frontend::Output::OutputFilterTextAutoLink'}->{'Setting2'} = {
  'RegExp' => [
    'RegExp'
  ],
  'Templates' => {
    'AgentTicketZoom' => '1'
  },
  'URL1' => {
    'Description' => 'Description',
    'Image' => 'right-small.png',
    'Target' => '_blank',
    'URL' => 'URL'
  }
};

```

```

},
'URL2' => {
  'Description' => 'Description',
  'Image' => 'Image',
  'Target' => '_blank',
  'URL' => 'URL'
},
'URL3' => {
  'Description' => 'Description',
  'Image' => 'Image',
  'Target' => '_blank',
  'URL' => 'URL'
}
};

```

### Loader::Enabled::CSS

If enabled, OTRS will deliver all CSS files in minified form. WARNING: If you turn this off, there will likely be problems in IE 7, because it cannot load more than 32 CSS files.

不能使这个设置失效。

默认值:

```
$Self->{'Loader::Enabled::CSS'} = '1';
```

### Loader::Enabled::JS

If enabled, OTRS will deliver all JavaScript files in minified form.

不能使这个设置失效。

默认值:

```
$Self->{'Loader::Enabled::JS'} = '1';
```

### Loader::Agent::CommonCSS###000-Framework

List of CSS files to always be loaded for the agent interface.

不能使这个设置失效。

默认值:

```

$Self->{'Loader::Agent::CommonCSS'}->{'000-Framework'} = [
  'Core.Reset.css',
  'Core.Default.css',
  'Core.Header.css',
  'Core.OverviewControl.css',
  'Core.OverviewSmall.css',
  'Core.OverviewMedium.css',
  'Core.OverviewLarge.css',
  'Core.Footer.css',
  'Core.PageLayout.css',
  'Core.Form.css',
  'Core.Table.css',
  'Core.Widget.css',
  'Core.WidgetMenu.css',
  'Core.TicketDetail.css',
  'Core.Tooltip.css',
  'Core.Dialog.css',
  'Core.InputFields.css',
  'Core.Print.css',
  'thirdparty/fontawesome/font-awesome.css'
];

```

### Loader::Agent::ResponsiveCSS###000-Framework

List of responsive CSS files to always be loaded for the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Loader::Agent::ResponsiveCSS'}->{'000-Framework'} = [  
  'Core.Responsive.css'  
];
```

### Loader::Agent::CommonJS###000-Framework

List of JS files to always be loaded for the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Loader::Agent::CommonJS'}->{'000-Framework'} = [  
  'thirdparty/jquery-2.1.4/jquery.js',  
  'thirdparty/jquery-browser-detection/jquery-browser-detection.js',  
  'thirdparty/jquery-ui-1.11.4/jquery-ui.js',  
  'thirdparty/jquery-ui-touch-punch-0.2.3/jquery.ui.touch-punch.js',  
  'thirdparty/jquery-validate-1.14.0/jquery.validate.js',  
  'thirdparty/stacktrace-0.6.4/stacktrace.js',  
  'thirdparty/jquery-pubsub/pubsub.js',  
  'thirdparty/jquery-jstree-3.1.1/jquery.jstree.js',  
  'Core.JavaScriptEnhancements.js',  
  'Core.Debug.js',  
  'Core.Exception.js',  
  'Core.Data.js',  
  'Core.Config.js',  
  'Core.JSON.js',  
  'Core.App.js',  
  'Core.App.Responsive.js',  
  'Core.AJAX.js',  
  'Core.UI.js',  
  'Core.UI.InputFields.js',  
  'Core.UI.Accordion.js',  
  'Core.UI.Datepicker.js',  
  'Core.UI.DnD.js',  
  'Core.UI.Floater.js',  
  'Core.UI.Resizable.js',  
  'Core.UI.Table.js',  
  'Core.UI.Accessibility.js',  
  'Core.UI.RichTextEditor.js',  
  'Core.UI.Dialog.js',  
  'Core.UI.ActionRow.js',  
  'Core.UI.Popup.js',  
  'Core.UI.TreeSelection.js',  
  'Core.UI.Autocomplete.js',  
  'Core.Form.js',  
  'Core.Form.ErrorTooltips.js',  
  'Core.Form.Validate.js',  
  'Core.Agent.js',  
  'Core.Agent.Search.js',  
  'Core.Agent.CustomerInformationCenterSearch.js',  
  'Core.UI.Notification.js',  
  'Core.Agent.Responsive.js'  
];
```

### Loader::Agent::CommonJS###001-JQueryMigrate

List of JS files to always be loaded for the agent interface.

这个设置默认没有激活。

默认值:

```
$Self->{'Loader::Agent::CommonJS'}->{'001-JQueryMigrate'} = [  
  'thirdparty/jquery-migrate-1.2.1/jquery-migrate.js'  
];
```

### Loader::Customer::CommonCSS###000-Framework

List of CSS files to always be loaded for the customer interface.

不能使这个设置失效。

默认值:

```
$Self->{'Loader::Customer::CommonCSS'}->{'000-Framework'} = [  
  'Core.Reset.css',  
  'Core.Default.css',  
  'Core.Form.css',  
  'Core.Dialog.css',  
  'Core.Tooltip.css',  
  'Core.Login.css',  
  'Core.Control.css',  
  'Core.Table.css',  
  'Core.TicketZoom.css',  
  'Core.InputFields.css',  
  'Core.Print.css',  
  'thirdparty/fontawesome/font-awesome.css'  
];
```

### Loader::Customer::ResponsiveCSS###000-Framework

List of responsive CSS files to always be loaded for the customer interface.

不能使这个设置失效。

默认值:

```
$Self->{'Loader::Customer::ResponsiveCSS'}->{'000-Framework'} = [  
  'Core.Responsive.css'  
];
```

### Loader::Customer::CommonJS###000-Framework

List of JS files to always be loaded for the customer interface.

不能使这个设置失效。

默认值:

```
$Self->{'Loader::Customer::CommonJS'}->{'000-Framework'} = [  
  'thirdparty/jquery-2.1.4/jquery.js',  
  'thirdparty/jquery-browser-detection/jquery-browser-detection.js',  
  'thirdparty/jquery-validate-1.14.0/jquery.validate.js',  
  'thirdparty/jquery-ui-1.11.4/jquery-ui.js',  
  'thirdparty/stacktrace-0.6.4/stacktrace.js',  
  'thirdparty/jquery-pubsub/pubsub.js',  
  'thirdparty/jquery-jstree-3.1.1/jquery.jstree.js',  
  'Core.Debug.js',  
  'Core.Exception.js',  
  'Core.Data.js',  
  'Core.JSON.js',  
  'Core.JavaScriptEnhancements.js',  
  'Core.Config.js',  
  'Core.App.js',  
  'Core.App.Responsive.js',  
  'Core.AJAX.js',  
  'Core.UI.js',  
  'Core.UI.InputFields.js',  
  'Core.UI.Accessibility.js',  
  'Core.UI.Dialog.js',  
  'Core.UI.Floater.js',  
  'Core.UI.RichTextEditor.js',  
  'Core.UI.Datepicker.js',  
  'Core.UI.Popup.js',  
  'Core.UI.TreeSelection.js',  
  'Core.UI.Autocomplete.js',  
];
```

```
'Core.Form.js',  
'Core.Form.ErrorTooltips.js',  
'Core.Form.Validate.js',  
'Core.Customer.js',  
'Core.Customer.Responsive.js'  
];
```

### Loader::Customer::CommonJS###001-JQueryMigrate

List of JS files to always be loaded for the agent interface.

这个设置默认没有激活。

默认值:

```
$Self->{'Loader::Customer::CommonJS'}->{'001-JQueryMigrate'} = [  
  'thirdparty/jquery-migrate-1.2.1/jquery-migrate.js'  
];
```

## Framework → Core::WebUserAgent

### WebUserAgent::Timeout

Sets the timeout (in seconds) for http/ftp downloads.

不能使这个设置失效。

默认值:

```
$Self->{'WebUserAgent::Timeout'} = '15';
```

### WebUserAgent::Proxy

Defines the connections for http/ftp, via a proxy.

这个设置默认没有激活。

默认值:

```
$Self->{'WebUserAgent::Proxy'} = 'http://proxy.sn.no:8001/';
```

### WebUserAgent::DisableSSLVerification

Turns off SSL certificate validation, for example if you use a transparent HTTPS proxy. Use at your own risk!

不能使这个设置失效。

默认值:

```
$Self->{'WebUserAgent::DisableSSLVerification'} = '0';
```

## Framework → Crypt::PGP

### PGP

Enables PGP support. When PGP support is enabled for signing and encrypting mail, it is HIGHLY recommended that the web server runs as the OTRS user. Otherwise, there will be problems with the privileges when accessing .gnupg folder.

不能使这个设置失效。

默认值:

```
$Self->{'PGP'} = '0';
```

## PGP::Bin

Defines the path to PGP binary.

不能使这个设置失效。

默认值:

```
$Self->{'PGP::Bin'} = '/usr/bin/gpg';
```

## PGP::Options

Sets the options for PGP binary.

不能使这个设置失效。

默认值:

```
$Self->{'PGP::Options'} = '--homedir /opt/otrs/.gnupg/ --batch --no-tty --yes';
```

## PGP::Key::Password

Sets the password for private PGP key.

不能使这个设置失效。

默认值:

```
$Self->{'PGP::Key::Password'} = {  
  '488A0B8F' => 'SomePassword',  
  'D2DF79FA' => 'SomePassword'  
};
```

## PGP::TrustedNetwork

Set this to yes if you trust in all your public and private ppg keys, even if they are not certified with a trusted signature.

默认值:

```
$Self->{'PGP::TrustedNetwork'} = '0';
```

## PGP::Log

Configure your own log text for PGP.

不能使这个设置失效。

默认值:

```
$Self->{'PGP::Log'} = {  
  'BADSIG' => 'The PGP signature with the keyid has not been verified successfully.',  
  'ERRSIG' => 'It was not possible to check the PGP signature, this may be caused by a  
missing public key or an unsupported algorithm.',  
  'EXPKEYSIG' => 'The PGP signature was made by an expired key.',  
  'GOODSIG' => 'Good PGP signature.',  
  'KEYREVOKED' => 'The PGP signature was made by a revoked key, this could mean that the  
signature is forged.',  
  'NODATA' => 'No valid OpenPGP data found.',  
  'NO_PUBKEY' => 'No public key found.',  
  'REVKEYSIG' => 'The PGP signature was made by a revoked key, this could mean that the  
signature is forged.',  
  'SIGEXPIRED' => 'The PGP signature is expired.',  
  'SIG_ID' => 'Signature data.',  
  'TRUST_UNDEFINED' => 'This key is not certified with a trusted signature!.',  
  'VALIDSIG' => 'The PGP signature with the keyid is good.'  
};
```

## PGP::StoreDecryptedData

If this option is disabled, articles will not automatically be decrypted and stored in the database. Please note that this also means no decryption will take place and the articles will be shown in ticket zoom in their original (encrypted) form.

不能使这个设置失效。

默认值:

```
$Self->{'PGP::StoreDecryptedData'} = '1';
```

## Framework → Crypt::SMIME

### SMIME

Enables S/MIME support.

不能使这个设置失效。

默认值:

```
$Self->{'SMIME'} = '0';
```

### SMIME::Bin

Defines the path to open ssl binary. It may need a HOME env (`$ENV{HOME} = '/var/lib/wwwrun';`).

不能使这个设置失效。

默认值:

```
$Self->{'SMIME::Bin'} = '/usr/bin/openssl';
```

### SMIME::CertPath

Specifies the directory where SSL certificates are stored.

不能使这个设置失效。

默认值:

```
$Self->{'SMIME::CertPath'} = '/etc/ssl/certs';
```

### SMIME::PrivatePath

Specifies the directory where private SSL certificates are stored.

不能使这个设置失效。

默认值:

```
$Self->{'SMIME::PrivatePath'} = '/etc/ssl/private';
```

### SMIME::CacheTTL

Cache time in seconds for the SSL certificate attributes.

不能使这个设置失效。

默认值:

```
$Self->{'SMIME::CacheTTL'} = '86400';
```



## SMIME::StoreDecryptedData

If this option is disabled, articles will not automatically be decrypted and stored in the database. Please note that this also means no decryption will take place and the articles will be shown in ticket zoom in their original (encrypted) form.

不能使这个设置失效。

默认值:

```
$Self->{'SMIME::StoreDecryptedData'} = '1';
```

## SMIME::FetchFromCustomer

Enables fetch S/MIME from CustomerUser backend support.

不能使这个设置失效。

默认值:

```
$Self->{'SMIME::FetchFromCustomer'} = '0';
```

## Framework → CustomerInformationCenter

### AgentCustomerInformationCenter::MainMenu###010-EditCustomerID

Main menu registration.

这个设置默认没有激活。

默认值:

```
$Self->{'AgentCustomerInformationCenter::MainMenu'}->{'010-EditCustomerID'} = {
  'Link' => "[% Env('Baselink')
%]Action=AdminCustomerCompany;Subaction=Change;CustomerID=[% Data.CustomerID | uri
%];Nav=0",
  'Name' => 'Edit customer company'
};
```

## Framework → Frontend::Admin

### Events###Package

List of all Package events to be displayed in the GUI.

不能使这个设置失效。

默认值:

```
$Self->{'Events'}->{'Package'} = [
  'PackageInstall',
  'PackageReinstall',
  'PackageUpgrade',
  'PackageUninstall'
];
```

### Events###DynamicField

List of all DynamicField events to be displayed in the GUI.

不能使这个设置失效。

默认值:

```
$Self->{'Events'}->{'DynamicField'} = [
  'DynamicFieldAdd',
```

```
'DynamicFieldUpdate',  
'DynamicFieldDelete'  
];
```

### Events###CustomerUser

List of all CustomerUser events to be displayed in the GUI.

不能使这个设置失效。

默认值:

```
$Self->{'Events'}->{'CustomerUser'} = [  
'CustomerUserAdd',  
'CustomerUserUpdate'  
];
```

### Events###CustomerCompany

List of all CustomerCompany events to be displayed in the GUI.

不能使这个设置失效。

默认值:

```
$Self->{'Events'}->{'CustomerCompany'} = [  
'CustomerCompanyAdd',  
'CustomerCompanyUpdate'  
];
```

## Framework → Frontend::Admin::AdminCustomerCompany

### AdminCustomerCompany::RunInitialWildcardSearch

Runs an initial wildcard search of the existing customer company when accessing the AdminCustomerCompany module.

不能使这个设置失效。

默认值:

```
$Self->{'AdminCustomerCompany::RunInitialWildcardSearch'} = '1';
```

## Framework → Frontend::Admin::AdminCustomerUser

### AdminCustomerUser::RunInitialWildcardSearch

Runs an initial wildcard search of the existing customer users when accessing the AdminCustomerUser module.

不能使这个设置失效。

默认值:

```
$Self->{'AdminCustomerUser::RunInitialWildcardSearch'} = '1';
```

## Framework → Frontend::Admin::AdminSelectBox

### AdminSelectBox::AllowDatabaseModification

Controls if the admin is allowed to make changes to the database via AdminSelectBox.

不能使这个设置失效。

默认值:

```
$Self->{'AdminSelectBox::AllowDatabaseModification'} = '0';
```

## Framework → Frontend::Admin::ModuleRegistration

### Frontend::Module###Admin

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'Admin'} = {
  'Description' => 'Admin Area.',
  'Group' => [
    'admin'
  ],
  'Loader' => {
    'CSS' => [
      'Core.Agent.Admin.css'
    ],
    'JavaScript' => [
      'Core.Agent.Admin.SysConfig.js'
    ]
  },
  'NavBar' => [
    {
      'AccessKey' => 'a',
      'Block' => 'ItemArea',
      'Description' => '',
      'Link' => 'Action=Admin',
      'LinkOption' => '',
      'Name' => 'Admin',
      'NavBar' => 'Admin',
      'Prio' => '10000',
      'Type' => 'Menu'
    }
  ],
  'NavBarModule' => {
    'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin'
  },
  'NavBarName' => 'Admin',
  'Title' => ''
};
```

### Frontend::Module###AdminInit

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminInit'} = {
  'Description' => 'Admin',
  'Group' => [
    'admin'
  ],
  'NavBarName' => '',
  'Title' => 'Init'
};
```

### Frontend::Module###AdminUser

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminUser'} = {
  'Description' => 'Create and manage agents.',
  'Group' => [
    'admin'
  ],
};
```

```
'NavBarModule' => {
  'Block' => 'Agent',
  'Description' => 'Create and manage agents.',
  'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
  'Name' => 'Agents',
  'Prio' => '100'
},
'NavBarName' => 'Admin',
'Title' => 'Agents'
};
```

### Frontend::Module###AdminGroup

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminGroup'} = {
  'Description' => 'Admin',
  'Group' => [
    'admin'
  ],
  'NavBarModule' => {
    'Block' => 'Agent',
    'Description' => 'Create and manage groups.',
    'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
    'Name' => 'Groups',
    'Prio' => '150'
  },
  'NavBarName' => 'Admin',
  'Title' => 'Groups'
};
```

### Frontend::Module###AdminUserGroup

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminUserGroup'} = {
  'Description' => 'Admin',
  'Group' => [
    'admin'
  ],
  'NavBarModule' => {
    'Block' => 'Agent',
    'Description' => 'Link agents to groups.',
    'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
    'Name' => 'Agents <-> Groups',
    'Prio' => '200'
  },
  'NavBarName' => 'Admin',
  'Title' => 'Agents <-> Groups'
};
```

### Frontend::Module###AdminCustomerUser

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminCustomerUser'} = {
  'Description' => 'Edit Customer Users.',
  'Group' => [
    'admin',
    'users'
  ],
  'GroupRo' => [
    ''
  ],
};
```

```
'Loader' => {
  'JavaScript' => [
    'Core.Agent.TicketAction.js'
  ]
},
'NavBar' => [
  {
    'AccessKey' => '',
    'Block' => 'ItemArea',
    'Description' => '',
    'Link' => 'Action=AdminCustomerUser;Nav=Agent',
    'LinkOption' => '',
    'Name' => 'Customer User Administration',
    'NavBar' => 'Customers',
    'Prio' => '9000',
    'Type' => ''
  }
],
'NavBarModule' => {
  'Block' => 'Customer',
  'Description' => 'Create and manage customer users.',
  'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
  'Name' => 'Customer User',
  'Prio' => '300'
},
'NavBarName' => 'Customers',
'Title' => 'Customer Users'
};
```

### Frontend::Module###AdminCustomerCompany

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminCustomerCompany'} = {
  'Description' => 'Edit Customer Companies.',
  'Group' => [
    'admin',
    'users'
  ],
  'GroupRo' => [
    ''
  ],
  'NavBar' => [
    {
      'AccessKey' => '',
      'Block' => 'ItemArea',
      'Description' => '',
      'Link' => 'Action=AdminCustomerCompany;Nav=Agent',
      'LinkOption' => '',
      'Name' => 'Customer Administration',
      'NavBar' => 'Customers',
      'Prio' => '9100',
      'Type' => ''
    }
  ],
  'NavBarModule' => {
    'Block' => 'Customer',
    'Description' => 'Create and manage customers.',
    'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
    'Name' => 'Customers',
    'Prio' => '310'
  },
  'NavBarName' => 'Customers',
  'Title' => 'Customer Companies'
};
```

### Frontend::Module###AdminCustomerUserGroup

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminCustomerUserGroup'} = {
  'Description' => 'Admin',
  'Group' => [
    'admin'
  ],
  'NavBarModule' => {
    'Block' => 'Customer',
    'Description' => 'Link customer user to groups.',
    'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
    'Name' => 'Customer User <-> Groups',
    'Prio' => '400'
  },
  'NavBarName' => 'Admin',
  'Title' => 'Customers <-> Groups'
};
```

### Frontend::Module###AdminCustomerUserService

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminCustomerUserService'} = {
  'Description' => 'Admin',
  'Group' => [
    'admin'
  ],
  'NavBarModule' => {
    'Block' => 'Customer',
    'Description' => 'Link customer user to services.',
    'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
    'Name' => 'Customer User <-> Services',
    'Prio' => '500'
  },
  'NavBarName' => 'Admin',
  'Title' => 'Customer User <-> Services'
};
```

### Frontend::Module###AdminRole

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminRole'} = {
  'Description' => 'Admin',
  'Group' => [
    'admin'
  ],
  'NavBarModule' => {
    'Block' => 'Agent',
    'Description' => 'Create and manage roles.',
    'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
    'Name' => 'Roles',
    'Prio' => '600'
  },
  'NavBarName' => 'Admin',
  'Title' => 'Roles'
};
```

### Frontend::Module###AdminRoleUser

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminRoleUser'} = {
  'Description' => 'Admin',
```

```
'Group' => [
  'admin'
],
'NavBarModule' => {
  'Block' => 'Agent',
  'Description' => 'Link agents to roles.',
  'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
  'Name' => 'Agents <-> Roles',
  'Prio' => '700'
},
'NavBarName' => 'Admin',
'Title' => 'Agents <-> Roles'
};
```

### Frontend::Module###AdminRoleGroup

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminRoleGroup'} = {
  'Description' => 'Admin',
  'Group' => [
    'admin'
  ],
  'NavBarModule' => {
    'Block' => 'Agent',
    'Description' => 'Link roles to groups.',
    'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
    'Name' => 'Roles <-> Groups',
    'Prio' => '800'
  },
  'NavBarName' => 'Admin',
  'Title' => 'Roles <-> Groups'
};
```

### Frontend::Module###AdminSMIME

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminSMIME'} = {
  'Description' => 'Admin',
  'Group' => [
    'admin'
  ],
  'NavBarModule' => {
    'Block' => 'Email',
    'Description' => 'Manage S/MIME certificates for email encryption.',
    'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
    'Name' => 'S/MIME Certificates',
    'Prio' => '1100'
  },
  'NavBarName' => 'Admin',
  'Title' => 'S/MIME Management'
};
```

### Frontend::Module###AdminPGP

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminPGP'} = {
  'Description' => 'Admin',
  'Group' => [
    'admin'
  ],
  'NavBarModule' => {
```

```
'Block' => 'Email',
'Description' => 'Manage PGP keys for email encryption.',
'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
'Name' => 'PGP Keys',
'Prio' => '1200'
},
'NavBarName' => 'Admin',
'Title' => 'PGP Key Management'
};
```

### Frontend::Module###AdminMailAccount

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminMailAccount'} = {
'Description' => 'Admin',
'Group' => [
'admin'
],
'NavBarModule' => {
'Block' => 'Email',
'Description' => 'Manage POP3 or IMAP accounts to fetch email from.',
'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
'Name' => 'PostMaster Mail Accounts',
'Prio' => '100'
},
'NavBarName' => 'Admin',
'Title' => 'Mail Accounts'
};
```

### Frontend::Module###AdminPostMasterFilter

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminPostMasterFilter'} = {
'Description' => 'Admin',
'Group' => [
'admin'
],
'NavBarModule' => {
'Block' => 'Email',
'Description' => 'Filter incoming emails.',
'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
'Name' => 'PostMaster Filters',
'Prio' => '200'
},
'NavBarName' => 'Admin',
'Title' => 'PostMaster Filters'
};
```

### Frontend::Module###AdminEmail

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminEmail'} = {
'Description' => 'Admin',
'Group' => [
'admin'
],
'NavBarModule' => {
'Block' => 'System',
'Description' => 'Send notifications to users.',
'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
'Name' => 'Admin Notification',
```



```
'Prio' => '400'
},
'NavBarName' => 'Admin',
'Title' => 'Admin Notification'
};
```

### Frontend::Module###AdminSession

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminSession'} = {
  'Description' => 'Admin',
  'Group' => [
    'admin'
  ],
  'NavBarModule' => {
    'Block' => 'System',
    'Description' => 'Manage existing sessions.',
    'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
    'Name' => 'Session Management',
    'Prio' => '500'
  },
  'NavBarName' => 'Admin',
  'Title' => 'Session Management'
};
```

### Frontend::Module###AdminPerformanceLog

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminPerformanceLog'} = {
  'Description' => 'Admin',
  'Group' => [
    'admin'
  ],
  'Loader' => {
    'CSS' => [
      'Core.Agent.Admin.PerformanceLog.css'
    ]
  },
  'NavBarModule' => {
    'Block' => 'System',
    'Description' => 'View performance benchmark results.',
    'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
    'Name' => 'Performance Log',
    'Prio' => '550'
  },
  'NavBarName' => 'Admin',
  'Title' => 'Performance Log'
};
```

### Frontend::Module###AdminRegistration

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminRegistration'} = {
  'Description' => 'Admin',
  'Group' => [
    'admin'
  ],
  'Loader' => {
    'CSS' => [
      'Core.Agent.Admin.Registration.css'
    ]
  },
};
```

```
'NavBarModule' => {
  'Block' => 'System',
  'Description' => 'Manage system registration.',
  'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
  'Name' => 'System Registration',
  'Prio' => '350'
},
'NavBarName' => 'Admin',
'Title' => 'System Registration'
};
```

### Frontend::Module###AdminOTRSBusiness

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminOTRSBusiness'} = {
  'Description' => 'Admin',
  'Group' => [
    'admin'
  ],
  'Loader' => {
    'CSS' => [
      'Core.Agent.Admin.OTRSBusiness.css'
    ]
  },
  'NavBarModule' => {
    'Block' => 'System',
    'Description' => 'Deploy and manage OTRS Business Solution™.',
    'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
    'Name' => 'OTRS Business Solution™',
    'Prio' => '360'
  },
  'NavBarName' => 'Admin',
  'Title' => 'OTRS Business Solution™'
};
```

### Frontend::Module###AdminSupportDataCollector

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminSupportDataCollector'} = {
  'Description' => 'Admin',
  'Group' => [
    'admin'
  ],
  'Loader' => {
    'CSS' => [
      'Core.Agent.Admin.SupportDataCollector.css'
    ]
  },
  'NavBarModule' => {
    'Block' => 'System',
    'Description' => 'Manage support data.',
    'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
    'Name' => 'Support Data Collector',
    'Prio' => '370'
  },
  'NavBarName' => 'Admin',
  'Title' => 'Support Data Collector'
};
```

### Frontend::Module###AdminCloudServices

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminCloudServices'} = {
  'Description' => 'Admin',
  'Group' => [
    'admin'
  ],
  'Loader' => {
    'CSS' => [
      'Core.Agent.Admin.CloudServices.css'
    ]
  },
  'NavBarModule' => {
    'Block' => 'System',
    'Description' => 'Manage OTRS Group cloud services.',
    'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
    'Name' => 'Cloud Services',
    'Prio' => '380'
  },
  'NavBarName' => 'Admin',
  'Title' => 'Cloud Services'
};
```

### Frontend::Module###AdminLog

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminLog'} = {
  'Description' => 'Admin',
  'Group' => [
    'admin'
  ],
  'NavBarModule' => {
    'Block' => 'System',
    'Description' => 'View system log messages.',
    'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
    'Name' => 'System Log',
    'Prio' => '600'
  },
  'NavBarName' => 'Admin',
  'Title' => 'System Log'
};
```

### Frontend::Module###AdminSelectBox

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminSelectBox'} = {
  'Description' => 'Admin',
  'Group' => [
    'admin'
  ],
  'NavBarModule' => {
    'Block' => 'System',
    'Description' => 'Execute SQL statements.',
    'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
    'Name' => 'SQL Box',
    'Prio' => '700'
  },
  'NavBarName' => 'Admin',
  'Title' => 'SQL Box'
};
```

### Frontend::Module###AdminPackageManager

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminPackageManager'} = {
  'Description' => 'Software Package Manager.',
  'Group' => [
    'admin'
  ],
  'NavBarModule' => {
    'Block' => 'System',
    'Description' => 'Update and extend your system with software packages.',
    'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
    'Name' => 'Package Manager',
    'Prio' => '1000'
  },
  'NavBarName' => 'Admin',
  'Title' => 'Package Manager'
};
```

### Frontend::Module###AdminSystemMaintenance

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminSystemMaintenance'} = {
  'Description' => 'Admin',
  'Group' => [
    'admin'
  ],
  'NavBarModule' => {
    'Block' => 'System',
    'Description' => 'Schedule a maintenance period.',
    'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
    'Name' => 'System Maintenance',
    'Prio' => '501'
  },
  'NavBarName' => 'Admin',
  'Title' => 'System Maintenance'
};
```

### Frontend::Module###AdminCloudServiceSupportDataCollector

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminCloudServiceSupportDataCollector'} = {
  'Description' => 'Admin',
  'Group' => [
    'admin'
  ],
  'Loader' => {
    'CSS' => [
      'Core.Agent.Admin.CloudService.SupportDataCollector.css'
    ]
  },
  'Title' => 'Support data collector'
};
```

## Framework → Frontend::Agent

### AgentLogo

The logo shown in the header of the agent interface. The URL to the image can be a relative URL to the skin image directory, or a full URL to a remote web server.

默认值:

```
$Self->{'AgentLogo'} = {
  'StyleHeight' => '55px',
  'StyleRight' => '24px',
```

```
'StyleTop' => '7px',  
'StyleWidth' => '300px',  
'URL' => 'skins/Agent/default/img/logo_bg.png'  
};
```

### **AgentLogoCustom###default**

The logo shown in the header of the agent interface for the skin "default". See "AgentLogo" for further description.

这个设置默认没有激活。

默认值:

```
$Self->{'AgentLogoCustom'}->{'default'} = {  
'StyleHeight' => '55px',  
'StyleRight' => '24px',  
'StyleTop' => '7px',  
'StyleWidth' => '300px',  
'URL' => 'skins/Agent/default/img/logo_bg.png'  
};
```

### **AgentLogoCustom###slim**

The logo shown in the header of the agent interface for the skin "slim". See "AgentLogo" for further description.

这个设置默认没有激活。

默认值:

```
$Self->{'AgentLogoCustom'}->{'slim'} = {  
'StyleHeight' => '55px',  
'StyleRight' => '24px',  
'StyleTop' => '7px',  
'StyleWidth' => '300px',  
'URL' => 'skins/Agent/default/img/logo_bg.png'  
};
```

### **AgentLogoCustom###ivory**

The logo shown in the header of the agent interface for the skin "ivory". See "AgentLogo" for further description.

这个设置默认没有激活。

默认值:

```
$Self->{'AgentLogoCustom'}->{'ivory'} = {  
'StyleHeight' => '55px',  
'StyleRight' => '24px',  
'StyleTop' => '7px',  
'StyleWidth' => '300px',  
'URL' => 'skins/Agent/default/img/logo_bg.png'  
};
```

### **AgentLogoCustom###ivory-slim**

The logo shown in the header of the agent interface for the skin "ivory-slim". See "AgentLogo" for further description.

这个设置默认没有激活。

默认值:

```
$Self->{'AgentLogoCustom'}->{'ivory-slim'} = {  
'StyleHeight' => '45px',  
'StyleRight' => '45px',
```

```
'StyleTop' => '10px',  
'StyleWidth' => '220px',  
'URL' => 'skins/Agent/default/img/logo_bg.png'  
};
```

## AgentLoginLogo

The logo shown on top of the login box of the agent interface. The URL to the image must be relative URL to the skin image directory.

这个设置默认没有激活。

默认值:

```
$Self->{'AgentLoginLogo'} = {  
  'StyleHeight' => '50px',  
  'URL' => 'skins/Agent/default/img/loginlogo_default.png'  
};
```

## LoginURL

Defines an alternate URL, where the login link refers to.

这个设置默认没有激活。

默认值:

```
$Self->{'LoginURL'} = 'http://host.example.com/login.html';
```

## LogoutURL

Defines an alternate URL, where the logout link refers to.

这个设置默认没有激活。

默认值:

```
$Self->{'LogoutURL'} = 'http://host.example.com/thanks-for-using-otrs.html';
```

## PreApplicationModule###AgentInfo

Defines a useful module to load specific user options or to display news.

这个设置默认没有激活。

默认值:

```
$Self->{'PreApplicationModule'}->{'AgentInfo'} = 'Kernel::Modules::AgentInfo';
```

## InfoKey

Defines the key to be checked with Kernel::Modules::AgentInfo module. If this user preferences key is true, the message is accepted by the system.

不能使这个设置失效。

默认值:

```
$Self->{'InfoKey'} = 'wpt22';
```

## InfoFile

File that is displayed in the Kernel::Modules::AgentInfo module, if located under Kernel/Output/HTML/Templates/Standard/AgentInfo.tt.

不能使这个设置失效。

默认值:

```
$Self->{'InfoFile'} = 'AgentInfo';
```

### LostPassword

Activates lost password feature for agents, in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'LostPassword'} = '1';
```

### ShowMotd

Shows the message of the day on login screen of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'ShowMotd'} = '0';
```

### NotificationSubjectLostPasswordToken

Defines the subject for notification mails sent to agents, with token about new requested password.

不能使这个设置失效。

默认值:

```
$Self->{'NotificationSubjectLostPasswordToken'} = 'New OTRS password request';
```

### NotificationBodyLostPasswordToken

Defines the body text for notification mails sent to agents, with token about new requested password (after using this link the new password will be sent).

不能使这个设置失效。

默认值:

```
$Self->{'NotificationBodyLostPasswordToken'} = 'Hi <OTRS_USERFIRSTNAME>,  
  
You or someone impersonating you has requested to change your OTRS  
password.  
  
If you want to do this, click on the link below. You will receive another email  
containing the password.  
  
<OTRS_CONFIG_ContentType>://<OTRS_CONFIG_FQDN>/<OTRS_CONFIG_ScriptAlias>index.pl?  
Action=LostPassword;Token=<OTRS_TOKEN>  
  
If you did not request a new password, please ignore this email.  
';
```

### NotificationSubjectLostPassword

Defines the subject for notification mails sent to agents, about new password.

不能使这个设置失效。

默认值:

```
$Self->{'NotificationSubjectLostPassword'} = 'New OTRS password';
```

## NotificationBodyLostPassword

Defines the body text for notification mails sent to agents, about new password (after using this link the new password will be sent).

不能使这个设置失效。

默认值:

```
$Self->{'NotificationBodyLostPassword'} = 'Hi <OTRS_USERFIRSTNAME>,

Here\'s your new OTRS password.

New password: <OTRS_NEWPW>

You can log in via the following URL:

<OTRS_CONFIG_ContentType>://<OTRS_CONFIG_FQDN>/<OTRS_CONFIG_ScriptAlias>index.pl';
```

## OpenMainMenuOnHover

If enabled, the first level of the main menu opens on mouse hover (instead of click only).

不能使这个设置失效。

默认值:

```
$Self->{'OpenMainMenuOnHover'} = '0';
```

## FirstnameLastnameOrder

Specifies the order in which the firstname and the lastname of agents will be displayed.

不能使这个设置失效。

默认值:

```
$Self->{'FirstnameLastnameOrder'} = '0';
```

## Loader::Agent::Skin###000-default

Default skin for the agent interface.

默认值:

```
$Self->{'Loader::Agent::Skin'}->{'000-default'} = {
  'Description' => 'This is the default orange - black skin.',
  'HomePage' => 'www.otrs.org',
  'InternalName' => 'default',
  'VisibleName' => 'Default'
};
```

## Loader::Agent::Skin###001-slim

Default skin for the agent interface (slim version).

默认值:

```
$Self->{'Loader::Agent::Skin'}->{'001-slim'} = {
  'Description' => '"Slim" skin which tries to save screen space for power users.',
  'HomePage' => 'www.otrs.org',
  'InternalName' => 'slim',
  'VisibleName' => 'Default (Slim)'
};
```



### Loader::Agent::Skin###001-ivory

Balanced white skin by Felix Niklas.

默认值:

```
$Self->{'Loader::Agent::Skin'}->{'001-ivory'} = {  
  'Description' => 'Balanced white skin by Felix Niklas.',  
  'HomePage' => 'www.felixniklas.de',  
  'InternalName' => 'ivory',  
  'VisibleName' => 'Ivory'  
};
```

### Loader::Agent::Skin###001-ivory-slim

Balanced white skin by Felix Niklas (slim version).

默认值:

```
$Self->{'Loader::Agent::Skin'}->{'001-ivory-slim'} = {  
  'Description' => 'Balanced white skin by Felix Niklas (slim version).',  
  'HomePage' => 'www.felixniklas.de',  
  'InternalName' => 'ivory-slim',  
  'VisibleName' => 'Ivory (Slim)'  
};
```

### Loader::Agent::DefaultSelectedSkin

The agent skin's InternalName which should be used in the agent interface. Please check the available skins in Frontend::Agent::Skins.

不能使这个设置失效。

默认值:

```
$Self->{'Loader::Agent::DefaultSelectedSkin'} = 'default';
```

### Loader::Agent::DefaultSelectedSkin::HostBased

It is possible to configure different skins, for example to distinguish between different agents, to be used on a per-domain basis within the application. Using a regular expression (regex), you can configure a Key/Content pair to match a domain. The value in "Key" should match the domain, and the value in "Content" should be a valid skin on your system. Please see the example entries for the proper form of the regex.

这个设置默认没有激活。

默认值:

```
$Self->{'Loader::Agent::DefaultSelectedSkin::HostBased'} = {  
  'host1\\.example\\.com' => 'SomeSkin1',  
  'host2\\.example\\.com' => 'SomeSkin2'  
};
```

### AutoComplete::Agent###Default

Defines the config options for the autocompletion feature.

默认值:

```
$Self->{'AutoComplete::Agent'}->{'Default'} = {  
  'AutoCompleteActive' => '1',  
  'ButtonText' => 'Search',  
  'MaxResultsDisplayed' => '20',  
  'MinQueryLength' => '2',  
  'QueryDelay' => '100'  
};
```

## AutoComplete::Agent###CustomerSearch

Defines the config options for the autocomplete feature.

默认值:

```
$Self->{'AutoComplete::Agent'}->{'CustomerSearch'} = {
  'AutoCompleteActive' => '1',
  'ButtonText' => 'Search Customer',
  'MaxResultsDisplayed' => '20',
  'MinQueryLength' => '2',
  'QueryDelay' => '100'
};
```

## AutoComplete::Agent###UserSearch

Defines the config options for the autocomplete feature.

默认值:

```
$Self->{'AutoComplete::Agent'}->{'UserSearch'} = {
  'AutoCompleteActive' => '1',
  'ButtonText' => 'Search User',
  'MaxResultsDisplayed' => '20',
  'MinQueryLength' => '2',
  'QueryDelay' => '100'
};
```

## PossibleNextActions

Defines the list of possible next actions on an error screen, a full path is required, then is possible to add external links if needed.

默认值:

```
$Self->{'PossibleNextActions'} = {
  'Go to dashboard!' => "[% Env(\`CGIHandle\`) %]?Action=AgentDashboard"
};
```

## ModernizeFormFields

Use new type of select and autocomplete fields in agent interface, where applicable (InputFields).

不能使这个设置失效。

默认值:

```
$Self->{'ModernizeFormFields'} = '1';
```

## Framework → Frontend::Agent::Auth::TwoFactor

### AuthTwoFactorModule

Defines the two-factor module to authenticate agents.

这个设置默认没有激活。

默认值:

```
$Self->{'AuthTwoFactorModule'} =
  'Kernel::System::Auth::TwoFactor::GoogleAuthenticator';
```

### AuthTwoFactorModule::SecretPreferencesKey

Defines the agent preferences key where the shared secret key is stored.

不能使这个设置失效。

默认值:

```
$Self->{'AuthTwoFactorModule::SecretPreferencesKey'} =  
'UserGoogleAuthenticatorSecretKey';
```

### **AuthTwoFactorModule::AllowEmptySecret**

Defines if agents should be allowed to login if they have no shared secret stored in their preferences and therefore are not using two-factor authentication.

默认值:

```
$Self->{'AuthTwoFactorModule::AllowEmptySecret'} = '1';
```

### **AuthTwoFactorModule::AllowPreviousToken**

Defines if the previously valid token should be accepted for authentication. This is slightly less secure but gives users 30 seconds more time to enter their one-time password.

默认值:

```
$Self->{'AuthTwoFactorModule::AllowPreviousToken'} = '1';
```

## **Framework → Frontend::Agent::Dashboard**

### **AgentCustomerInformationCenter::Backend###0600-CIC-CustomerCompanyInformation**

Parameters for the dashboard backend of the customer company information of the agent interface. "Group" is used to restrict the access to the plugin (e. g. Group: admin;group1;group2;). "Default" determines if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" is the cache time in minutes for the plugin.

默认值:

```
$Self->{'AgentCustomerInformationCenter::Backend'}->{'0600-CIC-CustomerCompanyInformation'} = {  
  'Attributes' => '',  
  'Block' => 'ContentSmall',  
  'Default' => '1',  
  'Description' => 'Customer Information',  
  'Group' => '',  
  'Module' => 'Kernel::Output::HTML::Dashboard::CustomerCompanyInformation',  
  'Title' => 'Customer Information'  
};
```

### **DashboardBackend###0000-ProductNotify**

Defines the parameters for the dashboard backend. "Group" is used to restrict access to the plugin (e. g. Group: admin;group1;group2;). "Default" indicates if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" defines the cache expiration period in minutes for the plugin.

默认值:

```
$Self->{'DashboardBackend'}->{'0000-ProductNotify'} = {  
  'Block' => 'ContentLarge',  
  'CacheTTLLocal' => '1440',  
  'Default' => '1',  
  'Description' => 'News about OTRS releases!',  
  'Group' => 'admin',  
  'Module' => 'Kernel::Output::HTML::Dashboard::ProductNotify',
```

```
'Title' => 'Product News'
};
```

### DashboardBackend###0390-UserOutOfOffice

Defines the parameters for the dashboard backend. "Limit" defines the number of entries displayed by default. "Group" is used to restrict access to the plugin (e. g. Group: admin;group1;group2;). "Default" indicates if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" defines the cache expiration period in minutes for the plugin.

默认值:

```
$Self->{'DashboardBackend'}->{'0390-UserOutOfOffice'} = {
  'Block' => 'ContentSmall',
  'CacheTTLLocal' => '5',
  'Default' => '1',
  'Description' => '',
  'Group' => '',
  'Limit' => '10',
  'Module' => 'Kernel::Output::HTML::Dashboard::UserOutOfOffice',
  'SortBy' => 'UserFullname',
  'Title' => 'Out Of Office'
};
```

### DashboardBackend###0400-UserOnline

Defines the parameters for the dashboard backend. "Limit" defines the number of entries displayed by default. "Group" is used to restrict access to the plugin (e. g. Group: admin;group1;group2;). "Default" indicates if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" defines the cache expiration period in minutes for the plugin.

默认值:

```
$Self->{'DashboardBackend'}->{'0400-UserOnline'} = {
  'Block' => 'ContentSmall',
  'CacheTTLLocal' => '5',
  'Default' => '0',
  'Description' => '',
  'Filter' => 'Agent',
  'Group' => '',
  'Limit' => '10',
  'Module' => 'Kernel::Output::HTML::Dashboard::UserOnline',
  'ShowEmail' => '0',
  'SortBy' => 'UserFullname',
  'Title' => 'Logged-In Users'
};
```

### DashboardBackend###0405-News

Defines the parameters for the dashboard backend. "Limit" defines the number of entries displayed by default. "Group" is used to restrict access to the plugin (e. g. Group: admin;group1;group2;). "Default" indicates if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTL" indicates the cache expiration period in minutes for the plugin.

默认值:

```
$Self->{'DashboardBackend'}->{'0405-News'} = {
  'Block' => 'ContentSmall',
  'CacheTTL' => '360',
  'Default' => '1',
  'Description' => '',
  'Group' => '',
  'Limit' => '6',
  'Module' => 'Kernel::Output::HTML::Dashboard::News',
};
```

```
'Title' => 'OTRS News'  
};
```

### DashboardBackend###0410-RSS

Defines the parameters for the dashboard backend. "Limit" defines the number of entries displayed by default. "Group" is used to restrict access to the plugin (e. g. Group: admin;group1;group2;). "Default" indicates if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTL" indicates the cache expiration period in minutes for the plugin.

这个设置默认没有激活。

默认值:

```
$Self->{'DashboardBackend'}->{'0410-RSS'} = {  
  'Block' => 'ContentSmall',  
  'CacheTTL' => '360',  
  'Default' => '1',  
  'Description' => '',  
  'Group' => '',  
  'Limit' => '6',  
  'Module' => 'Kernel::Output::HTML::Dashboard::RSS',  
  'Title' => 'Custom RSS Feed',  
  'URL' => 'http://www.otrs.com/en/rss.xml',  
  'URL_de' => 'http://www.otrs.com/de/rss.xml',  
  'URL_es' => 'http://www.otrs.com/es/rss.xml',  
  'URL_nl' => 'http://www.otrs.com/nl/rss.xml',  
  'URL_ru' => 'http://www.otrs.com/ru/rss.xml',  
  'URL_zh' => 'http://www.otrs.com/cn/rss.xml'  
};
```

### DashboardBackend###0420-CmdOutput

Defines the parameters for the dashboard backend. "Cmd" is used to specify command with parameters. "Group" is used to restrict access to the plugin (e. g. Group: admin;group1;group2;). "Default" indicates if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTL" indicates the cache expiration period in minutes for the plugin.

这个设置默认没有激活。

默认值:

```
$Self->{'DashboardBackend'}->{'0420-CmdOutput'} = {  
  'Block' => 'ContentSmall',  
  'CacheTTL' => '60',  
  'Cmd' => '/bin/echo Configure me please.',  
  'Default' => '0',  
  'Description' => '',  
  'Group' => '',  
  'Module' => 'Kernel::Output::HTML::Dashboard::CmdOutput',  
  'Title' => 'Sample command output'  
};
```

### DashboardBackend###0200-Image

Defines the parameters for the dashboard backend. "Group" is used to restrict access to the plugin (e. g. Group: admin;group1;group2;). "Default" indicates if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTL" indicates the cache expiration period in minutes for the plugin.

这个设置默认没有激活。

默认值:

```
$Self->{'DashboardBackend'}->{'0200-Image'} = {
```

```
'Block' => 'ContentLarge',
'Default' => '1',
'Description' => 'Some picture description!',
'Group' => '',
'Height' => '140',
'Link' => 'http://otrs.org/',
'LinkTitle' => 'http://otrs.org/',
'Module' => 'Kernel::Output::HTML::Dashboard::Image',
'Title' => 'A picture',
'URL' => 'http://www.otrs.com/wp-uploads//2013/10/OTRS_Logo-300x170.png',
'Width' => '198'
};
```

### DashboardBackend###0210-MOTD

Shows the message of the day (MOTD) in the agent dashboard. "Group" is used to restrict access to the plugin (e. g. Group: admin;group1;group2;). "Default" indicates if the plugin is enabled by default or if the user needs to enable it manually.

这个设置默认没有激活。

默认值:

```
$Self->{'DashboardBackend'}->{'0210-MOTD'} = {
  'Block' => 'ContentLarge',
  'Default' => '1',
  'Group' => '',
  'Module' => 'Kernel::Output::HTML::Dashboard::MOTD',
  'Title' => 'Message of the Day'
};
```

### DashboardBackend###0300-IFrame

Defines the parameters for the dashboard backend. "Group" is used to restrict access to the plugin (e. g. Group: admin;group1;group2;). "Default" indicates if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTL" indicates the cache expiration period in minutes for the plugin.

这个设置默认没有激活。

默认值:

```
$Self->{'DashboardBackend'}->{'0300-IFrame'} = {
  'Align' => 'left',
  'Block' => 'ContentLarge',
  'Default' => '1',
  'Description' => 'Some description!',
  'Frameborder' => '1',
  'Group' => '',
  'Height' => '800',
  'Link' => 'http://otrs.org/',
  'LinkTitle' => 'OTRS.org/',
  'Marginheight' => '5',
  'Marginwidth' => '5',
  'Module' => 'Kernel::Output::HTML::Dashboard::IFrame',
  'Scrolling' => 'auto',
  'Title' => 'A Website',
  'URL' => 'http://www.otrs.org/',
  'Width' => '1024'
};
```

### AgentCustomerInformationCenter::Backend###0050-CIC-CustomerUserList

Parameters for the dashboard backend of the customer user list overview of the agent interface. "Limit" is the number of entries shown by default. "Group" is used to restrict the access to the plugin (e. g. Group: admin;group1;group2;). "Default" determines if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" is the cache time in minutes for the plugin.

默认值:

```
$Self->{'AgentCustomerInformationCenter::Backend'}->{'0050-CIC-CustomerUserList'} = {
  'Attributes' => '',
  'Block' => 'ContentLarge',
  'CacheTTLLocal' => '0.5',
  'Default' => '1',
  'Description' => 'All customer users of a CustomerID',
  'Group' => '',
  'Limit' => '10',
  'Module' => 'Kernel::Output::HTML::Dashboard::CustomerUserList',
  'Permission' => 'ro',
  'Title' => 'Customer Users'
};
```

## Framework → Frontend::Agent::LinkObject

### Frontend::AgentLinkObject::WildcardSearch

Starts a wildcard search of the active object after the link object mask is started.

不能使这个设置失效。

默认值:

```
$Self->{'Frontend::AgentLinkObject::WildcardSearch'} = '0';
```

## Framework → Frontend::Agent::ModuleMetaHead

### Frontend::HeaderMetaModule###100-Refresh

Defines the module to generate code for periodic page reloads.

不能使这个设置失效。

默认值:

```
$Self->{'Frontend::HeaderMetaModule'}->{'100-Refresh'} = {
  'Module' => 'Kernel::Output::HTML::HeaderMeta::Refresh'
};
```

## Framework → Frontend::Agent::ModuleNotify

### Frontend::NotifyModule###100-OTRSBusiness

Defines the module to display a notification in different interfaces on different occasions for OTRS Business Solution™.

不能使这个设置失效。

默认值:

```
$Self->{'Frontend::NotifyModule'}->{'100-OTRSBusiness'} = {
  'Group' => 'admin',
  'Module' => 'Kernel::Output::HTML::Notification::AgentOTRSBusiness'
};
```

### Frontend::NotifyModule###200-UID-Check

Defines the module to display a notification in the agent interface, if the system is used by the admin user (normally you shouldn't work as admin).

不能使这个设置失效。

默认值:

```
$Self->{'Frontend::NotifyModule'}->{'200-UID-Check'} = {  
  'Module' => 'Kernel::Output::HTML::Notification::UIDCheck'  
};
```

### Frontend::NotifyModule###250-AgentSessionLimit

Defines the module to display a notification in the agent interface, if the agent session limit prior warning is reached.

不能使这个设置失效。

默认值:

```
$Self->{'Frontend::NotifyModule'}->{'250-AgentSessionLimit'} = {  
  'Module' => 'Kernel::Output::HTML::Notification::AgentSessionLimit'  
};
```

### Frontend::NotifyModule###300-ShowAgentOnline

Defines the module that shows all the currently logged in agents in the agent interface.

这个设置默认没有激活。

默认值:

```
$Self->{'Frontend::NotifyModule'}->{'300-ShowAgentOnline'} = {  
  'Module' => 'Kernel::Output::HTML::Notification::AgentOnline',  
  'ShowEmail' => '1'  
};
```

### Frontend::NotifyModule###400-ShowCustomerOnline

Defines the module that shows all the currently logged in customers in the agent interface.

这个设置默认没有激活。

默认值:

```
$Self->{'Frontend::NotifyModule'}->{'400-ShowCustomerOnline'} = {  
  'Module' => 'Kernel::Output::HTML::Notification::CustomerOnline',  
  'ShowEmail' => '1'  
};
```

### Frontend::NotifyModule###500-OutofOffice-Check

Defines the module to display a notification in the agent interface, if the agent is logged in while having out-of-office active.

不能使这个设置失效。

默认值:

```
$Self->{'Frontend::NotifyModule'}->{'500-OutofOffice-Check'} = {  
  'Module' => 'Kernel::Output::HTML::Notification::OutofOfficeCheck'  
};
```

### Frontend::NotifyModule###600-SystemMaintenance-Check

Defines the module to display a notification in the agent interface, if the agent is logged in while having system maintenance active.

不能使这个设置失效。

默认值:

```
$Self->{'Frontend::NotifyModule'}->{'600-SystemMaintenance-Check'} = {
```



```
'Module' => 'Kernel::Output::HTML::Notification::SystemMaintenanceCheck'
};
```

### Frontend::NotifyModule###900-Generic

Defines the module that shows a generic notification in the agent interface. Either "Text" - if configured - or the contents of "File" will be displayed.

这个设置默认没有激活。

默认值:

```
$Self->{'Frontend::NotifyModule'}->{'900-Generic'} = {
  'File' => '<OTRS_CONFIG_Home>/var/notify.txt',
  'Link' => 'http://www.otrs.com',
  'Module' => 'Kernel::Output::HTML::Notification::Generic',
  'Priority' => 'Warning',
  'Text' => 'The OTRS Website'
};
```

## Framework → Frontend::Agent::ModuleRegistration

### Frontend::Module###Logout

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'Logout'} = {
  'Description' => 'Logout',
  'NavBarName' => '',
  'Title' => ''
};
```

### Frontend::Module###AgentDashboard

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AgentDashboard'} = {
  'Description' => 'Agent Dashboard',
  'Loader' => {
    'CSS' => [
      'Core.Agent.Dashboard.css',
      'Core.AllocationList.css',
      'thirdparty/fullcalendar-2.4.0/fullcalendar.min.css',
      'thirdparty/nvd3-1.7.1/nv.d3.css'
    ],
    'JavaScript' => [
      'thirdparty/momentjs-2.10.6/moment.min.js',
      'thirdparty/fullcalendar-2.4.0/fullcalendar.min.js',
      'thirdparty/d3-3.5.6/d3.min.js',
      'thirdparty/nvd3-1.7.1/nvd3.min.js',
      'thirdparty/nvd3-1.7.1/models/OTRSLineChart.js',
      'thirdparty/nvd3-1.7.1/models/OTRSMultiBarChart.js',
      'thirdparty/nvd3-1.7.1/models/OTRSStackedAreaChart.js',
      'thirdparty/canvg-1.4/rgbcolor.js',
      'thirdparty/canvg-1.4/StackBlur.js',
      'thirdparty/canvg-1.4/canvg.js',
      'thirdparty/StringView-8/stringview.js',
      'Core.UI.AdvancedChart.js',
      'Core.UI.AllocationList.js',
      'Core.Agent.TableFilters.js',
      'Core.Agent.Dashboard.js'
    ]
  },
  'NavBar' => [
    {
```

```

    'AccessKey' => 'd',
    'Block' => 'ItemArea',
    'Description' => '',
    'Link' => 'Action=AgentDashboard',
    'LinkOption' => '',
    'Name' => 'Dashboard',
    'NavBar' => 'Dashboard',
    'Prio' => '50',
    'Type' => 'Menu'
  }
],
'NavBarName' => 'Dashboard',
'Title' => ''
};

```

### Frontend::Module###AgentCustomerInformationCenter

Frontend module registration for the agent interface.

默认值:

```

$self->{'Frontend::Module'}->{'AgentCustomerInformationCenter'} = {
  'Description' => 'Customer Information Center.',
  'Loader' => {
    'CSS' => [
      'Core.AllocationList.css'
    ],
    'JavaScript' => [
      'Core.UI.AllocationList.js',
      'Core.Agent.Dashboard.js',
      'Core.Agent.TableFilters.js'
    ]
  },
  'NavBar' => [
    {
      'AccessKey' => 'c',
      'Block' => 'ItemArea',
      'Description' => '',
      'Link' => 'Action=AgentCustomerInformationCenter',
      'LinkOption' => 'onclick="window.setTimeout(function()
{Core.Agent.CustomerInformationCenterSearch.OpenSearchDialog();}, 0); return false;"',
      'Name' => 'Customer Information Center',
      'NavBar' => 'Customers',
      'Prio' => '50',
      'Type' => ''
    },
    {
      'AccessKey' => '',
      'Block' => 'ItemArea',
      'Description' => '',
      'Link' => 'Action=AgentCustomerInformationCenter',
      'LinkOption' => '',
      'Name' => 'Customers',
      'NavBar' => 'Customers',
      'Prio' => '60',
      'Type' => 'Menu'
    }
  ],
  'NavBarName' => 'Customers',
  'Title' => ''
};

```

### Frontend::Module###AgentCustomerInformationCenterSearch

Frontend module registration for the agent interface.

默认值:

```

$self->{'Frontend::Module'}->{'AgentCustomerInformationCenterSearch'} = {
  'Description' => 'Customer Information Center Search.',
  'Title' => ''
};

```

```
};
```

### Frontend::Module###AgentPreferences

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AgentPreferences'} = {  
  'Description' => 'Agent Preferences.',  
  'Loader' => {  
    'CSS' => [  
      'Core.Agent.Preferences.css'  
    ]  
  },  
  'NavBarName' => 'Preferences',  
  'Title' => ''  
};
```

### Frontend::Module###PictureUpload

Frontend module registration for the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Frontend::Module'}->{'PictureUpload'} = {  
  'Description' => 'Picture upload module.',  
  'NavBarName' => 'Ticket',  
  'Title' => 'Picture Upload'  
};
```

### Frontend::Module###AgentSpelling

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AgentSpelling'} = {  
  'Description' => 'Spell checker.',  
  'Loader' => {  
    'JavaScript' => [  
      'Core.Agent.TicketAction.js'  
    ]  
  },  
  'NavBarName' => '',  
  'Title' => 'Spell Checker'  
};
```

### Frontend::Module###SpellingInline

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'SpellingInline'} = {  
  'Description' => 'Spell checker.',  
  'NavBarName' => '',  
  'Title' => 'Spell Checker'  
};
```

### Frontend::Module###AgentBook

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AgentBook'} = {
```

```
'Description' => 'Address book of CustomerUser sources.',
'Loader' => {
  'JavaScript' => [
    'Core.Agent.CustomerSearch.js',
    'Core.Agent.TicketAction.js'
  ]
},
'NavBarName' => '',
'Title' => 'Address Book'
};
```

### Frontend::Module###AgentLinkObject

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AgentLinkObject'} = {
  'Description' => 'Link Object.',
  'NavBarName' => '',
  'Title' => 'Link Object'
};
```

### Frontend::Module###AgentInfo

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AgentInfo'} = {
  'Description' => 'Generic Info module.',
  'NavBarName' => '',
  'Title' => 'Info'
};
```

### Frontend::Module###AgentSearch

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AgentSearch'} = {
  'Description' => 'Global Search Module.',
  'NavBarName' => '',
  'Title' => 'Search'
};
```

### Frontend::Module###AgentOTRSBusiness

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AgentOTRSBusiness'} = {
  'Description' => 'Agent',
  'Loader' => {
    'CSS' => [
      'Core.Agent.Admin.OTRSBusiness.css'
    ]
  },
  'NavBarName' => '',
  'Title' => 'OTRS Business Solution™'
};
```

### CustomerFrontend::Module###SpellingInline

Frontend module registration for the customer interface.

默认值:

```
$Self->{'CustomerFrontend::Module'}->{'SpellingInline'} = {
  'Description' => 'Spell checker.',
  'NavBarName' => '',
  'Title' => 'Spell Checker'
};
```

### Frontend::Module###AgentHTMLReference

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AgentHTMLReference'} = {
  'Description' => 'HTML Reference.',
  'Group' => [
    'users'
  ],
  'GroupRo' => [
    'users'
  ],
  'Loader' => {
    'CSS' => [
      'Core.Agent.HTMLReference.css'
    ]
  },
  'NavBarName' => '',
  'Title' => 'HTML Reference'
};
```

### Frontend::Module###AgentStatistics

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AgentStatistics'} = {
  'Description' => '',
  'Group' => [
    'stats'
  ],
  'GroupRo' => [
    'stats'
  ],
  'Loader' => {
    'CSS' => [
      'thirdparty/nvd3-1.7.1/nv.d3.css',
      'Core.Agent.Statistics.css'
    ],
    'JavaScript' => [
      'thirdparty/d3-3.5.6/d3.min.js',
      'thirdparty/nvd3-1.7.1/nvd3.min.js',
      'thirdparty/nvd3-1.7.1/models/OTRSLineChart.js',
      'thirdparty/nvd3-1.7.1/models/OTRSMultiBarChart.js',
      'thirdparty/nvd3-1.7.1/models/OTRSStackedAreaChart.js',
      'thirdparty/canvg-1.4/rgbcolor.js',
      'thirdparty/canvg-1.4/StackBlur.js',
      'thirdparty/canvg-1.4/canvg.js',
      'thirdparty/StringView-8/stringview.js',
      'Core.Agent.Statistics.js',
      'Core.UI.AdvancedChart.js'
    ]
  },
  'NavBar' => [
    {
      'AccessKey' => '',
      'Block' => 'ItemArea',
      'Description' => '',
      'Link' => 'Action=AgentStatistics;Subaction=0verview',
      'LinkOption' => '',
      'Name' => 'Reports',
      'NavBar' => 'Reports',
    }
  ]
};
```

```

    'Prio' => '8500',
    'Type' => 'Menu'
  },
  {
    'AccessKey' => '',
    'Block' => '',
    'Description' => '',
    'GroupRo' => [
      'stats'
    ],
    'Link' => 'Action=AgentStatisticsReports;Subaction=Overview',
    'LinkOption' => 'class="OTRSBusinessRequired"',
    'Name' => 'Reports (OTRS Business Solution™)',
    'NavBar' => 'Reports',
    'Prio' => '100',
    'Type' => ''
  },
  {
    'AccessKey' => '',
    'Block' => '',
    'Description' => '',
    'GroupRo' => [
      'stats'
    ],
    'Link' => 'Action=AgentStatistics;Subaction=Overview',
    'LinkOption' => '',
    'Name' => 'Statistics',
    'NavBar' => 'Reports',
    'Prio' => '200',
    'Type' => ''
  }
],
'NavBarName' => 'Reports',
'Title' => 'Statistics'
};

```

## Framework → Frontend::Agent::NavBarModule

### Frontend::NavBarModule###6-CustomerCompany

Frontend module registration (disable company link if no company feature is used).

默认值:

```

$self->{'Frontend::NavBarModule'}->{'6-CustomerCompany'} = {
  'Module' => 'Kernel::Output::HTML::NavBar::CustomerCompany'
};

```

### Frontend::NavBarModule###7-AgentTicketService

Frontend module registration (disable AgentTicketService link if Ticket Service feature is not used).

默认值:

```

$self->{'Frontend::NavBarModule'}->{'7-AgentTicketService'} = {
  'Module' => 'Kernel::Output::HTML::NavBar::AgentTicketService'
};

```

## Framework → Frontend::Agent::Preferences

### PreferencesTable

Defines the name of the table where the user preferences are stored.

不能使这个设置失效。

默认值:

```
$Self->{'PreferencesTable'} = 'user_preferences';
```

### PreferencesTableKey

Defines the column to store the keys for the preferences table.

不能使这个设置失效。

默认值:

```
$Self->{'PreferencesTableKey'} = 'preferences_key';
```

### PreferencesTableValue

Defines the name of the column to store the data in the preferences table.

不能使这个设置失效。

默认值:

```
$Self->{'PreferencesTableValue'} = 'preferences_value';
```

### PreferencesTableUserID

Defines the name of the column to store the user identifier in the preferences table.

不能使这个设置失效。

默认值:

```
$Self->{'PreferencesTableUserID'} = 'user_id';
```

### PreferencesView

Sets the display order of the different items in the preferences view.

不能使这个设置失效。

默认值:

```
$Self->{'PreferencesView'} = [  
  'User Profile',  
  'Notification Settings',  
  'Other Settings'  
];
```

### PreferencesGroups###Password

Defines the config parameters of this item, to be shown in the preferences view. 'PasswordRegExp' allows to match passwords against a regular expression. Define the minimum number of characters using 'PasswordMinSize'. Define if at least 2 lowercase and 2 uppercase letter characters are needed by setting the appropriate option to '1'. 'PasswordMin2Characters' defines if the password needs to contain at least 2 letter characters (set to 0 or 1). 'PasswordNeedDigit' controls the need of at least 1 digit (set to 0 or 1 to control). 'PasswordMaxLoginFailed' allows to set an agent to invalid temporarily if max failed logins reached.

默认值:

```
$Self->{'PreferencesGroups'}->{'Password'} = {  
  'Active' => '1',  
  'Area' => 'Agent',  
  'Column' => 'User Profile',  
  'Desc' => 'Set a new password by filling in your current password and a new one.',  
  'Label' => 'Change password',
```

```
'Module' => 'Kernel::Output::HTML::Preferences::Password',
'PasswordMaxLoginFailed' => '0',
'PasswordMin2Characters' => '0',
'PasswordMin2Lower2UpperCharacters' => '0',
'PasswordMinSize' => '0',
'PasswordNeedDigit' => '0',
'PasswordRegExp' => '',
'Prio' => '0500'
};
```

### PreferencesGroups###GoogleAuthenticatorSecretKey

Defines the config parameters of this item, to be shown in the preferences view.

默认值:

```
$Self->{'PreferencesGroups'}->{'GoogleAuthenticatorSecretKey'} = {
'Active' => '0',
'Block' => 'Input',
'Column' => 'User Profile',
'Desc' => 'Enter your shared secret to enable two factor authentication.',
'Key' => 'Shared Secret',
'Label' => 'Google Authenticator',
'Module' => 'Kernel::Output::HTML::Preferences::Generic',
'PrefKey' => 'UserGoogleAuthenticatorSecretKey',
'Prio' => '0600'
};
```

### PreferencesGroups###SpellDict

Defines the config parameters of this item, to be shown in the preferences view. Take care to maintain the dictionaries installed in the system in the data section.

这个设置默认没有激活。

默认值:

```
$Self->{'PreferencesGroups'}->{'SpellDict'} = {
'Active' => '1',
'Column' => 'User Profile',
'Data' => {
'deutsch' => 'Deutsch',
'english' => 'English'
},
'DataSelected' => 'english',
'Desc' => 'Select your default spelling dictionary.',
'Key' => 'Default spelling dictionary',
'Label' => 'Spelling Dictionary',
'Module' => 'Kernel::Output::HTML::Preferences::Generic',
'PrefKey' => 'UserSpellDict',
'Prio' => '2000'
};
```

### PreferencesGroups###Comment

Defines the config parameters of this item, to be shown in the preferences view.

默认值:

```
$Self->{'PreferencesGroups'}->{'Comment'} = {
'Active' => '0',
'Block' => 'Input',
'Column' => 'Other Settings',
'Data' => "[% Env('UserComment') %]",
'Desc' => 'Add a comment.',
'Key' => 'Comment',
'Label' => 'Comment',
'Module' => 'Kernel::Output::HTML::Preferences::Generic',
'PrefKey' => 'UserComment',
'Prio' => '6000'
};
```



```
};
```

### PreferencesGroups###Language

Defines the config parameters of this item, to be shown in the preferences view.

默认值:

```
$Self->{'PreferencesGroups'}->{'Language'} = {  
  'Active' => '1',  
  'Column' => 'User Profile',  
  'Desc' => 'Select the main interface language.',  
  'Key' => 'Language',  
  'Label' => 'Language',  
  'Module' => 'Kernel::Output::HTML::Preferences::Language',  
  'PrefKey' => 'UserLanguage',  
  'Prio' => '1000'  
};
```

### PreferencesGroups###Skin

Defines the config parameters of this item, to be shown in the preferences view.

默认值:

```
$Self->{'PreferencesGroups'}->{'Skin'} = {  
  'Active' => '1',  
  'Column' => 'Other Settings',  
  'Desc' => 'Select your preferred layout for OTRS.',  
  'Key' => 'Skin',  
  'Label' => 'Skin',  
  'Module' => 'Kernel::Output::HTML::Preferences::Skin',  
  'PrefKey' => 'UserSkin',  
  'Prio' => '100'  
};
```

### PreferencesGroups###Theme

Defines the config parameters of this item, to be shown in the preferences view.

默认值:

```
$Self->{'PreferencesGroups'}->{'Theme'} = {  
  'Active' => '1',  
  'Column' => 'User Profile',  
  'Desc' => 'Select your preferred theme for OTRS.',  
  'Key' => 'Frontend theme',  
  'Label' => 'Theme',  
  'Module' => 'Kernel::Output::HTML::Preferences::Theme',  
  'PrefKey' => 'UserTheme',  
  'Prio' => '3000'  
};
```

### PreferencesGroups###OutOfOffice

Defines the config parameters of this item, to be shown in the preferences view.

默认值:

```
$Self->{'PreferencesGroups'}->{'OutOfOffice'} = {  
  'Active' => '1',  
  'Block' => 'OutOfOffice',  
  'Column' => 'User Profile',  
  'Desc' => 'If you\'re going to be out of office, you may wish to let other users know  
by setting the exact dates of your absence.',  
  'Key' => '',  
  'Label' => 'Out Of Office Time',  
  'Module' => 'Kernel::Output::HTML::Preferences::OutOfOffice',  
  'PrefKey' => 'UserOutOfOffice',  
};
```

```
'Prio' => '4000'
};
```

### PreferencesGroups###TimeZone

Defines the config parameters of this item, to be shown in the preferences view.

默认值:

```
$Self->{'PreferencesGroups'}->{'TimeZone'} = {
  'Active' => '1',
  'Column' => 'User Profile',
  'Desc' => 'Select your time zone.',
  'Key' => 'Time Zone',
  'Label' => 'Time Zone',
  'Module' => 'Kernel::Output::HTML::Preferences::TimeZone',
  'PrefKey' => 'UserTimeZone',
  'Prio' => '5000'
};
```

### PreferencesGroups###CSVSeparator

Gives end users the possibility to override the separator character for CSV files, defined in the translation files.

这个设置默认没有激活。

默认值:

```
$Self->{'PreferencesGroups'}->{'CSVSeparator'} = {
  'Active' => '1',
  'Column' => 'Other Settings',
  'Data' => {
    '' => ',',
    ',' => ',',
    ';' => ';',
    '\\t' => 'tab',
    '|' => '|'
  },
  'DataSelected' => '0',
  'Desc' => 'Select the separator character used in CSV files (stats and searches). If you don\\t select a separator here, the default separator for your language will be used.',
  'Key' => 'CSV Separator',
  'Label' => 'CSV Separator',
  'Module' => 'Kernel::Output::HTML::Preferences::Generic',
  'PrefKey' => 'UserCSVSeparator',
  'Prio' => '4000'
};
```

## Framework → Frontend::Agent::SearchRouter

### Frontend::SearchDefault

Search backend default router.

默认值:

```
$Self->{'Frontend::SearchDefault'} = 'Action=AgentTicketSearch;Subaction=AJAX';
```

## Framework → Frontend::Agent::Stats

### Stats::SearchPageShown

Defines the default maximum number of statistics per page on the overview screen.

不能使这个设置失效。

默认值:

```
$Self->{'Stats::SearchPageShown'} = '50';
```

### **Stats::DefaultSelectedDynamicObject**

Defines the default selection at the drop down menu for dynamic objects (Form: Common Specification).

不能使这个设置失效。

默认值:

```
$Self->{'Stats::DefaultSelectedDynamicObject'} = 'Ticket';
```

### **Stats::DefaultSelectedPermissions**

Defines the default selection at the drop down menu for permissions (Form: Common Specification).

不能使这个设置失效。

默认值:

```
$Self->{'Stats::DefaultSelectedPermissions'} = [  
  'stats'  
];
```

### **Stats::DefaultSelectedFormat**

Defines the default selection at the drop down menu for stats format (Form: Common Specification). Please insert the format key (see Stats::Format).

不能使这个设置失效。

默认值:

```
$Self->{'Stats::DefaultSelectedFormat'} = [  
  'Print',  
  'CSV',  
  'Excel',  
  'D3::BarChart',  
  'D3::LineChart',  
  'D3::StackedAreaChart'  
];
```

### **Stats::SearchLimit**

Defines the search limit for the stats.

不能使这个设置失效。

默认值:

```
$Self->{'Stats::SearchLimit'} = '1000';
```

### **Stats::Format**

Defines all the possible stats output formats.

不能使这个设置失效。

默认值:

```
$Self->{'Stats::Format'} = {  
  'CSV' => 'CSV',
```

```
'D3::BarChart' => 'Graph: Bar Chart',
'D3::LineChart' => 'Graph: Line Chart',
'D3::StackedAreaChart' => 'Graph: Stacked Area Chart',
'Excel' => 'Excel',
'Print' => 'Print'
};
```

### Stats::ExchangeAxis

Allows agents to exchange the axis of a stat if they generate one.

不能使这个设置失效。

默认值:

```
$Self->{'Stats::ExchangeAxis'} = '0';
```

### Stats::UseAgentElementInStats

Allows agents to generate individual-related stats.

不能使这个设置失效。

默认值:

```
$Self->{'Stats::UseAgentElementInStats'} = '0';
```

### Stats::UseInvalidAgentInStats

Allows invalid agents to generate individual-related stats.

不能使这个设置失效。

默认值:

```
$Self->{'Stats::UseInvalidAgentInStats'} = '1';
```

### Stats::CustomerIDAsMultiSelect

Shows all the customer identifiers in a multi-select field (not useful if you have a lot of customer identifiers).

不能使这个设置失效。

默认值:

```
$Self->{'Stats::CustomerIDAsMultiSelect'} = '1';
```

### Stats::CustomerUserLoginsAsMultiSelect

Shows all the customer user identifiers in a multi-select field (not useful if you have a lot of customer user identifiers).

不能使这个设置失效。

默认值:

```
$Self->{'Stats::CustomerUserLoginsAsMultiSelect'} = '0';
```

## Framework → Frontend::Customer

### CustomerHeadline

The headline shown in the customer interface.

不能使这个设置失效。

默认值:

```
$Self->{'CustomerHeadline'} = 'Example Company';
```

## CustomerLogo

The logo shown in the header of the customer interface. The URL to the image can be a relative URL to the skin image directory, or a full URL to a remote web server.

这个设置默认没有激活。

默认值:

```
$Self->{'CustomerLogo'} = {  
  'StyleHeight' => '45px',  
  'StyleRight' => '25px',  
  'StyleTop' => '9px',  
  'StyleWidth' => '300px',  
  'URL' => 'skins/Customer/default/img/logo.png'  
};
```

## CustomerPanelUserID

Defines the user identifier for the customer panel.

不能使这个设置失效。

默认值:

```
$Self->{'CustomerPanelUserID'} = '1';
```

## CustomerGroupSupport

Activates support for customer groups.

不能使这个设置失效。

默认值:

```
$Self->{'CustomerGroupSupport'} = '0';
```

## CustomerGroupAlwaysGroups

Defines the groups every customer user will be in (if CustomerGroupSupport is enabled and you don't want to manage every user for these groups).

不能使这个设置失效。

默认值:

```
$Self->{'CustomerGroupAlwaysGroups'} = [  
  'users'  
];
```

## CustomerPanelLoginURL

Defines an alternate login URL for the customer panel..

这个设置默认没有激活。

默认值:

```
$Self->{'CustomerPanelLoginURL'} = 'http://host.example.com/cgi-bin/login.pl';
```

## CustomerPanelLogoutURL

Defines an alternate logout URL for the customer panel.

这个设置默认没有激活。

默认值:

```
$Self->{'CustomerPanelLogoutURL'} = 'http://host.example.com/cgi-bin/login.pl';
```

## Frontend::CustomerUser::Item###1-GoogleMaps

Defines a customer item, which generates a google maps icon at the end of a customer info block.

默认值:

```
$Self->{'Frontend::CustomerUser::Item'}->{'1-GoogleMaps'} = {
  'Attributes' => 'UserStreet;UserCity;UserCountry;',
  'CSS' => 'Core.Agent.CustomerUser.GoogleMaps.css',
  'CSSClass' => 'GoogleMaps',
  'IconName' => 'fa-globe',
  'Module' => 'Kernel::Output::HTML::CustomerUser::Generic',
  'Required' => 'UserStreet;UserCity;',
  'Target' => '_blank',
  'Text' => 'Location',
  'URL' => 'http://maps.google.com/maps?z=7&q='
};
```

## Frontend::CustomerUser::Item###2-Google

Defines a customer item, which generates a google icon at the end of a customer info block.

这个设置默认没有激活。

默认值:

```
$Self->{'Frontend::CustomerUser::Item'}->{'2-Google'} = {
  'Attributes' => 'UserFirstname;UserLastname;',
  'CSS' => 'Core.Agent.CustomerUser.Google.css',
  'CSSClass' => 'Google',
  'IconName' => 'fa-google',
  'Module' => 'Kernel::Output::HTML::CustomerUser::Generic',
  'Required' => 'UserFirstname;UserLastname;',
  'Target' => '_blank',
  'Text' => 'Google',
  'URL' => 'http://google.com/search?q='
};
```

## Frontend::CustomerUser::Item###2-LinkedIn

Defines a customer item, which generates a LinkedIn icon at the end of a customer info block.

这个设置默认没有激活。

默认值:

```
$Self->{'Frontend::CustomerUser::Item'}->{'2-LinkedIn'} = {
  'Attributes' => 'UserFirstname;UserLastname;',
  'CSS' => 'Core.Agent.CustomerUser.Linkedin.css',
  'CSSClass' => 'LinkedIn',
  'IconName' => 'fa-linkedin',
  'Module' => 'Kernel::Output::HTML::CustomerUser::Generic',
  'Required' => 'UserFirstname;UserLastname;',
  'Target' => '_blank',
};
```

```
'Text' => 'LinkedIn',
'URL' => 'http://www.linkedin.com/commonSearch?type=people&keywords='
};
```

### Frontend::CustomerUser::Item###3-XING

Defines a customer item, which generates a XING icon at the end of a customer info block.

这个设置默认没有激活。

默认值:

```
$Self->{'Frontend::CustomerUser::Item'}->{'3-XING'} = {
  'Attributes' => 'UserFirstname;UserLastname;',
  'CSS' => 'Core.Agent.CustomerUser.Xing.css',
  'CSSClass' => 'Xing',
  'IconName' => 'fa-xing',
  'Module' => 'Kernel::Output::HTML::CustomerUser::Generic',
  'Required' => 'UserFirstname;UserLastname;',
  'Target' => '_blank',
  'Text' => 'XING',
  'URL' => 'https://www.xing.com/app/search?op=search;keywords='
};
```

### CustomerPanelPreApplicationModule###CustomerAccept

This module and its PreRun() function will be executed, if defined, for every request. This module is useful to check some user options or to display news about new applications.

这个设置默认没有激活。

默认值:

```
$Self->{'CustomerPanelPreApplicationModule'}->{'CustomerAccept'} =
'Kernel::Modules::CustomerAccept';
```

### CustomerPanel::InfoKey

Defines the key to check with CustomerAccept. If this user preferences key is true, then the message is accepted by the system.

这个设置默认没有激活。

默认值:

```
$Self->{'CustomerPanel::InfoKey'} = 'CustomerAccept1';
```

### CustomerPanel::InfoFile

Defines the path of the shown info file, that is located under Kernel/Output/HTML/Templates/Standard/CustomerAccept.tt.

这个设置默认没有激活。

默认值:

```
$Self->{'CustomerPanel::InfoFile'} = 'CustomerAccept';
```

### CustomerPanelLostPassword

Activates lost password feature for customers.

不能使这个设置失效。

默认值:

```
$Self->{'CustomerPanelLostPassword'} = '1';
```

### CustomerPanelCreateAccount

Enables customers to create their own accounts.

不能使这个设置失效。

默认值:

```
$Self->{'CustomerPanelCreateAccount'} = '1';
```

### CustomerPanelCreateAccount::MailRestrictions::Whitelist

If active, one of the regular expressions has to match the user's email address to allow registration.

这个设置默认没有激活。

默认值:

```
$Self->{'CustomerPanelCreateAccount::MailRestrictions::Whitelist'} = [  
  '\\@your\\.domain\\.example$'  
];
```

### CustomerPanelCreateAccount::MailRestrictions::Blacklist

If active, none of the regular expressions may match the user's email address to allow registration.

这个设置默认没有激活。

默认值:

```
$Self->{'CustomerPanelCreateAccount::MailRestrictions::Blacklist'} = [  
  '\\@your\\.domain\\.example$'  
];
```

### CustomerPanelSubjectLostPasswordToken

Defines the subject for notification mails sent to customers, with token about new requested password.

不能使这个设置失效。

默认值:

```
$Self->{'CustomerPanelSubjectLostPasswordToken'} = 'New OTRS password request';
```

### CustomerPanelBodyLostPasswordToken

Defines the body text for notification mails sent to customers, with token about new requested password (after using this link the new password will be sent).

不能使这个设置失效。

默认值:

```
$Self->{'CustomerPanelBodyLostPasswordToken'} = 'Hi <OTRS_USERFIRSTNAME>,  
  
You or someone impersonating you has requested to change your OTRS  
password.'
```



If you want to do this, click on this link. You will receive another email containing the password.

```
<OTRS_CONFIG_ContentType>://<OTRS_CONFIG_FQDN>/<OTRS_CONFIG_ScriptAlias>customer.pl?
Action=CustomerLostPassword;Token=<OTRS_TOKEN>
```

If you did not request a new password, please ignore this email.

### CustomerPanelSubjectLostPassword

Defines the subject for notification mails sent to customers, about new password.

不能使这个设置失效。

默认值:

```
$Self->{'CustomerPanelSubjectLostPassword'} = 'New OTRS password';
```

### CustomerPanelBodyLostPassword

Defines the body text for notification mails sent to customers, about new password (after using this link the new password will be sent).

不能使这个设置失效。

默认值:

```
$Self->{'CustomerPanelBodyLostPassword'} = 'Hi <OTRS_USERFIRSTNAME>,

New password: <OTRS_NEWPW>

<OTRS_CONFIG_ContentType>://<OTRS_CONFIG_FQDN>/<OTRS_CONFIG_ScriptAlias>customer.pl';
```

### CustomerPanelSubjectNewAccount

Defines the subject for notification mails sent to customers, about new account.

不能使这个设置失效。

默认值:

```
$Self->{'CustomerPanelSubjectNewAccount'} = 'New OTRS Account!';
```

### CustomerPanelBodyNewAccount

Defines the body text for notification mails sent to customers, about new account.

不能使这个设置失效。

默认值:

```
$Self->{'CustomerPanelBodyNewAccount'} = 'Hi <OTRS_USERFIRSTNAME>,

You or someone impersonating you has created a new OTRS account for
you.

Full name: <OTRS_USERFIRSTNAME> <OTRS_USERLASTNAME>
User name: <OTRS_USERLOGIN>
Password : <OTRS_USERPASSWORD>

You can log in via the following URL. We encourage you to change your password
via the Preferences button after logging in.

<OTRS_CONFIG_ContentType>://<OTRS_CONFIG_FQDN>/<OTRS_CONFIG_ScriptAlias>customer.pl';
```

## Loader::Customer::Skin###000-default

Default skin for the customer interface.

默认值:

```
$Self->{'Loader::Customer::Skin'}->{'000-default'} = {  
  'Description' => 'This is the default orange - black skin for the customer  
  interface.',  
  'HomePage' => 'www.otrs.org',  
  'InternalName' => 'default',  
  'VisibleName' => 'Default'  
};
```

## Loader::Customer::SelectedSkin

The customer skin's InternalName which should be used in the customer interface. Please check the available skins in Frontend::Customer::Skins.

不能使这个设置失效。

默认值:

```
$Self->{'Loader::Customer::SelectedSkin'} = 'default';
```

## Loader::Customer::SelectedSkin::HostBased

It is possible to configure different skins, for example to distinguish between different customers, to be used on a per-domain basis within the application. Using a regular expression (regex), you can configure a Key/Content pair to match a domain. The value in "Key" should match the domain, and the value in "Content" should be a valid skin on your system. Please see the example entries for the proper form of the regex.

这个设置默认没有激活。

默认值:

```
$Self->{'Loader::Customer::SelectedSkin::HostBased'} = {  
  'host1\\.example\\.com' => 'Someskin1',  
  'host2\\.example\\.com' => 'Someskin2'  
};
```

## AutoComplete::Customer###Default

Defines the config options for the autocomplete feature.

默认值:

```
$Self->{'AutoComplete::Customer'}->{'Default'} = {  
  'AutoCompleteActive' => '1',  
  'ButtonText' => 'Search',  
  'MaxResultsDisplayed' => '20',  
  'MinQueryLength' => '2',  
  'QueryDelay' => '100'  
};
```

## ModernizeCustomerFormFields

Use new type of select and autocomplete fields in customer interface, where applicable (InputFields).

不能使这个设置失效。

默认值:

```
$Self->{'ModernizeCustomerFormFields'} = '1';
```

## Framework → Frontend::Customer::Auth

### Customer::AuthModule

Defines the module to authenticate customers.

不能使这个设置失效。

默认值:

```
$Self->{'Customer::AuthModule'} = 'Kernel::System::CustomerAuth::DB';
```

### Customer::AuthModule::DB::CryptType

If "DB" was selected for Customer::AuthModule, the crypt type of passwords must be specified.

不能使这个设置失效。

默认值:

```
$Self->{'Customer::AuthModule::DB::CryptType'} = 'sha2';
```

### Customer::AuthModule::DB::Table

If "DB" was selected for Customer::AuthModule, the name of the table where your customer data should be stored must be specified.

不能使这个设置失效。

默认值:

```
$Self->{'Customer::AuthModule::DB::Table'} = 'customer_user';
```

### Customer::AuthModule::DB::CustomerKey

If "DB" was selected for Customer::AuthModule, the name of the column for the CustomerKey in the customer table must be specified.

不能使这个设置失效。

默认值:

```
$Self->{'Customer::AuthModule::DB::CustomerKey'} = 'login';
```

### Customer::AuthModule::DB::CustomerPassword

If "DB" was selected for Customer::AuthModule, the column name for the CustomerPassword in the customer table must be specified.

不能使这个设置失效。

默认值:

```
$Self->{'Customer::AuthModule::DB::CustomerPassword'} = 'pw';
```

### Customer::AuthModule::DB::DSN

If "DB" was selected for Customer::AuthModule, the DSN for the connection to the customer table must be specified.

这个设置默认没有激活。

默认值:

```
$Self->{'Customer::AuthModule::DB::DSN'} =  
'DBI:mysql:database=customerdb;host=customerdbhost';
```

### Customer::AuthModule::DB::User

If "DB" was selected for Customer::AuthModule, a username to connect to the customer table can be specified.

这个设置默认没有激活。

默认值:

```
$Self->{'Customer::AuthModule::DB::User'} = 'some_user';
```

### Customer::AuthModule::DB::Password

If "DB" was selected for Customer::AuthModule, a password to connect to the customer table can be specified.

这个设置默认没有激活。

默认值:

```
$Self->{'Customer::AuthModule::DB::Password'} = 'some_password';
```

### Customer::AuthModule::DB::Type

If "DB" was selected for Customer::AuthModule, a database driver (normally autodetection is used) can be specified.

这个设置默认没有激活。

默认值:

```
$Self->{'Customer::AuthModule::DB::Type'} = 'mysql';
```

### Customer::AuthModule::HTTPBasicAuth::Replace

If "HTTPBasicAuth" was selected for Customer::AuthModule, you can specify to strip leading parts of user names (e. g. for domains like example\_domain\user to user).

这个设置默认没有激活。

默认值:

```
$Self->{'Customer::AuthModule::HTTPBasicAuth::Replace'} = 'example_domain\\';
```

### Customer::AuthModule::HTTPBasicAuth::ReplaceRegExp

If "HTTPBasicAuth" was selected for Customer::AuthModule, you can specify (by using a RegExp) to strip parts of REMOTE\_USER (e. g. for to remove trailing domains).  
RegExp-Note, \$1 will be the new Login.

这个设置默认没有激活。

默认值:

```
$Self->{'Customer::AuthModule::HTTPBasicAuth::ReplaceRegExp'} = '^(.+?)@.+?';
```

### Customer::AuthModule::LDAP::Host

If "LDAP" was selected for Customer::AuthModule, the LDAP host can be specified.

这个设置默认没有激活。

默认值:

```
$Self->{'Customer::AuthModule::LDAP::Host'} = 'ldap.example.com';
```

### **Customer::AuthModule::LDAP::BaseDN**

If "LDAP" was selected for Customer::AuthModule, the BaseDN must be specified.

这个设置默认没有激活。

默认值:

```
$Self->{'Customer::AuthModule::LDAP::BaseDN'} = 'dc=example,dc=com';
```

### **Customer::AuthModule::LDAP::UID**

If "LDAP" was selected for Customer::AuthModule, the user identifier must be specified.

这个设置默认没有激活。

默认值:

```
$Self->{'Customer::AuthModule::LDAP::UID'} = 'uid';
```

### **Customer::AuthModule::LDAP::GroupDN**

If "LDAP" was selected for Customer::AuthModule, you can check if the user is allowed to authenticate because he is in a posixGroup, e.g. user needs to be in a group xyz to use OTRS. Specify the group, who may access the system.

这个设置默认没有激活。

默认值:

```
$Self->{'Customer::AuthModule::LDAP::GroupDN'} =  
'cn=otrsallow,ou=posixGroups,dc=example,dc=com';
```

### **Customer::AuthModule::LDAP::AccessAttr**

If "LDAP" was selected for Customer::AuthModule, you can specify access attributes here.

这个设置默认没有激活。

默认值:

```
$Self->{'Customer::AuthModule::LDAP::AccessAttr'} = 'memberUid';
```

### **Customer::AuthModule::LDAP::UserAttr**

If "LDAP" was selected for Customer::AuthModule, user attributes can be specified. For LDAP posixGroups use UID, for non LDAP posixGroups use full user DN.

这个设置默认没有激活。

默认值:

```
$Self->{'Customer::AuthModule::LDAP::UserAttr'} = 'UID';
```

### **Customer::AuthModule::LDAP::SearchUserDN**

If "LDAP" was selected for Customer::AuthModule and your users have only anonymous access to the LDAP tree, but you want to search through the data, you can do this with a user who has access to the LDAP directory. Specify the username for this special user here.

这个设置默认没有激活。

默认值:

```
$Self->{'Customer::AuthModule::LDAP::SearchUserDN'} =  
'cn=binduser,ou=users,dc=example,dc=com';
```

### Customer::AuthModule::LDAP::SearchUserPw

If "LDAP" was selected for Customer::AuthModule and your users have only anonymous access to the LDAP tree, but you want to search through the data, you can do this with a user who has access to the LDAP directory. Specify the password for this special user here.

这个设置默认没有激活。

默认值:

```
$Self->{'Customer::AuthModule::LDAP::SearchUserPw'} = 'some_password';
```

### Customer::AuthModule::LDAP::AlwaysFilter

If "LDAP" was selected, you can add a filter to each LDAP query, e.g. (mail=\*), (objectclass=user) or (!objectclass=computer).

这个设置默认没有激活。

默认值:

```
$Self->{'Customer::AuthModule::LDAP::AlwaysFilter'} = '(!objectclass=computer)';
```

### Customer::AuthModule::LDAP::UserSuffix

If "LDAP" was selected for Customer::AuthModule and if you want to add a suffix to every customer login name, specify it here, e. g. you just want to write the username user but in your LDAP directory exists user@domain.

这个设置默认没有激活。

默认值:

```
$Self->{'Customer::AuthModule::LDAP::UserSuffix'} = '@domain.com';
```

### Customer::AuthModule::LDAP::Params

If "LDAP" was selected for Customer::AuthModule and special parameters are needed for the Net::LDAP perl module, you can specify them here. See "perldoc Net::LDAP" for more information about the parameters.

这个设置默认没有激活。

默认值:

```
$Self->{'Customer::AuthModule::LDAP::Params'} = {  
'async' => '0',  
'port' => '389',  
'timeout' => '120',  
'version' => '3'  
};
```

### Customer::AuthModule::LDAP::Die

If "LDAP" was selected for Customer::AuthModule, you can specify if the applications will stop if e. g. a connection to a server can't be established due to network problems.

默认值:

```
$Self->{'Customer::AuthModule::LDAP::Die'} = '1';
```

### **Customer::AuthModule::Radius::Host**

If "Radius" was selected for Customer::AuthModule, the radius host must be specified.

这个设置默认没有激活。

默认值:

```
$Self->{'Customer::AuthModule::Radius::Host'} = 'radiushost';
```

### **Customer::AuthModule::Radius::Password**

If "Radius" was selected for Customer::AuthModule, the password to authenticate to the radius host must be specified.

这个设置默认没有激活。

默认值:

```
$Self->{'Customer::AuthModule::Radius::Password'} = 'radiussecret';
```

### **Customer::AuthModule::Radius::Die**

If "Radius" was selected for Customer::AuthModule, you can specify if the applications will stop if e. g. a connection to a server can't be established due to network problems.

默认值:

```
$Self->{'Customer::AuthModule::Radius::Die'} = '1';
```

## **Framework → Frontend::Customer::Auth::TwoFactor**

### **Customer::AuthTwoFactorModule**

Defines the two-factor module to authenticate customers.

这个设置默认没有激活。

默认值:

```
$Self->{'Customer::AuthTwoFactorModule'} =  
'Kernel::System::CustomerAuth::TwoFactor::GoogleAuthenticator';
```

### **Customer::AuthTwoFactorModule::SecretPreferencesKey**

Defines the customer preferences key where the shared secret key is stored.

不能使这个设置失效。

默认值:

```
$Self->{'Customer::AuthTwoFactorModule::SecretPreferencesKey'} =  
'UserGoogleAuthenticatorSecretKey';
```

### **Customer::AuthTwoFactorModule::AllowEmptySecret**

Defines if customers should be allowed to login if they have no shared secret stored in their preferences and therefore are not using two-factor authentication.

默认值:

```
$Self->{'Customer::AuthTwoFactorModule::AllowEmptySecret'} = '1';
```

### Customer::AuthTwoFactorModule::AllowPreviousToken

Defines if the previously valid token should be accepted for authentication. This is slightly less secure but gives users 30 seconds more time to enter their one-time password.

默认值:

```
$Self->{'Customer::AuthTwoFactorModule::AllowPreviousToken'} = '1';
```

## Framework → Frontend::Customer::ModuleMetaHead

### CustomerFrontend::HeaderMetaModule###1-Refresh

Defines the module to generate code for periodic page reloads.

不能使这个设置失效。

默认值:

```
$Self->{'CustomerFrontend::HeaderMetaModule'}->{'1-Refresh'} = {
  'Module' => 'Kernel::Output::HTML::HeaderMeta::Refresh'
};
```

## Framework → Frontend::Customer::ModuleNotify

### CustomerFrontend::NotifyModule###1-OTRSBusiness

Defines the module to display a notification in different interfaces on different occasions for OTRS Business Solution™.

不能使这个设置失效。

默认值:

```
$Self->{'CustomerFrontend::NotifyModule'}->{'1-OTRSBusiness'} = {
  'Module' => 'Kernel::Output::HTML::Notification::CustomerOTRSBusiness'
};
```

### CustomerFrontend::NotifyModule###1-ShowAgentOnline

Defines the module that shows the currently logged in agents in the customer interface.

这个设置默认没有激活。

默认值:

```
$Self->{'CustomerFrontend::NotifyModule'}->{'1-ShowAgentOnline'} = {
  'Module' => 'Kernel::Output::HTML::Notification::AgentOnline',
  'ShowEmail' => '1'
};
```

### CustomerFrontend::NotifyModule###1-ShowCustomerOnline

Defines the module that shows the currently logged in customers in the customer interface.

这个设置默认没有激活。

默认值:

```
$Self->{'CustomerFrontend::NotifyModule'}->{'1-ShowCustomerOnline'} = {
```



```
'Module' => 'Kernel::Output::HTML::Notification::CustomerOnline',
'ShowEmail' => '1'
};
```

### CustomerFrontend::NotifyModule###6-CustomerSystemMaintenance-Check

Defines the module to display a notification in the agent interface, if the agent is logged in while having system maintenance active.

不能使这个设置失效。

默认值:

```
$Self->{'CustomerFrontend::NotifyModule'}->{'6-CustomerSystemMaintenance-Check'} = {
  'Module' => 'Kernel::Output::HTML::Notification::CustomerSystemMaintenanceCheck'
};
```

## Framework → Frontend::Customer::ModuleRegistration

### CustomerFrontend::Module###Logout

Frontend module registration for the customer interface.

默认值:

```
$Self->{'CustomerFrontend::Module'}->{'Logout'} = {
  'Description' => 'Logout of customer panel.',
  'NavBarName' => '',
  'Title' => ''
};
```

### CustomerFrontend::Module###CustomerPreferences

Frontend module registration for the customer interface.

默认值:

```
$Self->{'CustomerFrontend::Module'}->{'CustomerPreferences'} = {
  'Description' => 'Customer preferences.',
  'NavBarName' => '',
  'Title' => 'Preferences'
};
```

### CustomerFrontend::Module###CustomerAccept

Frontend module registration for the customer interface.

默认值:

```
$Self->{'CustomerFrontend::Module'}->{'CustomerAccept'} = {
  'Description' => 'To accept login information, such as an EULA or license.',
  'NavBarName' => '',
  'Title' => 'Info'
};
```

### CustomerFrontend::Module###PictureUpload

Frontend module registration for the customer interface.

不能使这个设置失效。

默认值:

```
$Self->{'CustomerFrontend::Module'}->{'PictureUpload'} = {
  'Description' => 'Picture upload module.',
  'NavBarName' => 'Ticket',
};
```

```
'Title' => 'Picture-Upload'
};
```

## Framework → Frontend::Customer::Preferences

### CustomerPreferences

Defines the parameters for the customer preferences table.

不能使这个设置失效。

默认值:

```
$Self->{'CustomerPreferences'} = {
  'Module' => 'Kernel::System::CustomerUser::Preferences::DB',
  'Params' => {
    'Table' => 'customer_preferences',
    'TableKey' => 'preferences_key',
    'TableUserID' => 'user_id',
    'TableValue' => 'preferences_value'
  }
};
```

### CustomerPreferencesView

Sets the order of the different items in the customer preferences view.

不能使这个设置失效。

默认值:

```
$Self->{'CustomerPreferencesView'} = [
  'User Profile',
  'Other Settings'
];
```

### CustomerPreferencesGroups###Password

Defines all the parameters for this item in the customer preferences. 'PasswordRegExp' allows to match passwords against a regular expression. Define the minimum number of characters using 'PasswordMinSize'. Define if at least 2 lowercase and 2 uppercase letter characters are needed by setting the appropriate option to '1'. 'PasswordMin2Characters' defines if the password needs to contain at least 2 letter characters (set to 0 or 1). 'PasswordNeedDigit' controls the need of at least 1 digit (set to 0 or 1 to control).

默认值:

```
$Self->{'CustomerPreferencesGroups'}->{'Password'} = {
  'Active' => '1',
  'Area' => 'Customer',
  'Column' => 'Other Settings',
  'Desc' => 'Set a new password by filling in your current password and a new one.',
  'Label' => 'Change password',
  'Module' => 'Kernel::Output::HTML::Preferences::Password',
  'PasswordMin2Characters' => '0',
  'PasswordMin2Lower2UpperCharacters' => '0',
  'PasswordMinSize' => '0',
  'PasswordNeedDigit' => '0',
  'PasswordRegExp' => '',
  'Prio' => '1000'
};
```

### CustomerPreferencesGroups###GoogleAuthenticatorSecretKey

Defines the config parameters of this item, to be shown in the preferences view.

默认值:

```
$Self->{'CustomerPreferencesGroups'}->{'GoogleAuthenticatorSecretKey'} = {  
  'Active' => '0',  
  'Block' => 'Input',  
  'Column' => 'Other Settings',  
  'Desc' => 'Enter your shared secret to enable two factor authentication.',  
  'Key' => 'Shared Secret',  
  'Label' => 'Google Authenticator',  
  'Module' => 'Kernel::Output::HTML::Preferences::Generic',  
  'PrefKey' => 'UserGoogleAuthenticatorSecretKey',  
  'Prio' => '1100'  
};
```

### CustomerPreferencesGroups###Language

Defines all the parameters for this item in the customer preferences.

默认值:

```
$Self->{'CustomerPreferencesGroups'}->{'Language'} = {  
  'Active' => '1',  
  'Column' => 'User Profile',  
  'Desc' => 'Select the main interface language.',  
  'Key' => 'Language',  
  'Label' => 'Interface language',  
  'Module' => 'Kernel::Output::HTML::Preferences::Language',  
  'PrefKey' => 'UserLanguage',  
  'Prio' => '2000'  
};
```

### CustomerPreferencesGroups###Theme

Defines all the parameters for this item in the customer preferences.

默认值:

```
$Self->{'CustomerPreferencesGroups'}->{'Theme'} = {  
  'Active' => '0',  
  'Column' => 'User Profile',  
  'Desc' => 'Select your preferred theme for OTRS.',  
  'Key' => 'Frontend theme.',  
  'Label' => 'Theme',  
  'Module' => 'Kernel::Output::HTML::Preferences::Theme',  
  'PrefKey' => 'UserTheme',  
  'Prio' => '1000'  
};
```

### CustomerPreferencesGroups###TimeZone

Defines all the parameters for this item in the customer preferences.

默认值:

```
$Self->{'CustomerPreferencesGroups'}->{'TimeZone'} = {  
  'Active' => '1',  
  'Column' => 'User Profile',  
  'Desc' => 'This is a description for TimeZone on Customer side.',  
  'Key' => 'Time Zone',  
  'Label' => 'Time Zone',  
  'Module' => 'Kernel::Output::HTML::Preferences::TimeZone',  
  'PrefKey' => 'UserTimeZone',  
  'Prio' => '5000'  
};
```

### CustomerPreferencesGroups###PGP

Defines all the parameters for this item in the customer preferences.

默认值:

```
$Self->{'CustomerPreferencesGroups'}->{'PGP'} = {
  'Active' => '1',
  'Column' => 'Other Settings',
  'Desc' => 'Upload your PGP key.',
  'Key' => 'PGP Key',
  'Label' => 'PGP Key',
  'Module' => 'Kernel::Output::HTML::Preferences::PGP',
  'PrefKey' => 'UserPGPKey',
  'Prio' => '10000'
};
```

### CustomerPreferencesGroups###SMIME

Defines all the parameters for this item in the customer preferences.

默认值:

```
$Self->{'CustomerPreferencesGroups'}->{'SMIME'} = {
  'Active' => '1',
  'Column' => 'Other Settings',
  'Desc' => 'Upload your S/MIME certificate.',
  'Key' => 'S/MIME Certificate',
  'Label' => 'S/MIME Certificate',
  'Module' => 'Kernel::Output::HTML::Preferences::SMIME',
  'PrefKey' => 'UserSMIMEKey',
  'Prio' => '11000'
};
```

## Framework → Frontend::Public

### PublicFrontend::FooterLinks

Defines the shown links in the footer area of the customer and public interface of this OTRS system. The value in "Key" is the external URL, the value in "Content" is the shown label.

这个设置默认没有激活。

默认值:

```
$Self->{'PublicFrontend::FooterLinks'} = {
  'https://www.otrs.com' => 'OTRS Homepage'
};
```

### PublicFrontend::CommonParam###Action

Defines the default value for the action parameter for the public frontend. The action parameter is used in the scripts of the system.

不能使这个设置失效。

默认值:

```
$Self->{'PublicFrontend::CommonParam'}->{'Action'} = 'PublicDefault';
```

## Framework → Frontend::Public::ModuleRegistration

### PublicFrontend::Module###PublicDefault

Frontend module registration for the public interface.

默认值:

```
$Self->{'PublicFrontend::Module'}->{'PublicDefault'} = {
  'Description' => 'PublicDefault',
};
```

```
'NavBarName' => '',  
'Title' => 'PublicDefault'  
};
```

### **PublicFrontend::Module###PublicRepository**

Frontend module registration for the public interface.

默认值:

```
$Self->{'PublicFrontend::Module'}->{'PublicRepository'} = {  
  'Description' => 'PublicRepository',  
  'NavBarName' => '',  
  'Title' => 'PublicRepository'  
};
```

### **PublicFrontend::Module###PublicSupportDataCollector**

Frontend module registration for the public interface.

默认值:

```
$Self->{'PublicFrontend::Module'}->{'PublicSupportDataCollector'} = {  
  'Description' => 'PublicSupportDataCollector',  
  'NavBarName' => '',  
  'Title' => 'PublicSupportDataCollector'  
};
```

## **Framework → SystemMaintenance**

### **SystemMaintenance::TimeNotifyUpcomingMaintenance**

Sets the minutes a notification is shown for notice about upcoming system maintenance period.

默认值:

```
$Self->{'SystemMaintenance::TimeNotifyUpcomingMaintenance'} = '30';
```

### **SystemMaintenance::IsActiveDefaultNotification**

Sets the default message for the notification is shown on a running system maintenance period.

默认值:

```
$Self->{'SystemMaintenance::IsActiveDefaultNotification'} = 'We are performing  
scheduled maintenance.';
```

### **SystemMaintenance::IsActiveDefaultLoginMessage**

Sets the default message for the login screen on Agent and Customer interface, it's shown when a running system maintenance period is active.

默认值:

```
$Self->{'SystemMaintenance::IsActiveDefaultLoginMessage'} = 'We are performing  
scheduled maintenance. We should be back online shortly.';
```

### **SystemMaintenance::IsActiveDefaultLoginErrorMessage**

Sets the default error message for the login screen on Agent and Customer interface, it's shown when a running system maintenance period is active.

默认值:

```
$Self->{'SystemMaintenance::IsActiveDefaultLoginErrorMessage'} = 'We are performing scheduled maintenance. Login is temporarily not available.';
```

## 5. GenericInterface

### GenericInterface → Core::CustomerCompany

#### CustomerCompany::EventModulePost###1000-GenericInterface

Performs the configured action for each event (as an Invoker) for each configured Webservice.

默认值:

```
$Self->{'CustomerCompany::EventModulePost'}->{'1000-GenericInterface'} = {
  'Event' => '',
  'Module' => 'Kernel::GenericInterface::Event::Handler',
  'Transaction' => '1'
};
```

### GenericInterface → Core::CustomerUser

#### CustomerUser::EventModulePost###1000-GenericInterface

Performs the configured action for each event (as an Invoker) for each configured Webservice.

默认值:

```
$Self->{'CustomerUser::EventModulePost'}->{'1000-GenericInterface'} = {
  'Event' => '',
  'Module' => 'Kernel::GenericInterface::Event::Handler',
  'Transaction' => '1'
};
```

### GenericInterface → Core::DynamicField

#### DynamicField::EventModulePost###1000-GenericInterface

Performs the configured action for each event (as an Invoker) for each configured Webservice.

默认值:

```
$Self->{'DynamicField::EventModulePost'}->{'1000-GenericInterface'} = {
  'Event' => '',
  'Module' => 'Kernel::GenericInterface::Event::Handler',
  'Transaction' => '1'
};
```

### GenericInterface → Core::Package

#### Package::EventModulePost###1000-GenericInterface

Performs the configured action for each event (as an Invoker) for each configured Webservice.

默认值:

```
$Self->{'Package::EventModulePost'}->{'1000-GenericInterface'} = {
  'Event' => '',
  'Module' => 'Kernel::GenericInterface::Event::Handler',
  'Transaction' => '1'
};
```

```
};
```

## GenericInterface → Core::Queue

### Queue::EventModulePost###1000-GenericInterface

Performs the configured action for each event (as an Invoker) for each configured Webservice.

默认值:

```
$Self->{'Queue::EventModulePost'}->{'1000-GenericInterface'} = {
  'Event' => '',
  'Module' => 'Kernel::GenericInterface::Event::Handler',
  'Transaction' => '1'
};
```

## GenericInterface → Core::Ticket

### Ticket::EventModulePost###999-GenericInterface

Performs the configured action for each event (as an Invoker) for each configured Webservice.

默认值:

```
$Self->{'Ticket::EventModulePost'}->{'999-GenericInterface'} = {
  'Event' => '',
  'Module' => 'Kernel::GenericInterface::Event::Handler',
  'Transaction' => '1'
};
```

## GenericInterface → Frontend::Admin::ModuleRegistration

### Frontend::Module###AdminGenericInterfaceDebugger

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminGenericInterfaceDebugger'} = {
  'Description' => 'Admin',
  'Group' => [
    'admin'
  ],
  'Loader' => {
    'CSS' => [
      'Core.Agent.Admin.GenericInterface.css'
    ],
    'JavaScript' => [
      'Core.Agent.Admin.GenericInterfaceDebugger.js'
    ]
  },
  'Title' => 'GenericInterface Debugger GUI'
};
```

### Frontend::Module###AdminGenericInterfaceWebservice

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminGenericInterfaceWebservice'} = {
  'Description' => 'Admin',
  'Group' => [
    'admin'
  ]
};
```

```

],
'Loader' => {
  'CSS' => [
    'Core.Agent.Admin.GenericInterface.css'
  ],
  'JavaScript' => [
    'Core.Agent.Admin.GenericInterfaceWebservice.js'
  ]
},
'NavBarModule' => {
  'Block' => 'System',
  'Description' => 'Create and manage web services.',
  'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
  'Name' => 'Web Services',
  'Prio' => '1000'
},
'NavBarName' => 'Admin',
'Title' => 'GenericInterface Web Service GUI'
};

```

### Frontend::Module###AdminGenericInterfaceTransportHTTPSOAP

Frontend module registration for the agent interface.

默认值:

```

$self->{'Frontend::Module'}->{'AdminGenericInterfaceTransportHTTPSOAP'} = {
  'Description' => 'Admin',
  'Group' => [
    'admin'
  ],
  'Loader' => {
    'CSS' => [
      'Core.Agent.Admin.GenericInterface.css',
      'Core.Agent.SortedTree.css'
    ],
    'JavaScript' => [
      'Core.Agent.SortedTree.js'
    ]
  },
  'Title' => 'GenericInterface TransportHTTPSOAP GUI'
};

```

### Frontend::Module###AdminGenericInterfaceTransportHTTPREST

Frontend module registration for the agent interface.

默认值:

```

$self->{'Frontend::Module'}->{'AdminGenericInterfaceTransportHTTPREST'} = {
  'Description' => 'Admin',
  'Group' => [
    'admin'
  ],
  'Loader' => {
    'CSS' => [
      'Core.Agent.Admin.GenericInterface.css'
    ]
  },
  'Title' => 'GenericInterface TransportHTTPREST GUI'
};

```

### Frontend::Module###AdminGenericInterfaceWebserviceHistory

Frontend module registration for the agent interface.

默认值:

```

$self->{'Frontend::Module'}->{'AdminGenericInterfaceWebserviceHistory'} = {
  'Description' => 'Admin',

```



```
'Group' => [
  'admin'
],
'Loader' => {
  'CSS' => [
    'Core.Agent.Admin.GenericInterface.css'
  ],
  'JavaScript' => [
    'Core.Agent.Admin.GenericInterfaceWebserviceHistory.js'
  ]
},
'Title' => 'GenericInterface Webservice History GUI'
};
```

### Frontend::Module###AdminGenericInterfaceOperationDefault

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminGenericInterfaceOperationDefault'} = {
  'Description' => 'Admin',
  'Group' => [
    'admin'
  ],
  'Loader' => {
    'CSS' => [
      'Core.Agent.Admin.GenericInterface.css'
    ],
    'JavaScript' => [
      'Core.Agent.Admin.GenericInterfaceOperation.js'
    ]
  },
  'Title' => 'GenericInterface Operation GUI'
};
```

### Frontend::Module###AdminGenericInterfaceInvokerDefault

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminGenericInterfaceInvokerDefault'} = {
  'Description' => 'Admin',
  'Group' => [
    'admin'
  ],
  'Loader' => {
    'CSS' => [
      'Core.Agent.Admin.GenericInterface.css'
    ],
    'JavaScript' => [
      'Core.Agent.Admin.GenericInterfaceInvoker.js'
    ]
  },
  'Title' => 'GenericInterface Invoker GUI'
};
```

### Frontend::Module###AdminGenericInterfaceMappingSimple

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminGenericInterfaceMappingSimple'} = {
  'Description' => 'Admin',
  'Group' => [
    'admin'
  ],
  'Loader' => {
```

```
'CSS' => [
  'Core.Agent.Admin.GenericInterface.css'
],
'JavaScript' => [
  'Core.Agent.Admin.GenericInterfaceMappingSimple.js'
]
},
'Title' => 'GenericInterface Webservice Mapping GUI'
};
```

### Frontend::Module###AdminGenericInterfaceMappingXSLT

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminGenericInterfaceMappingXSLT'} = {
  'Description' => 'Admin',
  'Group' => [
    'admin'
  ],
  'Loader' => {
    'CSS' => [
      'Core.Agent.Admin.GenericInterface.css'
    ]
  },
  'Title' => 'GenericInterface Webservice Mapping GUI'
};
```

### GenericInterface

### GenericInterface::Invoker::ModuleRegistration

### GenericInterface::Invoker::Module###Test::Test

GenericInterface module registration for the invoker layer.

这个设置默认没有激活。

默认值:

```
$Self->{'GenericInterface::Invoker::Module'}->{'Test::Test'} = {
  'ConfigDialog' => 'AdminGenericInterfaceInvokerDefault',
  'Controller' => 'Test',
  'Name' => 'Test'
};
```

### GenericInterface::Invoker::Module###Test::TestSimple

GenericInterface module registration for the invoker layer.

这个设置默认没有激活。

默认值:

```
$Self->{'GenericInterface::Invoker::Module'}->{'Test::TestSimple'} = {
  'ConfigDialog' => 'AdminGenericInterfaceInvokerDefault',
  'Controller' => 'Test',
  'Name' => 'TestSimple'
};
```

### GenericInterface

### GenericInterface::Mapping::ModuleRegistration

### GenericInterface::Mapping::Module###Test

GenericInterface module registration for the mapping layer.

这个设置默认没有激活。

默认值:

```
$Self->{'GenericInterface::Mapping::Module'}->{'Test'} = {  
  'ConfigDialog' => ''  
};
```

### **GenericInterface::Mapping::Module###Simple**

GenericInterface module registration for the mapping layer.

默认值:

```
$Self->{'GenericInterface::Mapping::Module'}->{'Simple'} = {  
  'ConfigDialog' => 'AdminGenericInterfaceMappingSimple'  
};
```

### **GenericInterface::Mapping::Module###XSLT**

GenericInterface module registration for the mapping layer.

默认值:

```
$Self->{'GenericInterface::Mapping::Module'}->{'XSLT'} = {  
  'ConfigDialog' => 'AdminGenericInterfaceMappingXSLT'  
};
```

## **GenericInterface**

### **GenericInterface::Operation::ModuleRegistration**

#### **GenericInterface::Operation::Module###Test::Test**

GenericInterface module registration for the operation layer.

这个设置默认没有激活。

默认值:

```
$Self->{'GenericInterface::Operation::Module'}->{'Test::Test'} = {  
  'ConfigDialog' => 'AdminGenericInterfaceOperationDefault',  
  'Controller' => 'Test',  
  'Name' => 'Test'  
};
```

#### **GenericInterface::Operation::Module###Session::SessionCreate**

GenericInterface module registration for the operation layer.

默认值:

```
$Self->{'GenericInterface::Operation::Module'}->{'Session::SessionCreate'} = {  
  'ConfigDialog' => 'AdminGenericInterfaceOperationDefault',  
  'Controller' => 'Session',  
  'Name' => 'SessionCreate'  
};
```

#### **GenericInterface::Operation::Module###Ticket::TicketCreate**

GenericInterface module registration for the operation layer.

默认值:

```
$Self->{'GenericInterface::Operation::Module'}->{'Ticket::TicketCreate'} = {  
  'ConfigDialog' => 'AdminGenericInterfaceOperationDefault',
```

```
'Controller' => 'Ticket',  
'Name' => 'TicketCreate'  
};
```

### **GenericInterface::Operation::Module###Ticket::TicketUpdate**

GenericInterface module registration for the operation layer.

默认值:

```
$Self->{'GenericInterface::Operation::Module'}->{'Ticket::TicketUpdate'} = {  
  'ConfigDialog' => 'AdminGenericInterfaceOperationDefault',  
  'Controller' => 'Ticket',  
  'Name' => 'TicketUpdate'  
};
```

### **GenericInterface::Operation::Module###Ticket::TicketGet**

GenericInterface module registration for the operation layer.

默认值:

```
$Self->{'GenericInterface::Operation::Module'}->{'Ticket::TicketGet'} = {  
  'ConfigDialog' => 'AdminGenericInterfaceOperationDefault',  
  'Controller' => 'Ticket',  
  'Name' => 'TicketGet'  
};
```

### **GenericInterface::Operation::Module###Ticket::TicketSearch**

GenericInterface module registration for the operation layer.

默认值:

```
$Self->{'GenericInterface::Operation::Module'}->{'Ticket::TicketSearch'} = {  
  'ConfigDialog' => 'AdminGenericInterfaceOperationDefault',  
  'Controller' => 'Ticket',  
  'Name' => 'TicketGet'  
};
```

## **GenericInterface**

### **GenericInterface::Operation::ResponseLoggingMaxSize**

#### **GenericInterface::Operation::ResponseLoggingMaxSize**

Defines the maximum size in KiloByte of GenericInterface responses that get logged to the gi\_debugger\_entry\_content table.

不能使这个设置失效。

默认值:

```
$Self->{'GenericInterface::Operation::ResponseLoggingMaxSize'} = '200';
```

### **GenericInterface → GenericInterface::Operation::TicketCreate**

#### **GenericInterface::Operation::TicketCreate###ArticleType**

Defines the default type of the article for this operation.

不能使这个设置失效。

默认值:

```
$Self->{'GenericInterface::Operation::TicketCreate'}->{'ArticleType'} = 'webrequest';
```

### **GenericInterface::Operation::TicketCreate###HistoryType**

Defines the history type for this operation, which gets used for ticket history in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'GenericInterface::Operation::TicketCreate'}->{'HistoryType'} = 'NewTicket';
```

### **GenericInterface::Operation::TicketCreate###HistoryComment**

Defines the history comment for this operation, which gets used for ticket history in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'GenericInterface::Operation::TicketCreate'}->{'HistoryComment'} = '%GenericInterface Create';
```

### **GenericInterface::Operation::TicketCreate###AutoResponseType**

Defines the default auto response type of the article for this operation.

不能使这个设置失效。

默认值:

```
$Self->{'GenericInterface::Operation::TicketCreate'}->{'AutoResponseType'} = 'auto reply';
```

## **GenericInterface → GenericInterface::Operation::TicketSearch**

### **GenericInterface::Operation::TicketSearch###SearchLimit**

Maximum number of tickets to be displayed in the result of this operation.

不能使这个设置失效。

默认值:

```
$Self->{'GenericInterface::Operation::TicketSearch'}->{'SearchLimit'} = '500';
```

### **GenericInterface::Operation::TicketSearch###SortBy::Default**

Defines the default ticket attribute for ticket sorting of the ticket search result of this operation.

不能使这个设置失效。

默认值:

```
$Self->{'GenericInterface::Operation::TicketSearch'}->{'SortBy::Default'} = 'Age';
```

### **GenericInterface::Operation::TicketSearch###Order::Default**

Defines the default ticket order in the ticket search result of the this operation. Up: oldest on top. Down: latest on top.

不能使这个设置失效。

默认值:

```
$Self->{'GenericInterface::Operation::TicketSearch'}->{'Order::Default'} = 'Down';
```

## GenericInterface → GenericInterface::Operation::TicketUpdate

### GenericInterface::Operation::TicketUpdate###ArticleType

Defines the default type of the article for this operation.

不能使这个设置失效。

默认值:

```
$Self->{'GenericInterface::Operation::TicketUpdate'}->{'ArticleType'} = 'webrequest';
```

### GenericInterface::Operation::TicketUpdate###HistoryType

Defines the history type for this operation, which gets used for ticket history in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'GenericInterface::Operation::TicketUpdate'}->{'HistoryType'} = 'AddNote';
```

### GenericInterface::Operation::TicketUpdate###HistoryComment

Defines the history comment for this operation, which gets used for ticket history in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'GenericInterface::Operation::TicketUpdate'}->{'HistoryComment'} = '%GenericInterface Note';
```

### GenericInterface::Operation::TicketUpdate###AutoResponseType

Defines the default auto response type of the article for this operation.

不能使这个设置失效。

默认值:

```
$Self->{'GenericInterface::Operation::TicketUpdate'}->{'AutoResponseType'} = 'auto follow up';
```

## GenericInterface



### GenericInterface::Transport::ModuleRegistration

#### GenericInterface::Transport::Module###HTTP::SOAP

GenericInterface module registration for the transport layer.

默认值:

```
$Self->{'GenericInterface::Transport::Module'}->{'HTTP::SOAP'} = {
  'ConfigDialog' => 'AdminGenericInterfaceTransportHTTPSOAP',
  'Name' => 'SOAP',
  'Protocol' => 'HTTP'
};
```

#### GenericInterface::Transport::Module###HTTP::REST

GenericInterface module registration for the transport layer.

默认值:

```
$Self->{'GenericInterface::Transport::Module'}->{'HTTP::REST'} = {
  'ConfigDialog' => 'AdminGenericInterfaceTransportHTTPREST',
  'Name' => 'REST',
  'Protocol' => 'HTTP'
};
```

### **GenericInterface::Transport::Module###HTTP::Test**

GenericInterface module registration for the transport layer.

这个设置默认没有激活。

默认值:

```
$Self->{'GenericInterface::Transport::Module'}->{'HTTP::Test'} = {
  'ConfigDialog' => 'AdminGenericInterfaceTransportHTTPTest',
  'Name' => 'Test',
  'Protocol' => 'HTTP'
};
```

## **GenericInterface → GenericInterface::Webservice**

### **GenericInterface::WebserviceConfig::CacheTTL**

Cache time in seconds for the web service config backend.

不能使这个设置失效。

默认值:

```
$Self->{'GenericInterface::WebserviceConfig::CacheTTL'} = '86400';
```

### **GenericInterface::Operation::Common::CachedAuth::AgentCacheTTL**

Cache time in seconds for agent authentication in the GenericInterface.

不能使这个设置失效。

默认值:

```
$Self->{'GenericInterface::Operation::Common::CachedAuth::AgentCacheTTL'} = '300';
```

### **GenericInterface::Operation::Common::CachedAuth::CustomerCacheTTL**

Cache time in seconds for customer authentication in the GenericInterface.

不能使这个设置失效。

默认值:

```
$Self->{'GenericInterface::Operation::Common::CachedAuth::CustomerCacheTTL'} = '300';
```

# **6. ProcessManagement**

## **ProcessManagement → Core**

### **Process::DynamicFieldProcessManagementProcessID**

This option defines the dynamic field in which a Process Management process entity id is stored.

不能使这个设置失效。

默认值:

```
$Self->{'Process::DynamicFieldProcessManagementProcessID'} =  
'ProcessManagementProcessID';
```

### **Process::DynamicFieldProcessManagementActivityID**

This option defines the dynamic field in which a Process Management activity entity id is stored.

不能使这个设置失效。

默认值:

```
$Self->{'Process::DynamicFieldProcessManagementActivityID'} =  
'ProcessManagementActivityID';
```

### **Process::DefaultQueue**

This option defines the process tickets default queue.

不能使这个设置失效。

默认值:

```
$Self->{'Process::DefaultQueue'} = 'Raw';
```

### **Process::DefaultState**

This option defines the process tickets default state.

不能使这个设置失效。

默认值:

```
$Self->{'Process::DefaultState'} = 'new';
```

### **Process::DefaultLock**

This option defines the process tickets default lock.

不能使这个设置失效。

默认值:

```
$Self->{'Process::DefaultLock'} = 'unlock';
```

### **Process::DefaultPriority**

This option defines the process tickets default priority.

不能使这个设置失效。

默认值:

```
$Self->{'Process::DefaultPriority'} = '3 normal';
```

### **Process::Entity::Prefix**

Default ProcessManagement entity prefixes for entity IDs that are automatically generated.

不能使这个设置失效。

默认值:



```
$Self->{'Process::Entity::Prefix'} = {
  'Activity' => 'A',
  'ActivityDialog' => 'AD',
  'Process' => 'P',
  'Transition' => 'T',
  'TransitionAction' => 'TA'
};
```

### Process::CacheTTL

Cache time in seconds for the DB process backend.

不能使这个设置失效。

默认值:

```
$Self->{'Process::CacheTTL'} = '3600';
```

### Process::NavBarOutput::CacheTTL

Cache time in seconds for the ticket process navigation bar output module.

不能使这个设置失效。

默认值:

```
$Self->{'Process::NavBarOutput::CacheTTL'} = '900';
```

## ProcessManagement → Core::Ticket

### Ticket::EventModulePost###998-TicketProcessTransitions

Event module registration. For more performance you can define a trigger event (e.g. Event => TicketCreate).

默认值:

```
$Self->{'Ticket::EventModulePost'}->{'998-TicketProcessTransitions'} = {
  'Event' => '',
  'Module' => 'Kernel::System::Ticket::Event::TicketProcessTransitions',
  'Transaction' => '1'
};
```

## ProcessManagement → Core::Transition

### ProcessManagement::Transition::Debug::Enabled

If enabled debugging information for transitions is logged.

不能使这个设置失效。

默认值:

```
$Self->{'ProcessManagement::Transition::Debug::Enabled'} = '0';
```

### ProcessManagement::Transition::Debug::LogPriority

Defines the priority in which the information is logged and presented.

这个设置默认没有激活。

默认值:

```
$Self->{'ProcessManagement::Transition::Debug::LogPriority'} = 'debug';
```

## ProcessManagement::Transition::Debug::Filter###00-Default

Filter for debugging Transitions. Note: More filters can be added in the format <OTRS\_TICKET\_Attribute> e.g. <OTRS\_TICKET\_Priority>.

这个设置默认没有激活。

默认值:

```
$Self->{'ProcessManagement::Transition::Debug::Filter'}->{'00-Default'} = {
  '<OTRS_TICKET_TicketNumber>' => '',
  'TransitionEntityID' => ''
};
```

## ProcessManagement → DynamicFields::Driver::Registration

### DynamicFields::Driver###ProcessID

DynamicField backend registration.

默认值:

```
$Self->{'DynamicFields::Driver'}->{'ProcessID'} = {
  'ConfigDialog' => 'AdminDynamicFieldText',
  'DisabledAdd' => '1',
  'DisplayName' => 'ProcessID',
  'Module' => 'Kernel::System::DynamicField::Driver::ProcessManagement::ProcessID'
};
```

### DynamicFields::Driver###ActivityID

DynamicField backend registration.

默认值:

```
$Self->{'DynamicFields::Driver'}->{'ActivityID'} = {
  'ConfigDialog' => 'AdminDynamicFieldText',
  'DisabledAdd' => '1',
  'DisplayName' => 'ActivityID',
  'Module' => 'Kernel::System::DynamicField::Driver::ProcessManagement::ActivityID'
};
```

## ProcessManagement → Frontend::Admin::ModuleRegistration

### Frontend::Module###AdminProcessManagement

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminProcessManagement'} = {
  'Description' => 'This module is part of the admin area of OTRS.',
  'Group' => [
    'admin'
  ],
  'Loader' => {
    'CSS' => [
      'Core.Agent.Admin.ProcessManagement.css',
      'Core.AllocationList.css'
    ],
    'JavaScript' => [
      'thirdparty/jsplumb-1.6.4/jsplumb.js',
      'thirdparty/farahey-0.5/farahey.js',
      'thirdparty/jsplumb-labelspacer/label-spacer.js',
      'Core.Agent.Admin.ProcessManagement.js',
      'Core.Agent.Admin.ProcessManagement.Canvas.js',
      'Core.UI.AllocationList.js'
    ]
  }
};
```

```

    ]
  },
  'NavBarModule' => {
    'Block' => 'System',
    'Description' => 'Configure Processes.',
    'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
    'Name' => 'Process Management',
    'Prio' => '750'
  },
  },
  'NavBarName' => 'Admin',
  'Title' => 'Process Management'
};

```

### Frontend::Module###AdminProcessManagementActivity

Frontend module registration for the agent interface.

默认值:

```

$self->{'Frontend::Module'}->{'AdminProcessManagementActivity'} = {
  'Description' => 'This module is part of the admin area of OTRS.',
  'Group' => [
    'admin'
  ],
  'Loader' => {
    'CSS' => [
      'Core.Agent.Admin.ProcessManagement.css',
      'Core.AllocationList.css'
    ],
    'JavaScript' => [
      'Core.Agent.Admin.ProcessManagement.js',
      'Core.UI.AllocationList.js'
    ]
  },
  'Title' => 'Process Management Activity GUI'
};

```

### Frontend::Module###AdminProcessManagementActivityDialog

Frontend module registration for the agent interface.

默认值:

```

$self->{'Frontend::Module'}->{'AdminProcessManagementActivityDialog'} = {
  'Description' => 'This module is part of the admin area of OTRS.',
  'Group' => [
    'admin'
  ],
  'Loader' => {
    'CSS' => [
      'Core.Agent.Admin.ProcessManagement.css',
      'Core.AllocationList.css'
    ],
    'JavaScript' => [
      'Core.Agent.Admin.ProcessManagement.js',
      'Core.UI.AllocationList.js'
    ]
  },
  'Title' => 'Process Management Activity Dialog GUI'
};

```

### Frontend::Module###AdminProcessManagementTransition

Frontend module registration for the agent interface.

默认值:

```

$self->{'Frontend::Module'}->{'AdminProcessManagementTransition'} = {
  'Description' => 'This module is part of the admin area of OTRS.',
  'Group' => [

```

```

    'admin'
  ],
  'Loader' => {
    'CSS' => [
      'Core.Agent.Admin.ProcessManagement.css'
    ],
    'JavaScript' => [
      'Core.Agent.Admin.ProcessManagement.js'
    ]
  },
  'Title' => 'Process Management Transition GUI'
};

```

### Frontend::Module###AdminProcessManagementTransitionAction

Frontend module registration for the agent interface.

默认值:

```

$self->{'Frontend::Module'}->{'AdminProcessManagementTransitionAction'} = {
  'Description' => 'This module is part of the admin area of OTRS.',
  'Group' => [
    'admin'
  ],
  'Loader' => {
    'CSS' => [
      'Core.Agent.Admin.ProcessManagement.css'
    ],
    'JavaScript' => [
      'Core.Agent.Admin.ProcessManagement.js'
    ]
  },
  'Title' => 'Process Management Transition Action GUI'
};

```

### Frontend::Module###AdminProcessManagementPath

Frontend module registration for the agent interface.

默认值:

```

$self->{'Frontend::Module'}->{'AdminProcessManagementPath'} = {
  'Description' => 'This module is part of the admin area of OTRS.',
  'Group' => [
    'admin'
  ],
  'Loader' => {
    'CSS' => [
      'Core.Agent.Admin.ProcessManagement.css',
      'Core.AllocationList.css'
    ],
    'JavaScript' => [
      'Core.Agent.Admin.ProcessManagement.js',
      'Core.UI.AllocationList.js'
    ]
  },
  'Title' => 'Process Management Path GUI'
};

```

## ProcessManagement → Frontend::Agent::Dashboard

### DashboardBackend###0140-RunningTicketProcess

Parameters for the dashboard backend of the running process tickets overview of the agent interface. "Limit" is the number of entries shown by default. "Group" is used to restrict the access to the plugin (e. g. Group: admin;group1;group2;). "Default" determines if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" is the cache time in minutes for the plugin.

默认值:

```
$Self->{'DashboardBackend'}->{'0140-RunningTicketProcess'} = {
  'Attributes' => 'StateType=new;StateType=open;StateType=pending
reminder;StateType=pending auto',
  'Block' => 'ContentLarge',
  'CacheTTLLocal' => '0.5',
  'Default' => '0',
  'DefaultColumns' => {
    'Age' => '2',
    'Changed' => '1',
    'CustomerID' => '1',
    'CustomerName' => '1',
    'CustomerUserID' => '1',
    'DynamicField_ProcessManagementActivityID' => '2',
    'DynamicField_ProcessManagementProcessID' => '2',
    'EscalationResponseTime' => '1',
    'EscalationSolutionTime' => '1',
    'EscalationTime' => '1',
    'EscalationUpdateTime' => '1',
    'Lock' => '1',
    'Owner' => '1',
    'PendingTime' => '1',
    'Priority' => '1',
    'Queue' => '1',
    'Responsible' => '1',
    'SLA' => '1',
    'Service' => '1',
    'State' => '1',
    'TicketNumber' => '2',
    'Title' => '2',
    'Type' => '1'
  },
  'Description' => 'All tickets with a reminder set where the reminder date has been
reached',
  'Group' => '',
  'IsProcessWidget' => '1',
  'Limit' => '10',
  'Module' => 'Kernel::Output::HTML::Dashboard::TicketGeneric',
  'Permission' => 'rw',
  'Time' => 'UntilTime',
  'Title' => 'Running Process Tickets'
};
```

## ProcessManagement → Frontend::Agent::ModuleRegistration

### Frontend::Module###AgentTicketProcess

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AgentTicketProcess'} = {
  'Description' => 'Create new process ticket.',
  'Loader' => {
    'CSS' => [
      'Core.Agent.TicketProcess.css'
    ],
    'JavaScript' => [
      'Core.Agent.CustomerSearch.js',
      'Core.Agent.TicketAction.js',
      'Core.Agent.TicketProcess.js'
    ]
  },
  'NavBar' => [
    {
      'AccessKey' => 'p',
      'Block' => '',
      'Description' => 'Create New process ticket.',
      'Link' => 'Action=AgentTicketProcess',
      'LinkOption' => ''
    }
  ]
};
```

```

    'Name' => 'New process ticket',
    'NavBar' => 'Ticket',
    'Prio' => '220',
    'Type' => ''
  }
],
'NavBarName' => 'Ticket',
'Title' => 'New process ticket'
};

```

## ProcessManagement → Frontend::Agent::NavBarModule

### Frontend::NavBarModule###1-TicketProcesses

Frontend module registration (disable ticket processes screen if no process available).

默认值:

```

$self->{'Frontend::NavBarModule'}->{'1-TicketProcesses'} = {
  'Module' => 'Kernel::Output::HTML::NavBar::AgentTicketProcess'
};

```

## ProcessManagement → Frontend::Agent::Ticket::MenuModule

### Ticket::Frontend::MenuModule###480-Process

Shows a link in the menu to enroll a ticket into a process in the ticket zoom view of the agent interface.

默认值:

```

$self->{'Ticket::Frontend::MenuModule'}->{'480-Process'} = {
  'Action' => 'AgentTicketProcess',
  'Cluster' => '',
  'Description' => 'Enroll process for this ticket',
  'Link' => 'Action=AgentTicketProcess;IsProcessEnroll=1;TicketID=[% Data.TicketID |
html %]',
  'Module' => 'Kernel::Output::HTML::TicketMenu::Process',
  'Name' => 'Process',
  'PopupType' => 'TicketAction',
  'Target' => ''
};

```

## ProcessManagement → Frontend::Agent::Ticket::ViewProcess

### Ticket::Frontend::AgentTicketProcess###StateType

Determines the next possible ticket states, for process tickets in the agent interface.

不能使这个设置失效。

默认值:

```

$self->{'Ticket::Frontend::AgentTicketProcess'}->{'StateType'} = [
  'new',
  'open',
  'pending auto',
  'pending reminder',
  'closed'
];

```

### Ticket::Frontend::AgentTicketProcess::CustomerIDReadOnly

Controls if CustomerID is read-only in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketProcess::CustomerIDReadOnly'} = '1';
```

## ProcessManagement → Frontend::Agent::Ticket::ViewZoom

### Ticket::Frontend::AgentTicketZoom###ProcessDisplay

Display settings to override defaults for Process Tickets.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketZoom'}->{'ProcessDisplay'} = {
  'NavBarName' => 'Processes',
  'WidgetTitle' => 'Process Information'
};
```

### Ticket::Frontend::AgentTicketZoom###ProcessWidgetDynamicFieldGroups

Dynamic fields groups for process widget. The key is the name of the group, the value contains the fields to be shown. Example: 'Key => My Group', 'Content: Name\_X, NameY'.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketZoom'}->{'ProcessWidgetDynamicFieldGroups'} = {};
```

### Ticket::Frontend::AgentTicketZoom###ProcessWidgetDynamicField

Dynamic fields shown in the process widget in ticket zoom screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketZoom'}->{'ProcessWidgetDynamicField'} = {};
```

## ProcessManagement → Frontend::Customer::ModuleRegistration

### CustomerFrontend::Module###CustomerTicketProcess

Frontend module registration for the customer interface.

默认值:

```
$Self->{'CustomerFrontend::Module'}->{'CustomerTicketProcess'} = {
  'Description' => 'Process Ticket.',
  'Loader' => {
    'CSS' => [
      'Core.Customer.TicketProcess.css'
    ]
  },
  'NavBar' => [
    {
      'AccessKey' => 'o',
      'Block' => '',
      'Description' => 'Create new process ticket.',
      'Link' => 'Action=CustomerTicketProcess',
      'LinkOption' => ''
    }
  ]
};
```

```

    'Name' => 'New process ticket',
    'NavBar' => 'Ticket',
    'Prio' => '220',
    'Type' => 'Submenu'
  }
],
'NavBarName' => 'Ticket',
'Title' => 'Process ticket'
};

```

## ProcessManagement → Frontend::Customer::NavBarModule

### CustomerFrontend::NavBarModule###10-CustomerTicketProcesses

Frontend module registration (disable ticket processes screen if no process available) for Customer.

默认值:

```

$self->{'CustomerFrontend::NavBarModule'}->{'10-CustomerTicketProcesses'} = {
  'Module' => 'Kernel::Output::HTML::NavBar::CustomerTicketProcess'
};

```

## ProcessManagement → Frontend::Customer::Ticket::ViewProcess

### Ticket::Frontend::CustomerTicketProcess###StateType

Determines the next possible ticket states, for process tickets in the customer interface.

不能使这个设置失效。

默认值:

```

$self->{'Ticket::Frontend::CustomerTicketProcess'}->{'StateType'} = [
  'new',
  'open'
];

```

# 7. Ticket

## Ticket → Core::CustomerCompany

### CustomerCompany::EventModulePost###110-UpdateTickets

Event module that updates tickets after an update of the Customer.

默认值:

```

$self->{'CustomerCompany::EventModulePost'}->{'110-UpdateTickets'} = {
  'Event' => 'CustomerCompanyUpdate',
  'Module' => 'Kernel::System::CustomerCompany::Event::TicketUpdate',
  'Transaction' => '0'
};

```

### CustomerUser::EventModulePost###120-UpdateTickets

Event module that updates tickets after an update of the Customer User.

默认值:

```

$self->{'CustomerUser::EventModulePost'}->{'120-UpdateTickets'} = {
  'Event' => 'CustomerUserUpdate',

```



```
'Module' => 'Kernel::System::CustomerUser::Event::TicketUpdate',
'Transaction' => '0'
};
```

## Ticket → Core::FulltextSearch

### Ticket::SearchIndexModule

Helps to extend your articles full-text search (From, To, Cc, Subject and Body search). Runtime will do full-text searches on live data (it works fine for up to 50.000 tickets). StaticDB will strip all articles and will build an index after article creation, increasing fulltext searches about 50%. To create an initial index use "bin/otrs.Console.pl Maint::Ticket::FulltextIndexRebuild".

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::SearchIndexModule'} =
'Kernel::System::Ticket::ArticleSearchIndex::RuntimeDB';
```

### Ticket::SearchIndex::WarnOnStopWordUsage

Display a warning and prevent search when using stop words within fulltext search.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::SearchIndex::WarnOnStopWordUsage'} = '0';
```

### Ticket::SearchIndex::Attribute

Basic fulltext index settings. Execute "bin/otrs.Console.pl Maint::Ticket::FulltextIndexRebuild" in order to generate a new index.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::SearchIndex::Attribute'} = {
'WordCountMax' => '1000',
'WordLengthMax' => '30',
'WordLengthMin' => '3'
};
```

### Ticket::SearchIndex::Filters

Fulltext index regex filters to remove parts of the text.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::SearchIndex::Filters'} = [
'[,\\&\\<\\>\\?\"\\!\\*\\|\\;\\[\\]\\(\\)\\+\\$\\^=]',
'^[\\':.][\\':.]$',
'^[\\^\\w]+$'
];
```

### Ticket::SearchIndex::StopWords###en

English stop words for fulltext index. These words will be removed from the search index.

**默认值:**

```
$Self->{'Ticket::SearchIndex::StopWords'}->{'en'} = [  
  'a',  
  'about',  
  'above',  
  'after',  
  'again',  
  'against',  
  'all',  
  'am',  
  'an',  
  'and',  
  'any',  
  'are',  
  'aren\'t',  
  'as',  
  'at',  
  'be',  
  'because',  
  'been',  
  'before',  
  'being',  
  'below',  
  'between',  
  'both',  
  'but',  
  'by',  
  'can\'t',  
  'cannot',  
  'could',  
  'couldn\'t',  
  'did',  
  'didn\'t',  
  'do',  
  'does',  
  'doesn\'t',  
  'doing',  
  'don\'t',  
  'down',  
  'during',  
  'each',  
  'few',  
  'for',  
  'from',  
  'further',  
  'had',  
  'hadn\'t',  
  'has',  
  'hasn\'t',  
  'have',  
  'haven\'t',  
  'having',  
  'he',  
  'he\'d',  
  'he\'ll',  
  'he\'s',  
  'her',  
  'here',  
  'here\'s',  
  'hers',  
  'herself',  
  'him',  
  'himself',  
  'his',  
  'how',  
  'how\'s',  
  'i',  
  'i\'d',  
  'i\'ll',  
  'i\'m',  
  'i\'ve',
```

'if',  
'in',  
'into',  
'is',  
'isn\'t',  
'it',  
'it\'s',  
'its',  
'itself',  
'let\'s',  
'me',  
'more',  
'most',  
'mustn\'t',  
'my',  
'myself',  
'no',  
'nor',  
'not',  
'of',  
'off',  
'on',  
'once',  
'only',  
'or',  
'other',  
'ought',  
'our',  
'ours',  
'ourselves',  
'out',  
'over',  
'own',  
'same',  
'shan\'t',  
'she',  
'she\'d',  
'she\'ll',  
'she\'s',  
'should',  
'shouldn\'t',  
'so',  
'some',  
'such',  
'than',  
'that',  
'that\'s',  
'the',  
'their',  
'theirs',  
'them',  
'themselves',  
'then',  
'there',  
'there\'s',  
'these',  
'they',  
'they\'d',  
'they\'ll',  
'they\'re',  
'they\'ve',  
'this',  
'those',  
'through',  
'to',  
'too',  
'under',  
'until',  
'up',  
'very',  
'was',  
'wasn\'t',

```
'we',  
'we\'d',  
'we\'ll',  
'we\'re',  
'we\'ve',  
'were',  
'weren\'t',  
'what',  
'what\'s',  
'when',  
'when\'s',  
'where',  
'where\'s',  
'which',  
'while',  
'who',  
'who\'s',  
'whom',  
'why',  
'why\'s',  
'with',  
'won\'t',  
'would',  
'wouldn\'t',  
'you',  
'you\'d',  
'you\'ll',  
'you\'re',  
'you\'ve',  
'your',  
'yours',  
'yourself',  
'yourselves'  
];
```

### **Ticket::SearchIndex::StopWords###de**

German stop words for fulltext index. These words will be removed from the search index.

默认值:

```
$Self->{'Ticket::SearchIndex::StopWords'}->{'de'} = [  
'aber',  
'als',  
'am',  
'an',  
'auch',  
'auf',  
'aus',  
'bei',  
'bin',  
'bis',  
'bist',  
'da',  
'dadurch',  
'daher',  
'darum',  
'das',  
'daß',  
'dass',  
'dein',  
'deine',  
'dem',  
'den',  
'der',  
'des',  
'dessen',  
'deshalb',  
'die',  
'dies',  
'dieser',
```

'dieses',  
'doch',  
'dort',  
'du',  
'durch',  
'ein',  
'eine',  
'einem',  
'einen',  
'einer',  
'eines',  
'er',  
'es',  
'euer',  
'eure',  
'für',  
'hatte',  
'hatten',  
'hattest',  
'hattet',  
'hier',  
'hinter',  
'ich',  
'ihr',  
'ihre',  
'im',  
'in',  
'ist',  
'ja',  
'jede',  
'jedem',  
'jeden',  
'jeder',  
'jedes',  
'jener',  
'jenes',  
'jetzt',  
'kann',  
'kannst',  
'können',  
'könnt',  
'machen',  
'mein',  
'meine',  
'mit',  
'muß',  
'mußt',  
'musst',  
'müssen',  
'mußt',  
'nach',  
'nachdem',  
'nein',  
'nicht',  
'nun',  
'oder',  
'seid',  
'sein',  
'seine',  
'sich',  
'sie',  
'sind',  
'soll',  
'sollen',  
'sollst',  
'sollt',  
'sonst',  
'soweit',  
'sowie',  
'und',  
'unser',  
'unsere',

```
'unter',
'vom',
'von',
'vor',
'wann',
'warum',
'was',
'weiter',
'weitere',
'wenn',
'wer',
'werde',
'werden',
'werdet',
'weshalb',
'wie',
'wieder',
'wieso',
'wir',
'wird',
'wirst',
'wo',
'woher',
'wohin',
'zu',
'zum',
'zur',
'über'
];
```

### **Ticket::SearchIndex::StopWords###nl**

Dutch stop words for fulltext index. These words will be removed from the search index.

默认值:

```
$Self->{'Ticket::SearchIndex::StopWords'}->{'nl'} = [
'de',
'zijn',
'een',
'en',
'in',
'je',
'het',
'van',
'op',
'ze',
'hebben',
'het',
'hij',
'niet',
'met',
'er',
'dat',
'die',
'te',
'wat',
'voor',
'naar',
'gaan',
'kunnen',
'zeggen',
'dat',
'maar',
'aan',
'veel',
'zijn',
'worden',
'uit',
'ook',
'komen',
```

'als',  
'om',  
'moeten',  
'we',  
'doen',  
'bij',  
'goed',  
'haar',  
'dan',  
'nog',  
'of',  
'maken',  
'zo',  
'wel',  
'mijn',  
'zien',  
'over',  
'willen',  
'staan',  
'door',  
'kijken',  
'zullen',  
'heel',  
'nu',  
'weten',  
'zitten',  
'hem',  
'schrijven',  
'vinden',  
'woord',  
'hoe',  
'geen',  
'dit',  
'mens',  
'al',  
'jij',  
'ander',  
'groot',  
'waar',  
'maar',  
'weer',  
'kind',  
'me',  
'vragen',  
'een',  
'denken',  
'twee',  
'horen',  
'iets',  
'deze',  
'krijgen',  
'ons',  
'zich',  
'lezen',  
'hun',  
'welk',  
'zin',  
'laten',  
'mogen',  
'hier',  
'jullie',  
'toch',  
'geven',  
'jaar',  
'tegen',  
'al',  
'eens',  
'echt',  
'houden',  
'alleen',  
'lopen',  
'mee',

```
'ja',  
'roepen',  
'tijd',  
'dag',  
'elkaar',  
'even',  
'lang',  
'land',  
'liggen',  
'waarom',  
'zetten',  
'vader',  
'laat',  
'beginnen',  
'blijven',  
'nee',  
'moeder',  
'huis',  
'nou',  
'na',  
'af',  
'keer',  
'dus',  
'tot',  
'vertellen',  
'wie',  
'net',  
'jou',  
'les',  
'want',  
'man',  
'nieuw',  
'elk',  
'tekst',  
'omdat',  
'gebruiken',  
'u'  
];
```

### **Ticket::SearchIndex::StopWords###es**

Spanish stop words for fulltext index. These words will be removed from the search index.

默认值:

```
$Self->{'Ticket::SearchIndex::StopWords'}->{'es'} = [  
'un',  
'una',  
'unas',  
'unos',  
'uno',  
'sobre',  
'todo',  
'también',  
'tras',  
'otro',  
'algún',  
'alguno',  
'alguna',  
'algunos',  
'algunas',  
'ser',  
'es',  
'soy',  
'eres',  
'somos',  
'sois',  
'estoy',  
'esta',  
'estamos',  
'estais',
```



'estan',  
'como',  
'en',  
'para',  
'atras',  
'porque',  
'por qué',  
'estado',  
'estaba',  
'ante',  
'antes',  
'siendo',  
'ambos',  
'pero',  
'por',  
'poder',  
'puede',  
'puedo',  
'podemos',  
'podeis',  
'pueden',  
'fui',  
'fue',  
'fuimos',  
'fueron',  
'hacer',  
'hago',  
'hace',  
'hacemos',  
'haceis',  
'hacen',  
'cada',  
'fin',  
'incluso',  
'primero',  
'desde',  
'conseguir',  
'consigo',  
'consigue',  
'consigues',  
'conseguimos',  
'consiguen',  
'ir',  
'voy',  
'va',  
'vamos',  
'vais',  
'van',  
'vaya',  
'gueno',  
'ha',  
'tener',  
'tengo',  
'tiene',  
'tenemos',  
'teneis',  
'tienen',  
'el',  
'la',  
'lo',  
'las',  
'los',  
'su',  
'aqui',  
'mio',  
'tuyo',  
'ellos',  
'ellas',  
'nos',  
'nosotros',  
'vosotros',  
'vosotras',

'si',  
'dentro',  
'solo',  
'solamente',  
'saber',  
'sabes',  
'sabe',  
'sabemos',  
'sabeis',  
'saben',  
'ultimo',  
'largo',  
'bastante',  
'haces',  
'muchos',  
'aquellos',  
'aquellas',  
'sus',  
'entonces',  
'tiempo',  
'verdad',  
'verdadero',  
'verdadera',  
'cierto',  
'ciertos',  
'cierta',  
'ciertas',  
'intentar',  
'intento',  
'intenta',  
'intentas',  
'intentamos',  
'intentais',  
'intentan',  
'dos',  
'bajo',  
'arriba',  
'encima',  
'usar',  
'uso',  
'usas',  
'usa',  
'usamos',  
'usais',  
'usan',  
'emplear',  
'empleo',  
'empleas',  
'emplean',  
'empleamos',  
'empleais',  
'valor',  
'muy',  
'era',  
'eras',  
'eramos',  
'eran',  
'modo',  
'bien',  
'cual',  
'cuando',  
'donde',  
'mientras',  
'quien',  
'con',  
'entre',  
'sin',  
'trabajo',  
'trabajar',  
'trabajas',  
'trabaja',  
'trabajamos',

```
'trabajais',  
'trabajan',  
'podria',  
'podrias',  
'podriamos',  
'podrian',  
'podriais',  
'yo',  
'aquel'  
];
```

### **Ticket::SearchIndex::StopWords###fr**

French stop words for fulltext index. These words will be removed from the search index.

默认值:

```
$Self->{'Ticket::SearchIndex::StopWords'}->{'fr'} = [  
'alors',  
'au',  
'aucuns',  
'aussi',  
'autre',  
'avant',  
'avec',  
'avoir',  
'bon',  
'car',  
'ce',  
'cela',  
'ces',  
'ceux',  
'chaque',  
'ci',  
'comme',  
'comment',  
'dans',  
'des',  
'du',  
'dedans',  
'dehors',  
'depuis',  
'deux',  
'devrait',  
'doit',  
'donc',  
'dos',  
'droite',  
'début',  
'elle',  
'elles',  
'en',  
'encore',  
'essai',  
'est',  
'et',  
'eu',  
'fait',  
'faites',  
'fois',  
'font',  
'force',  
'haut',  
'hors',  
'ici',  
'il',  
'ils',  
'je',  
'juste',  
'la',  
'le',
```

'les',  
'leur',  
'là',  
'ma',  
'maintenant',  
'mais',  
'mes',  
'mine',  
'moins',  
'mon',  
'mot',  
'même',  
'ni',  
'nommés',  
'notre',  
'nous',  
'nouveaux',  
'ou',  
'où',  
'par',  
'parce',  
'parole',  
'pas',  
'personnes',  
'peut',  
'peu',  
'pièce',  
'plupart',  
'pour',  
'pourquoi',  
'quand',  
'que',  
'quel',  
'quelle',  
'quelles',  
'quels',  
'qui',  
'sa',  
'sans',  
'ses',  
'seulement',  
'si',  
'sien',  
'son',  
'sont',  
'sous',  
'soyez',  
'sujet',  
'sur',  
'ta',  
'tandis',  
'tellement',  
'tels',  
'tes',  
'ton',  
'tous',  
'tout',  
'trop',  
'très',  
'tu',  
'valeur',  
'voie',  
'voient',  
'vont',  
'votre',  
'vous',  
'vu',  
'ça',  
'étaient',  
'état',  
'étions',  
'été',

```
'être'  
];
```

### **Ticket::SearchIndex::StopWords###it**

Italian stop words for fulltext index. These words will be removed from the search index.

默认值:

```
$Self->{'Ticket::SearchIndex::StopWords'}->{'it'} = [  
'a',  
'adesso',  
'ai',  
'al',  
'alla',  
'allo',  
'allora',  
'altre',  
'altri',  
'altro',  
'anche',  
'ancora',  
'avere',  
'aveva',  
'avevano',  
'ben',  
'buono',  
'che',  
'chi',  
'cinque',  
'comprare',  
'con',  
'consecutivi',  
'consecutivo',  
'cosa',  
'cui',  
'da',  
'del',  
'della',  
'dello',  
'dentro',  
'deve',  
'devo',  
'di',  
'doppio',  
'due',  
'e',  
'ecco',  
'fare',  
'fine',  
'fino',  
'fra',  
'gente',  
'giu',  
'ha',  
'hai',  
'hanno',  
'ho',  
'il',  
'indietro',  
'invece',  
'io',  
'la',  
'lavoro',  
'le',  
'lei',  
'lo',  
'loro',  
'lui',  
'lungo',  
'ma',
```

'me',  
'meglio',  
'molta',  
'molti',  
'molto',  
'nei',  
'nella',  
'no',  
'noi',  
'nome',  
'nostro',  
'nove',  
'nuovi',  
'nuovo',  
'o',  
'oltre',  
'ora',  
'otto',  
'peggio',  
'pero',  
'persone',  
'piu',  
'poco',  
'primo',  
'promesso',  
'qua',  
'quarto',  
'quasi',  
'quattro',  
'quello',  
'questo',  
'qui',  
'quindi',  
'quinto',  
'rispetto',  
'sara',  
'secondo',  
'sei',  
'sembra',  
'sembrava',  
'senza',  
'sette',  
'sia',  
'siamo',  
'siete',  
'solo',  
'sono',  
'sopra',  
'soprattutto',  
'sotto',  
'stati',  
'stato',  
'stesso',  
'su',  
'subito',  
'sul',  
'sulla',  
'tanto',  
'te',  
'tempo',  
'terzo',  
'tra',  
'tre',  
'triplo',  
'ultimo',  
'un',  
'una',  
'uno',  
'va',  
'vai',  
'voi',  
'volte',

```
'vostro'
];
```

### **Ticket::SearchIndex::StopWords###Custom**

Customizable stop words for fulltext index. These words will be removed from the search index.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::SearchIndex::StopWords'}->{'Custom'} = [
  'MyStopWord'
];
```

### **Ticket::EventModulePost###098-ArticleSearchIndex**

Builds an article index right after the article's creation.

默认值:

```
$Self->{'Ticket::EventModulePost'}->{'098-ArticleSearchIndex'} = {
  'Event' => '(ArticleCreate|ArticleUpdate)',
  'Module' => 'Kernel::System::Ticket::Event::ArticleSearchIndex'
};
```

## **Ticket → Core::LinkObject**

### **LinkObject::PossibleLink###0200**

Links 2 tickets with a "Normal" type link.

默认值:

```
$Self->{'LinkObject::PossibleLink'}->{'0200'} = {
  'Object1' => 'Ticket',
  'Object2' => 'Ticket',
  'Type' => 'Normal'
};
```

### **LinkObject::PossibleLink###0201**

Links 2 tickets with a "ParentChild" type link.

默认值:

```
$Self->{'LinkObject::PossibleLink'}->{'0201'} = {
  'Object1' => 'Ticket',
  'Object2' => 'Ticket',
  'Type' => 'ParentChild'
};
```

### **LinkObject::IgnoreLinkedTicketStateTypes**

Defines, which tickets of which ticket state types should not be listed in linked ticket lists.

默认值:

```
$Self->{'LinkObject::IgnoreLinkedTicketStateTypes'} = [
  'merged',
  'removed'
];
```

## Ticket → Core::PostMaster

### PostmasterMaxEmails

Maximal auto email responses to own email-address a day (Loop-Protection).

不能使这个设置失效。

默认值:

```
$Self->{'PostmasterMaxEmails'} = '40';
```

### PostMasterMaxEmailSize

Maximal size in KBytes for mails that can be fetched via POP3/POP3S/IMAP/IMAPS (KBytes).

不能使这个设置失效。

默认值:

```
$Self->{'PostMasterMaxEmailSize'} = '16384';
```

### PostMasterReconnectMessage

The maximum number of mails fetched at once before reconnecting to the server.

不能使这个设置失效。

默认值:

```
$Self->{'PostMasterReconnectMessage'} = '20';
```

### LoopProtectionModule

Default loop protection module.

不能使这个设置失效。

默认值:

```
$Self->{'LoopProtectionModule'} = 'Kernel::System::PostMaster::LoopProtection::DB';
```

### LoopProtectionLog

Path for the log file (it only applies if "FS" was selected for LoopProtectionModule and it is mandatory).

不能使这个设置失效。

默认值:

```
$Self->{'LoopProtectionLog'} = '<OTRS_CONFIG_Home>/var/Log/LoopProtection';
```

### PostmasterAutoHTML2Text

Converts HTML mails into text messages.

不能使这个设置失效。

默认值:

```
$Self->{'PostmasterAutoHTML2Text'} = '1';
```



### PostmasterUserID

Specifies user id of the postmaster data base.

不能使这个设置失效。

默认值:

```
$Self->{'PostmasterUserID'} = '1';
```

### PostmasterDefaultQueue

Defines the postmaster default queue.

不能使这个设置失效。

默认值:

```
$Self->{'PostmasterDefaultQueue'} = 'Raw';
```

### PostmasterDefaultPriority

Defines the default priority of new tickets.

不能使这个设置失效。

默认值:

```
$Self->{'PostmasterDefaultPriority'} = '3 normal';
```

### PostmasterDefaultState

Defines the default state of new tickets.

不能使这个设置失效。

默认值:

```
$Self->{'PostmasterDefaultState'} = 'new';
```

### PostmasterFollowUpState

Defines the state of a ticket if it gets a follow-up.

不能使这个设置失效。

默认值:

```
$Self->{'PostmasterFollowUpState'} = 'open';
```

### PostmasterFollowUpStateClosed

Defines the state of a ticket if it gets a follow-up and the ticket was already closed.

这个设置默认没有激活。

默认值:

```
$Self->{'PostmasterFollowUpStateClosed'} = 'open';
```

### PostmasterFollowUpOnUnlockAgentNotifyOnlyToOwner

Sends agent follow-up notification only to the owner, if a ticket is unlocked (the default is to send the notification to all agents).

不能使这个设置失效。

默认值:

```
$Self->{'PostmasterFollowUpOnUnlockAgentNotifyOnlyToOwner'} = '0';
```

### PostmasterHeaderFieldCount

Defines the number of header fields in frontend modules for add and update postmaster filters. It can be up to 99 fields.

不能使这个设置失效。

默认值:

```
$Self->{'PostmasterHeaderFieldCount'} = '12';
```

### PostmasterX-Header

Defines all the X-headers that should be scanned.

不能使这个设置失效。

默认值:

```
$Self->{'PostmasterX-Header'} = [  
  'From',  
  'To',  
  'Cc',  
  'Reply-To',  
  'ReplyTo',  
  'Subject',  
  'Message-ID',  
  'Message-Id',  
  'Resent-To',  
  'Resent-From',  
  'Precedence',  
  'Mailing-List',  
  'List-Id',  
  'List-Archive',  
  'Errors-To',  
  'References',  
  'In-Reply-To',  
  'Auto-Submitted',  
  'X-Loop',  
  'X-Spam',  
  'X-Spam-Flag',  
  'X-Spam-Level',  
  'X-Spam-Score',  
  'X-Spam-Status',  
  'X-No-Loop',  
  'X-Priority',  
  'Importance',  
  'X-Mailer',  
  'User-Agent',  
  'Organization',  
  'X-Original-To',  
  'Delivered-To',  
  'Envelope-To',  
  'X-Envelope-To',  
  'Return-Path',  
  'X-OTRS-Owner',  
  'X-OTRS-OwnerID',  
  'X-OTRS-Responsible',  
  'X-OTRS-ResponsibleID',  
  'X-OTRS-Loop',  
  'X-OTRS-Priority',  
  'X-OTRS-Queue',  
  'X-OTRS-Lock',  
  'X-OTRS-Ignore',
```

```
'X-OTRS-State',
'X-OTRS-State-PendingTime',
'X-OTRS-Type',
'X-OTRS-Service',
'X-OTRS-SLA',
'X-OTRS-CustomerNo',
'X-OTRS-CustomerUser',
'X-OTRS-SenderType',
'X-OTRS-ArticleType',
'X-OTRS-FollowUp-Priority',
'X-OTRS-FollowUp-Queue',
'X-OTRS-FollowUp-Lock',
'X-OTRS-FollowUp-State',
'X-OTRS-FollowUp-State-PendingTime',
'X-OTRS-FollowUp-Type',
'X-OTRS-FollowUp-Service',
'X-OTRS-FollowUp-SLA',
'X-OTRS-FollowUp-SenderType',
'X-OTRS-FollowUp-ArticleType',
'X-OTRS-BodyDecrypted'
];
```

### PostMaster::PreFilterModule###1-Match

Module to filter and manipulate incoming messages. Block/ignore all spam email with From: noreply@ address.

这个设置默认没有激活。

默认值:

```
$Self->{'PostMaster::PreFilterModule'}->{'1-Match'} = {
  'Match' => {
    'From' => 'noreply@'
  },
  'Module' => 'Kernel::System::PostMaster::Filter::Match',
  'Set' => {
    'X-OTRS-Ignore' => 'yes'
  }
};
```

### PostMaster::PreFilterModule###2-Match

Module to filter and manipulate incoming messages. Get a 4 digit number to ticket free text, use regex in Match e. g. From => '(.+?)@.+?', and use () as [\*\*\*] in Set =>.

这个设置默认没有激活。

默认值:

```
$Self->{'PostMaster::PreFilterModule'}->{'2-Match'} = {
  'Match' => {
    'Subject' => 'SomeNumber:(\\d\\d\\d\\d)'
  },
  'Module' => 'Kernel::System::PostMaster::Filter::Match',
  'Set' => {
    'X-OTRS-DynamicField-TicketFreeKey1' => 'SomeNumber',
    'X-OTRS-DynamicField-TicketFreeText1' => '***'
  }
};
```

### PostMaster::PreFilterModule###3-NewTicketReject

Blocks all the incoming emails that do not have a valid ticket number in subject with From: @example.com address.

这个设置默认没有激活。

默认值:

```
$Self->{'PostMaster::PreFilterModule'}->{'3-NewTicketReject'} = {  
  'Match' => {  
    'From' => '@example.com'  
  },  
  'Module' => 'Kernel::System::PostMaster::Filter::NewTicketReject',  
  'Set' => {  
    'X-OTRS-Ignore' => 'yes'  
  }  
};
```

### PostMaster::PreFilterModule::NewTicketReject::Sender

Defines the sender for rejected emails.

这个设置默认没有激活。

默认值:

```
$Self->{'PostMaster::PreFilterModule::NewTicketReject::Sender'} =  
'noreply@example.com';
```

### PostMaster::PreFilterModule::NewTicketReject::Subject

Defines the subject for rejected emails.

不能使这个设置失效。

默认值:

```
$Self->{'PostMaster::PreFilterModule::NewTicketReject::Subject'} = 'Email Rejected';
```

### PostMaster::PreFilterModule::NewTicketReject::Body

Defines the body text for rejected emails.

不能使这个设置失效。

默认值:

```
$Self->{'PostMaster::PreFilterModule::NewTicketReject::Body'} = '  
Dear Customer,  
  
Unfortunately we could not detect a valid ticket number  
in your subject, so this email can\'t be processed.  
  
Please create a new ticket via the customer panel.  
  
Thanks for your help!  
  
Your Helpdesk Team  
';
```

### PostMaster::PreFilterModule###4-CMD

CMD example setup. Ignores emails where external CMD returns some output on STDOUT (email will be piped into STDIN of some.bin).

这个设置默认没有激活。

默认值:

```
$Self->{'PostMaster::PreFilterModule'}->{'4-CMD'} = {  
  'CMD' => '/usr/bin/some.bin',  
  'Module' => 'Kernel::System::PostMaster::Filter::CMD',  
  'Set' => {  
    'X-OTRS-Ignore' => 'yes'  
  }  
};
```

### PostMaster::PreFilterModule###5-SpamAssassin

Spam Assassin example setup. Ignores emails that are marked with SpamAssassin.

这个设置默认没有激活。

默认值:

```
$Self->{'PostMaster::PreFilterModule'}->{'5-SpamAssassin'} = {  
  'CMD' => '/usr/bin/spamassassin | grep -i "X-Spam-Status: yes"',  
  'Module' => 'Kernel::System::PostMaster::Filter::CMD',  
  'Set' => {  
    'X-OTRS-Ignore' => 'yes'  
  }  
};
```

### PostMaster::PreFilterModule###6-SpamAssassin

Spam Assassin example setup. Moves marked mails to spam queue.

这个设置默认没有激活。

默认值:

```
$Self->{'PostMaster::PreFilterModule'}->{'6-SpamAssassin'} = {  
  'CMD' => '/usr/bin/spamassassin | grep -i "X-Spam-Status: yes"',  
  'Module' => 'Kernel::System::PostMaster::Filter::CMD',  
  'Set' => {  
    'X-OTRS-Queue' => 'spam'  
  }  
};
```

### PostMaster::PreFilterModule###000-MatchDBSource

Module to use database filter storage.

不能使这个设置失效。

默认值:

```
$Self->{'PostMaster::PreFilterModule'}->{'000-MatchDBSource'} = {  
  'Module' => 'Kernel::System::PostMaster::Filter::MatchDBSource'  
};
```

### PostMaster::PreCreateFilterModule###000-FollowUpArticleTypeCheck

Module to check if arrived emails should be marked as email-internal (because of original forwarded internal email). ArticleType and SenderType define the values for the arrived email/article.

默认值:

```
$Self->{'PostMaster::PreCreateFilterModule'}->{'000-FollowUpArticleTypeCheck'} = {  
  'ArticleType' => 'email-internal',  
  'Module' => 'Kernel::System::PostMaster::Filter::FollowUpArticleTypeCheck',  
  'SenderType' => 'customer'  
};
```

### PostMaster::PreFilterModule###000-ExternalTicketNumberRecognition1

Recognize if a ticket is a follow-up to an existing ticket using an external ticket number.

这个设置默认没有激活。

默认值:

```
$Self->{'PostMaster::PreFilterModule'}->{'000-ExternalTicketNumberRecognition1'} = {
```

```
'ArticleType' => 'note-report',
'DynamicFieldName' => 'Name_X',
'FromAddressRegExp' => '\\s*@example.com',
'Module' => 'Kernel::System::PostMaster::Filter::ExternalTicketNumberRecognition',
'Name' => 'Some Description',
'NumberRegExp' => '\\s*Incident-(\\d.*)\\s*',
'SearchInBody' => '1',
'SearchInSubject' => '1',
'SenderType' => 'system',
'TicketStateTypes' => 'new;open'
};
```

### PostMaster::PreFilterModule###000-ExternalTicketNumberRecognition2

Recognize if a ticket is a follow-up to an existing ticket using an external ticket number.

这个设置默认没有激活。

默认值:

```
$Self->{'PostMaster::PreFilterModule'}->{'000-ExternalTicketNumberRecognition2'} = {
'ArticleType' => 'note-report',
'DynamicFieldName' => 'Name_X',
'FromAddressRegExp' => '\\s*@example.com',
'Module' => 'Kernel::System::PostMaster::Filter::ExternalTicketNumberRecognition',
'Name' => 'Some Description',
'NumberRegExp' => '\\s*Incident-(\\d.*)\\s*',
'SearchInBody' => '1',
'SearchInSubject' => '1',
'SenderType' => 'system',
'TicketStateTypes' => 'new;open'
};
```

### PostMaster::PreFilterModule###000-ExternalTicketNumberRecognition3

Recognize if a ticket is a follow-up to an existing ticket using an external ticket number.

这个设置默认没有激活。

默认值:

```
$Self->{'PostMaster::PreFilterModule'}->{'000-ExternalTicketNumberRecognition3'} = {
'ArticleType' => 'note-report',
'DynamicFieldName' => 'Name_X',
'FromAddressRegExp' => '\\s*@example.com',
'Module' => 'Kernel::System::PostMaster::Filter::ExternalTicketNumberRecognition',
'Name' => 'Some Description',
'NumberRegExp' => '\\s*Incident-(\\d.*)\\s*',
'SearchInBody' => '1',
'SearchInSubject' => '1',
'SenderType' => 'system',
'TicketStateTypes' => 'new;open'
};
```

### PostMaster::PreFilterModule###000-ExternalTicketNumberRecognition4

Recognize if a ticket is a follow-up to an existing ticket using an external ticket number.

这个设置默认没有激活。

默认值:

```
$Self->{'PostMaster::PreFilterModule'}->{'000-ExternalTicketNumberRecognition4'} = {
'ArticleType' => 'note-report',
'DynamicFieldName' => 'Name_X',
'FromAddressRegExp' => '\\s*@example.com',
'Module' => 'Kernel::System::PostMaster::Filter::ExternalTicketNumberRecognition',
'Name' => 'Some Description',
'NumberRegExp' => '\\s*Incident-(\\d.*)\\s*',
'SearchInBody' => '1',
```

```
'SearchInSubject' => '1',  
'SenderType' => 'system',  
'TicketStateTypes' => 'new;open'  
};
```

### **PostMaster::PreFilterModule###000-DecryptBody**

Module to filter encrypted bodies of incoming messages.

这个设置默认没有激活。

默认值:

```
$Self->{'PostMaster::PreFilterModule'}->{'000-DecryptBody'} = {  
  'Module' => 'Kernel::System::PostMaster::Filter::Decrypt',  
  'StoreDecryptedBody' => '0'  
};
```

### **PostMaster::PreFilterModule###000-SMIMEFetchFromCustomer**

Module to fetch customer users SMIME certificates of incoming messages.

默认值:

```
$Self->{'PostMaster::PreFilterModule'}->{'000-SMIMEFetchFromCustomer'} = {  
  'Module' => 'Kernel::System::PostMaster::Filter::SMIMEFetchFromCustomer'  
};
```

### **PostMaster::CheckFollowUpModule###0100-Subject**

Checks if an E-Mail is a followup to an existing ticket by searching the subject for a valid ticket number.

默认值:

```
$Self->{'PostMaster::CheckFollowUpModule'}->{'0100-Subject'} = {  
  'Module' => 'Kernel::System::PostMaster::FollowUpCheck::Subject'  
};
```

### **PostMaster::CheckFollowUpModule###0200-References**

Executes follow-up checks on In-Reply-To or References headers for mails that don't have a ticket number in the subject.

默认值:

```
$Self->{'PostMaster::CheckFollowUpModule'}->{'0200-References'} = {  
  'Module' => 'Kernel::System::PostMaster::FollowUpCheck::References'  
};
```

### **PostMaster::CheckFollowUpModule###0300-Body**

Executes follow-up checks on email body for mails that don't have a ticket number in the subject.

这个设置默认没有激活。

默认值:

```
$Self->{'PostMaster::CheckFollowUpModule'}->{'0300-Body'} = {  
  'Module' => 'Kernel::System::PostMaster::FollowUpCheck::Body'  
};
```

### **PostMaster::CheckFollowUpModule###0400-Attachments**

Executes follow-up checks on attachment contents for mails that don't have a ticket number in the subject.

这个设置默认没有激活。

默认值:

```
$Self->{'PostMaster::CheckFollowUpModule'}->{'0400-Attachments'} = {  
  'Module' => 'Kernel::System::PostMaster::FollowUpCheck::Attachments'  
};
```

### **PostMaster::CheckFollowUpModule###0500-RawEmail**

Executes follow-up checks on the raw source email for mails that don't have a ticket number in the subject.

这个设置默认没有激活。

默认值:

```
$Self->{'PostMaster::CheckFollowUpModule'}->{'0500-RawEmail'} = {  
  'Module' => 'Kernel::System::PostMaster::FollowUpCheck::RawEmail'  
};
```

### **PostMaster::NewTicket::AutoAssignCustomerIDForUnknownCustomers**

Controls if CustomerID is automatically copied from the sender address for unknown customers.

不能使这个设置失效。

默认值:

```
$Self->{'PostMaster::NewTicket::AutoAssignCustomerIDForUnknownCustomers'} = '1';
```

### **SendNoAutoResponseRegExp**

If this regex matches, no message will be send by the autoresponder.

不能使这个设置失效。

默认值:

```
$Self->{'SendNoAutoResponseRegExp'} = '(MAILER-DAEMON|postmaster|abuse)@.+?\\.\\.?.+?';
```

### **AutoResponseForWebTickets**

If this option is set to 'Yes', tickets created via the web interface, via Customers or Agents, will receive an autoresponse if configured. If this option is set to 'No', no autoresponses will be sent.

不能使这个设置失效。

默认值:

```
$Self->{'AutoResponseForWebTickets'} = '1';
```

## **Ticket → Core::Queue**

### **Queue::EventModulePost###130-UpdateQueue**

Event module that performs an update statement on TicketIndex to rename the queue name there if needed and if StaticDB is actually used.

默认值:

```
$Self->{'Queue::EventModulePost'}->{'130-UpdateQueue'} = {  
  'Event' => 'QueueUpdate',  
};
```



```
'Module' => 'Kernel::System::Queue::Event::TicketAcceleratorUpdate',
'Transaction' => '0'
};
```

## Ticket → Core::Stats

### Stats::DynamicObjectRegistration###Ticket

Module to generate ticket statistics.

默认值:

```
$Self->{'Stats::DynamicObjectRegistration'}->{'Ticket'} = {
'Module' => 'Kernel::System::Stats::Dynamic::Ticket'
};
```

### Stats::DynamicObjectRegistration###TicketList

Determines if the statistics module may generate ticket lists.

默认值:

```
$Self->{'Stats::DynamicObjectRegistration'}->{'TicketList'} = {
'Module' => 'Kernel::System::Stats::Dynamic::TicketList'
};
```

### Stats::DynamicObjectRegistration###TicketAccountedTime

Module to generate accounted time ticket statistics.

默认值:

```
$Self->{'Stats::DynamicObjectRegistration'}->{'TicketAccountedTime'} = {
'Module' => 'Kernel::System::Stats::Dynamic::TicketAccountedTime'
};
```

### Stats::DynamicObjectRegistration###TicketSolutionResponseTime

Module to generate ticket solution and response time statistics.

默认值:

```
$Self->{'Stats::DynamicObjectRegistration'}->{'TicketSolutionResponseTime'} = {
'Module' => 'Kernel::System::Stats::Dynamic::TicketSolutionResponseTime'
};
```

## Ticket → Core::Ticket

### Ticket::Hook

The identifier for a ticket, e.g. Ticket#, Call#, MyTicket#. The default is Ticket#.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Hook'} = 'Ticket#';
```

### Ticket::HookDivider

The divider between TicketHook and ticket number. E.g ': '.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::HookDivider'} = '';
```

### **Ticket::SubjectSize**

Max size of the subjects in an email reply and in some overview screens.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::SubjectSize'} = '100';
```

### **Ticket::SubjectRe**

The text at the beginning of the subject in an email reply, e.g. RE, AW, or AS.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::SubjectRe'} = 'Re';
```

### **Ticket::SubjectFwd**

The text at the beginning of the subject when an email is forwarded, e.g. FW, Fwd, or WG.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::SubjectFwd'} = 'Fwd';
```

### **Ticket::SubjectFormat**

The format of the subject. 'Left' means '[TicketHook#:12345] Some Subject', 'Right' means 'Some Subject [TicketHook#:12345]', 'None' means 'Some Subject' and no ticket number. In the latter case you should verify that the setting PostMaster::CheckFollowUpModule###0200-References is activated to recognize followups based on email headers.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::SubjectFormat'} = 'Left';
```

### **Ticket::MergeDynamicFields**

A list of dynamic fields that are merged into the main ticket during a merge operation. Only dynamic fields that are empty in the main ticket will be set.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::MergeDynamicFields'} = [];
```

### **Ticket::CustomQueue**

Name of custom queue. The custom queue is a queue selection of your preferred queues and can be selected in the preferences settings.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::CustomQueue'} = 'My Queues';
```

### **Ticket::CustomService**

Name of custom service. The custom service is a service selection of your preferred services and can be selected in the preferences settings.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::CustomService'} = 'My Services';
```

### **Ticket::NewArticleIgnoreSystemSender**

Ignore system sender article types (e. g. auto responses or email notifications) to be flagged as 'Unread Article' in AgentTicketZoom or expanded automatically in Large view screens.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::NewArticleIgnoreSystemSender'} = '0';
```

### **Ticket::ChangeOwnerToEveryone**

Changes the owner of tickets to everyone (useful for ASP). Normally only agent with rw permissions in the queue of the ticket will be shown.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::ChangeOwnerToEveryone'} = '0';
```

### **Ticket::Responsible**

Enables ticket responsible feature, to keep track of a specific ticket.

默认值:

```
$Self->{'Ticket::Responsible'} = '0';
```

### **Ticket::ResponsibleAutoSet**

Automatically sets the owner of a ticket as the responsible for it (if ticket responsible feature is enabled). This will only work by manually actions of the logged in user. It does not work for automated actions e.g. GenericAgent, Postmaster and GenericInterface.

默认值:

```
$Self->{'Ticket::ResponsibleAutoSet'} = '1';
```

### **Ticket::Type**

Allows defining new types for ticket (if ticket type feature is enabled).

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Type'} = '0';
```

### **Ticket::Type::Default**

Defines the default ticket type.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Type::Default'} = 'Unclassified';
```

### **Ticket::Service**

Allows defining services and SLAs for tickets (e. g. email, desktop, network, ...), and escalation attributes for SLAs (if ticket service/SLA feature is enabled).

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Service'} = '0';
```

### **Ticket::Service::KeepChildren**

Retains all services in listings even if they are children of invalid elements.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Service::KeepChildren'} = '0';
```

### **Ticket::Service::Default::UnknownCustomer**

Allows default services to be selected also for non existing customers.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Service::Default::UnknownCustomer'} = '0';
```

### **Ticket::ArchiveSystem**

Activates the ticket archive system to have a faster system by moving some tickets out of the daily scope. To search for these tickets, the archive flag has to be enabled in the ticket search.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::ArchiveSystem'} = '0';
```

### **Ticket::ArchiveSystem::RemoveSeenFlags**

Controls if the ticket and article seen flags are removed when a ticket is archived.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::ArchiveSystem::RemoveSeenFlags'} = '1';
```

### **Ticket::ArchiveSystem::RemoveTicketWatchers**

Removes the ticket watcher information when a ticket is archived.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::ArchiveSystem::RemoveTicketWatchers'} = '1';
```

### **Ticket::CustomerArchiveSystem**

Activates the ticket archive system search in the customer interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::CustomerArchiveSystem'} = '0';
```

### **Ticket::NumberGenerator**

Selects the ticket number generator module. "AutoIncrement" increments the ticket number, the SystemID and the counter are used with SystemID.counter format (e.g. 1010138, 1010139). With "Date" the ticket numbers will be generated by the current date, the SystemID and the counter. The format looks like Year.Month.Day.SystemID.counter (e.g. 200206231010138, 200206231010139). With "DateChecksum" the counter will be appended as checksum to the string of date and SystemID. The checksum will be rotated on a daily basis. The format looks like Year.Month.Day.SystemID.Counter.CheckSum (e.g. 2002070110101520, 2002070110101535). "Random" generates randomized ticket numbers in the format "SystemID.Random" (e.g. 100057866352, 103745394596).

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::NumberGenerator'} = 'Kernel::System::Ticket::Number::DateChecksum';
```

### **Ticket::NumberGenerator::CheckSystemID**

Checks the SystemID in ticket number detection for follow-ups (use "No" if SystemID has been changed after using the system).

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::NumberGenerator::CheckSystemID'} = '1';
```

### **Ticket::NumberGenerator::MinCounterSize**

Sets the minimal ticket counter size if "AutoIncrement" was selected as TicketNumberGenerator. Default is 5, this means the counter starts from 10000.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::NumberGenerator::MinCounterSize'} = '5';
```

### **Ticket::NumberGenerator::Date::UseFormattedCounter**

Enables the minimal ticket counter size (if "Date" was selected as TicketNumberGenerator).

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::NumberGenerator::Date::UseFormattedCounter'} = '0';
```

### **Ticket::CounterLog**

Log file for the ticket counter.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::CounterLog'} = '<OTRS_CONFIG_Home>/var/log/TicketCounter.log';
```

### **Ticket::IndexModule**

IndexAccelerator: to choose your backend TicketViewAccelerator module. "RuntimeDB" generates each queue view on the fly from ticket table (no performance problems up to approx. 60.000 tickets in total and 6.000 open tickets in the system). "StaticDB" is the most powerful module, it uses an extra ticket-index table that works like a view (recommended if more than 80.000 and 6.000 open tickets are stored in the system). Use the command "bin/otrs.Console.pl Maint::Ticket::QueueIndexRebuild" for initial index creation.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::IndexModule'} =  
'Kernel::System::Ticket::IndexAccelerator::RuntimeDB';
```

### **Ticket::StorageModule**

Saves the attachments of articles. "DB" stores all data in the database (not recommended for storing big attachments). "FS" stores the data on the filesystem; this is faster but the webserver should run under the OTRS user. You can switch between the modules even on a system that is already in production without any loss of data. Note: Searching for attachment names is not supported when "FS" is used.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::StorageModule'} = 'Kernel::System::Ticket::ArticleStorageDB';
```

### **Ticket::StorageModule::CheckAllBackends**

Specifies whether all storage backends should be checked when looking for attachments. This is only required for installations where some attachments are in the file system, and others in the database.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::StorageModule::CheckAllBackends'} = '0';
```

### **ArticleDir**

Specifies the directory to store the data in, if "FS" was selected for TicketStorageModule.

不能使这个设置失效。

默认值:

```
$Self->{'ArticleDir'} = '<OTRS_CONFIG_Home>/var/article';
```

### **OTRSEscalationEvents::DecayTime**

The duration in minutes after emitting an event, in which the new escalation notify and start events are suppressed.

默认值:

```
$Self->{'OTRSEscalationEvents::DecayTime'} = '1440';
```

### **Ticket::EventModulePost###100-ArchiveRestore**

Restores a ticket from the archive (only if the event is a state change to any open available state).

默认值:

```
$Self->{'Ticket::EventModulePost'}->{'100-ArchiveRestore'} = {  
  'Event' => 'TicketStateUpdate',  
  'Module' => 'Kernel::System::Ticket::Event::ArchiveRestore'  
};
```

### **Ticket::EventModulePost###110-AcceleratorUpdate**

Updates the ticket index accelerator.

默认值:

```
$Self->{'Ticket::EventModulePost'}->{'110-AcceleratorUpdate'} = {  
  'Event' => 'TicketStateUpdate|TicketQueueUpdate|TicketLockUpdate',  
  'Module' => 'Kernel::System::Ticket::Event::TicketAcceleratorUpdate'  
};
```

### **Ticket::EventModulePost###120-ForceOwnerResetOnMove**

Resets and unlocks the owner of a ticket if it was moved to another queue.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::EventModulePost'}->{'120-ForceOwnerResetOnMove'} = {  
  'Event' => 'TicketQueueUpdate',  
  'Module' => 'Kernel::System::Ticket::Event::ForceOwnerReset'  
};
```

### **Ticket::EventModulePost###130-ForceStateChangeOnLock**

Forces to choose a different ticket state (from current) after lock action. Define the current state as key, and the next state after lock action as content.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::EventModulePost'}->{'130-ForceStateChangeOnLock'} = {  
  'Event' => 'TicketLockUpdate',  
  'Module' => 'Kernel::System::Ticket::Event::ForceState',  
  'new' => 'open'  
};
```

### **Ticket::EventModulePost###140-ResponsibleAutoSet**

Automatically sets the responsible of a ticket (if it is not set yet) after the first owner update.

默认值:

```
$Self->{'Ticket::EventModulePost'}->{'140-ResponsibleAutoSet'} = {
  'Event' => 'TicketOwnerUpdate',
  'Module' => 'Kernel::System::Ticket::Event::ResponsibleAutoSet'
};
```

### **Ticket::EventModulePost###150-TicketPendingTimeReset**

Sets the PendingTime of a ticket to 0 if the state is changed to a non-pending state.

默认值:

```
$Self->{'Ticket::EventModulePost'}->{'150-TicketPendingTimeReset'} = {
  'Event' => 'TicketStateUpdate',
  'Module' => 'Kernel::System::Ticket::Event::TicketPendingTimeReset'
};
```

### **Ticket::EventModulePost###500-NotificationEvent**

Sends the notifications which are configured in the admin interface under "Ticket Notifications".

默认值:

```
$Self->{'Ticket::EventModulePost'}->{'500-NotificationEvent'} = {
  'Event' => '',
  'Module' => 'Kernel::System::Ticket::Event::NotificationEvent',
  'Transaction' => '1'
};
```

### **Ticket::EventModulePost###910-EscalationIndex**

Updates the ticket escalation index after a ticket attribute got updated.

默认值:

```
$Self->{'Ticket::EventModulePost'}->{'910-EscalationIndex'} = {
  'Event' => 'TicketSLAUpdate|TicketQueueUpdate|TicketStateUpdate|TicketCreate|
ArticleCreate',
  'Module' => 'Kernel::System::Ticket::Event::TicketEscalationIndex'
};
```

### **Ticket::EventModulePost###920-EscalationStopEvents**

Ticket event module that triggers the escalation stop events.

默认值:

```
$Self->{'Ticket::EventModulePost'}->{'920-EscalationStopEvents'} = {
  'Event' => 'TicketSLAUpdate|TicketQueueUpdate|TicketStateUpdate|ArticleCreate',
  'Module' => 'Kernel::System::Ticket::Event::TriggerEscalationStopEvents'
};
```

### **Ticket::EventModulePost###930-ForceUnlockOnMove**

Forces to unlock tickets after being moved to another queue.

默认值:

```
$Self->{'Ticket::EventModulePost'}->{'930-ForceUnlockOnMove'} = {
  'Event' => 'TicketQueueUpdate',
  'Module' => 'Kernel::System::Ticket::Event::ForceUnlock'
};
```



```
};
```

### **Ticket::EventModulePost###940-TicketArticleNewMessageUpdate**

Update Ticket "Seen" flag if every article got seen or a new Article got created.

默认值:

```
$Self->{'Ticket::EventModulePost'}->{'940-TicketArticleNewMessageUpdate'} = {
  'Event' => 'ArticleCreate|ArticleFlagSet',
  'Module' => 'Kernel::System::Ticket::Event::TicketNewMessageUpdate'
};
```

### **DynamicFieldFromCustomerUser::Mapping**

Define a mapping between variables of the customer user data (keys) and dynamic fields of a ticket (values). The purpose is to store customer user data in ticket dynamic fields. The dynamic fields must be present in the system and should be enabled for AgentTicketFreeText, so that they can be set/updated manually by the agent. They mustn't be enabled for AgentTicketPhone, AgentTicketEmail and AgentTicketCustomer. If they were, they would have precedence over the automatically set values. To use this mapping, you have to also activate the next setting below.

这个设置默认没有激活。

默认值:

```
$Self->{'DynamicFieldFromCustomerUser::Mapping'} = {
  'UserFirstname' => 'CustomerFirstname'
};
```

### **Ticket::EventModulePost###950-DynamicFieldFromCustomerUser**

This event module stores attributes from CustomerUser as DynamicFields tickets. Please see the setting above for how to configure the mapping.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::EventModulePost'}->{'950-DynamicFieldFromCustomerUser'} = {
  'Event' => '(TicketCreate|TicketCustomerUpdate)',
  'Module' => 'Kernel::System::Ticket::Event::DynamicFieldFromCustomerUser'
};
```

### **Ticket::CustomModule###001-CustomModule**

Overloads (redefines) existing functions in Kernel::System::Ticket. Used to easily add customizations.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::CustomModule'}->{'001-CustomModule'} =
  'Kernel::System::Ticket::Custom';
```

### **Ticket::ViewableSenderTypes**

Defines the default viewable sender types of a ticket (default: customer).

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::ViewableSenderTypes'} = [  
  '\customer\  
];
```

### **Ticket::ViewableLocks**

Defines the viewable locks of a ticket. NOTE: When you change this setting, make sure to delete the cache in order to use the new value. Default: unlock, tmp\_lock.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::ViewableLocks'} = [  
  '\unlock\  
  '\tmp_lock\  
];
```

### **Ticket::ViewableStateType**

Defines the valid state types for a ticket.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::ViewableStateType'} = [  
  'new',  
  'open',  
  'pending reminder',  
  'pending auto'  
];
```

### **Ticket::UnlockStateType**

Defines the valid states for unlocked tickets. To unlock tickets the script "bin/otrs.Console.pl Maint::Ticket::UnlockTimeout" can be used.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::UnlockStateType'} = [  
  'new',  
  'open'  
];
```

### **Ticket::PendingNotificationOnlyToOwner**

Sends reminder notifications of unlocked ticket after reaching the reminder date (only sent to ticket owner).

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::PendingNotificationOnlyToOwner'} = '0';
```

### **Ticket::PendingNotificationNotToResponsible**

Disables sending reminder notifications to the responsible agent of a ticket (Ticket::Responsible needs to be activated).

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::PendingNotificationNotToResponsible'} = '0';
```

### **Ticket::PendingReminderStateType**

Defines the state type of the reminder for pending tickets.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::PendingReminderStateType'} = [  
  'pending reminder'  
];
```

### **Ticket::PendingAutoStateType**

Determines the possible states for pending tickets that changed state after reaching time limit.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::PendingAutoStateType'} = [  
  'pending auto'  
];
```

### **Ticket::StateAfterPending**

Defines which states should be set automatically (Content), after the pending time of state (Key) has been reached.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::StateAfterPending'} = {  
  'pending auto close+' => 'closed successful',  
  'pending auto close-' => 'closed unsuccessful'  
};
```

### **System::Permission**

Standard available permissions for agents within the application. If more permissions are needed, they can be entered here. Permissions must be defined to be effective. Some other good permissions have also been provided built-in: note, close, pending, customer, freetext, move, compose, responsible, forward, and bounce. Make sure that "rw" is always the last registered permission.

不能使这个设置失效。

默认值:

```
$Self->{'System::Permission'} = [  
  'ro',  
  'move_into',  
  'create',  
  'note',  
  'owner',  
  'priority',  
  'rw'  
];
```

### **Ticket::Permission###1-OwnerCheck**

Module to grant access to the owner of a ticket.

默认值:

```
$Self->{'Ticket::Permission'}->{'1-OwnerCheck'} = {  
  'Granted' => '1',  
  'Module' => 'Kernel::System::Ticket::Permission::OwnerCheck',  
  'Required' => '0'  
};
```

### **Ticket::Permission::OwnerCheck::Queues**

Optional queue limitation for the OwnerCheck permission module. If set, permission is only granted for tickets in the specified queues.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Permission::OwnerCheck::Queues'} = {  
  'Misc' => 'note',  
  'Postmaster' => 'ro, move, note',  
  'Raw' => 'rw'  
};
```

### **Ticket::Permission###2-ResponsibleCheck**

Module to grant access to the agent responsible of a ticket.

默认值:

```
$Self->{'Ticket::Permission'}->{'2-ResponsibleCheck'} = {  
  'Granted' => '1',  
  'Module' => 'Kernel::System::Ticket::Permission::ResponsibleCheck',  
  'Required' => '0'  
};
```

### **Ticket::Permission::ResponsibleCheck::Queues**

Optional queue limitation for the ResponsibleCheck permission module. If set, permission is only granted for tickets in the specified queues.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Permission::ResponsibleCheck::Queues'} = {  
  'Misc' => 'note',  
  'Postmaster' => 'ro, move, note',  
  'Raw' => 'rw'  
};
```

### **Ticket::Permission###3-GroupCheck**

Module to check the group permissions for the access to tickets.

默认值:

```
$Self->{'Ticket::Permission'}->{'3-GroupCheck'} = {  
  'Granted' => '1',  
  'Module' => 'Kernel::System::Ticket::Permission::GroupCheck',  
  'Required' => '0'  
};
```

### **Ticket::Permission###4-WatcherCheck**

Module to grant access to the watcher agents of a ticket.

默认值:

```
$Self->{'Ticket::Permission'}->{'4-WatcherCheck'} = {  
  'Granted' => '1',  
  'Module' => 'Kernel::System::Ticket::Permission::WatcherCheck',  
  'Required' => '0'  
};
```

### **Ticket::Permission###5-CreatorCheck**

Module to grant access to the creator of a ticket.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Permission'}->{'5-CreatorCheck'} = {  
  'Granted' => '1',  
  'Module' => 'Kernel::System::Ticket::Permission::CreatorCheck',  
  'Required' => '0'  
};
```

### **Ticket::Permission::CreatorCheck::Queues**

Optional queue limitation for the CreatorCheck permission module. If set, permission is only granted for tickets in the specified queues.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Permission::CreatorCheck::Queues'} = {  
  'Misc' => 'note',  
  'Postmaster' => 'ro, move, note',  
  'Raw' => 'rw'  
};
```

### **Ticket::Permission###6-InvolvedCheck**

Module to grant access to any agent that has been involved in a ticket in the past (based on ticket history entries).

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Permission'}->{'6-InvolvedCheck'} = {  
  'Granted' => '1',  
  'Module' => 'Kernel::System::Ticket::Permission::InvolvedCheck',  
  'Required' => '0'  
};
```

### **Ticket::Permission::InvolvedCheck::Queues**

Optional queue limitation for the InvolvedCheck permission module. If set, permission is only granted for tickets in the specified queues.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Permission::InvolvedCheck::Queues'} = {  
  'Misc' => 'note',  
  'Postmaster' => 'ro, move, note',  
  'Raw' => 'rw'  
};
```

### **CustomerTicket::Permission###1-GroupCheck**

Module to check the group permissions for customer access to tickets.

默认值:

```
$Self->{'CustomerTicket::Permission'}->{'1-GroupCheck'} = {  
  'Granted' => '0',  
  'Module' => 'Kernel::System::Ticket::CustomerPermission::GroupCheck',  
  'Required' => '1'  
};
```

### **CustomerTicket::Permission###2-CustomerUserIDCheck**

Module to grant access if the CustomerUserID of the ticket matches the CustomerUserID of the customer.

默认值:

```
$Self->{'CustomerTicket::Permission'}->{'2-CustomerUserIDCheck'} = {  
  'Granted' => '1',  
  'Module' => 'Kernel::System::Ticket::CustomerPermission::CustomerUserIDCheck',  
  'Required' => '0'  
};
```

### **CustomerTicket::Permission###3-CustomerIDCheck**

Module to grant access if the CustomerID of the ticket matches the CustomerID of the customer.

默认值:

```
$Self->{'CustomerTicket::Permission'}->{'3-CustomerIDCheck'} = {  
  'Granted' => '1',  
  'Module' => 'Kernel::System::Ticket::CustomerPermission::CustomerIDCheck',  
  'Required' => '0'  
};
```

### **Ticket::DefineEmailFrom**

Defines how the From field from the emails (sent from answers and email tickets) should look like.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::DefineEmailFrom'} = 'SystemAddressName';
```

### **Ticket::DefineEmailFromSeparator**

Defines the separator between the agents real name and the given queue email address.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::DefineEmailFromSeparator'} = 'via';
```

### **CustomerNotifyJustToRealCustomer**

Sends customer notifications just to the mapped customer.

不能使这个设置失效。

默认值:

```
$Self->{'CustomerNotifyJustToRealCustomer'} = '0';
```

## AgentSelfNotifyOnAction

Specifies if an agent should receive email notification of his own actions.

不能使这个设置失效。

默认值:

```
$Self->{'AgentSelfNotifyOnAction'} = '0';
```

## Ticket::EventModulePost###900-GenericAgent

Event module registration. For more performance you can define a trigger event (e. g. Event => TicketCreate).

默认值:

```
$Self->{'Ticket::EventModulePost'}->{'900-GenericAgent'} = {  
  'Event' => '',  
  'Module' => 'Kernel::System::Ticket::Event::GenericAgent',  
  'Transaction' => '1'  
};
```

## Ticket::GenericAgentTicketSearch###ExtendedSearchCondition

Allows extended search conditions in ticket search of the generic agent interface. With this feature you can search e. g. ticket title with this kind of conditions like "(\*key1\*&&\*key2\*)" or "(\*key1\*||\*key2\*)".

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::GenericAgentTicketSearch'}->{'ExtendedSearchCondition'} = '1';
```

## Ticket::GenericAgentRunLimit

Set the limit of tickets that will be executed on a single genericagent job execution.

默认值:

```
$Self->{'Ticket::GenericAgentRunLimit'} = '4000';
```

## Ticket::GenericAgentAllowCustomScriptExecution

Allows generic agent to execute custom command line scripts.

默认值:

```
$Self->{'Ticket::GenericAgentAllowCustomScriptExecution'} = '1';
```

## Ticket::GenericAgentAllowCustomModuleExecution

Allows generic agent to execute custom modules.

默认值:

```
$Self->{'Ticket::GenericAgentAllowCustomModuleExecution'} = '1';
```

## Ticket::UnlockOnAway

Unlock tickets whenever a note is added and the owner is out of office.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::UnlockOnAway'} = '1';
```

### **Ticket::IncludeUnknownTicketCustomers**

Include unknown customers in ticket filter.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::IncludeUnknownTicketCustomers'} = '0';
```

### **StandardTemplate::Types**

Defines the list of types for templates.

默认值:

```
$Self->{'StandardTemplate::Types'} = {  
  'Answer' => 'Answer',  
  'Create' => 'Create',  
  'Email' => 'Email',  
  'Forward' => 'Forward',  
  'Note' => 'Note',  
  'PhoneCall' => 'Phone call'  
};
```

### **StandardTemplate2QueueByCreating**

List of default Standard Templates which are assigned automatically to new Queues upon creation.

这个设置默认没有激活。

默认值:

```
$Self->{'StandardTemplate2QueueByCreating'} = [  
  ''  
];
```

### **Ticket::Frontend::DefaultRecipientDisplayType**

Default display type for recipient (To,Cc) names in AgentTicketZoom and CustomerTicketZoom.

默认值:

```
$Self->{'Ticket::Frontend::DefaultRecipientDisplayType'} = 'Realname';
```

### **Ticket::Frontend::DefaultSenderDisplayType**

Default display type for sender (From) names in AgentTicketZoom and CustomerTicketZoom.

默认值:

```
$Self->{'Ticket::Frontend::DefaultSenderDisplayType'} = 'Realname';
```

## **Ticket → Core::TicketACL**

### **Ticket::Acl::Module###1-Ticket::Acl::Module**

ACL module that allows closing parent tickets only if all its children are already closed ("State" shows which states are not available for the parent ticket until all child tickets are closed).



这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Acl::Module'}->{'1-Ticket::Acl::Module'} = {  
  'Module' => 'Kernel::System::Ticket::Acl::CloseParentAfterClosedChilds',  
  'State' => [  
    'closed successful',  
    'closed unsuccessful'  
  ]  
};
```

### **TicketACL::Default::Action**

Default ACL values for ticket actions.

不能使这个设置失效。

默认值:

```
$Self->{'TicketACL::Default::Action'} = {};
```

### **ACLKeysLevel1Match**

Defines which items are available in first level of the ACL structure.

默认值:

```
$Self->{'ACLKeysLevel1Match'} = {  
  'Properties' => 'Properties',  
  'PropertiesDatabase' => 'PropertiesDatabase'  
};
```

### **ACLKeysLevel1Change**

Defines which items are available in first level of the ACL structure.

默认值:

```
$Self->{'ACLKeysLevel1Change'} = {  
  'Possible' => 'Possible',  
  'PossibleAdd' => 'PossibleAdd',  
  'PossibleNot' => 'PossibleNot'  
};
```

### **ACLKeysLevel2::Possible**

Defines which items are available in second level of the ACL structure.

默认值:

```
$Self->{'ACLKeysLevel2::Possible'} = {  
  'Action' => 'Action',  
  'ActivityDialog' => 'ActivityDialog',  
  'Process' => 'Process',  
  'Ticket' => 'Ticket'  
};
```

### **ACLKeysLevel2::PossibleAdd**

Defines which items are available in second level of the ACL structure.

默认值:

```
$Self->{'ACLKeysLevel2::PossibleAdd'} = {  
  'Action' => 'Action',  
  'ActivityDialog' => 'ActivityDialog',
```

```
'Process' => 'Process',
'Ticket' => 'Ticket'
};
```

### ACLKeysLevel2::PossibleNot

Defines which items are available in second level of the ACL structure.

默认值:

```
$Self->{'ACLKeysLevel2::PossibleNot'} = {
  'Action' => 'Action',
  'ActivityDialog' => 'ActivityDialog',
  'Process' => 'Process',
  'Ticket' => 'Ticket'
};
```

### ACLKeysLevel2::Properties

Defines which items are available in second level of the ACL structure.

默认值:

```
$Self->{'ACLKeysLevel2::Properties'} = {
  'CustomerUser' => 'CustomerUser',
  'DynamicField' => 'DynamicField',
  'Frontend' => 'Frontend',
  'Owner' => 'Owner',
  'Priority' => 'Priority',
  'Process' => 'Process',
  'Queue' => 'Queue',
  'Responsible' => 'Responsible',
  'SLA' => 'SLA',
  'Service' => 'Service',
  'State' => 'State',
  'Ticket' => 'Ticket',
  'Type' => 'Type',
  'User' => 'User'
};
```

### ACLKeysLevel2::PropertiesDatabase

Defines which items are available in second level of the ACL structure.

默认值:

```
$Self->{'ACLKeysLevel2::PropertiesDatabase'} = {
  'CustomerUser' => 'CustomerUser',
  'DynamicField' => 'DynamicField',
  'Owner' => 'Owner',
  'Priority' => 'Priority',
  'Process' => 'Process',
  'Queue' => 'Queue',
  'Responsible' => 'Responsible',
  'SLA' => 'SLA',
  'Service' => 'Service',
  'State' => 'State',
  'Ticket' => 'Ticket',
  'Type' => 'Type',
  'User' => 'User'
};
```

### ACLKeysLevel3::Actions###100-Default

Defines which items are available for 'Action' in third level of the ACL structure.

默认值:

```
$Self->{'ACLKeysLevel3::Actions'}->{'100-Default'} = [
```

```
'AgentTicketBounce',  
'AgentTicketClose',  
'AgentTicketCompose',  
'AgentTicketCustomer',  
'AgentTicketForward',  
'AgentTicketEmailOutbound',  
'AgentTicketFreeText',  
'AgentTicketHistory',  
'AgentTicketLink',  
'AgentTicketLock',  
'AgentTicketMerge',  
'AgentTicketMove',  
'AgentTicketNote',  
'AgentTicketOwner',  
'AgentTicketPending',  
'AgentTicketPhone',  
'AgentTicketPhoneInbound',  
'AgentTicketPhoneOutbound',  
'AgentTicketPlain',  
'AgentTicketPrint',  
'AgentTicketPriority',  
'AgentTicketProcess',  
'AgentTicketResponsible',  
'AgentTicketSearch',  
'AgentTicketWatcher',  
'AgentTicketZoom',  
'AgentLinkObject',  
'CustomerTicketProcess'  
];
```

### **ACL::CacheTTL**

Cache time in seconds for the DB ACL backend.

不能使这个设置失效。

默认值:

```
$Self->{'ACL::CacheTTL'} = '3600';
```

### **TicketACL::Debug::Enabled**

If enabled debugging information for ACLs is logged.

不能使这个设置失效。

默认值:

```
$Self->{'TicketACL::Debug::Enabled'} = '0';
```

### **TicketACL::Debug::LogPriority**

Defines the priority in which the information is logged and presented.

这个设置默认没有激活。

默认值:

```
$Self->{'TicketACL::Debug::LogPriority'} = 'debug';
```

### **TicketACL::Debug::Filter###00-Default**

Filter for debugging ACLs. Note: More ticket attributes can be added in the format <OTRS\_TICKET\_Attribute> e.g. <OTRS\_TICKET\_Priority>.

这个设置默认没有激活。

默认值:

```
$Self->{'TicketACL::Debug::Filter'}->{'00-Default'} = {  
  '<OTRS_TICKET_TicketNumber>' => '',  
  'ACLName' => ''  
};
```

## Ticket → Core::TicketBulkAction

### Ticket::Frontend::BulkFeature

Enables ticket bulk action feature for the agent frontend to work on more than one ticket at a time.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::BulkFeature'} = '1';
```

### Ticket::Frontend::BulkFeatureGroup

Enables ticket bulk action feature only for the listed groups.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::BulkFeatureGroup'} = [  
  'admin',  
  'users'  
];
```

## Ticket → Core::TicketDynamicFieldDefault

### Ticket::EventModulePost###TicketDynamicFieldDefault

Event module registration. For more performance you can define a trigger event (e.g. Event => TicketCreate). This is only possible if all Ticket dynamic fields need the same event.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::EventModulePost'}->{'TicketDynamicFieldDefault'} = {  
  'Module' => 'Kernel::System::Ticket::Event::TicketDynamicFieldDefault',  
  'Transaction' => '1'  
};
```

### Ticket::TicketDynamicFieldDefault###Element1

Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (<http://otrs.github.io/doc/>), chapter "Ticket Event Module".

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element1'} = {  
  'Event' => 'TicketCreate',  
  'Name' => 'Field1',  
  'Value' => 'Default'  
};
```

## **Ticket::TicketDynamicFieldDefault###Element2**

Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (<http://otrs.github.io/doc/>), chapter "Ticket Event Module".

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element2'} = {  
  'Event' => '',  
  'Name' => '',  
  'Value' => ''  
};
```

## **Ticket::TicketDynamicFieldDefault###Element3**

Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (<http://otrs.github.io/doc/>), chapter "Ticket Event Module".

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element3'} = {  
  'Event' => '',  
  'Name' => '',  
  'Value' => ''  
};
```

## **Ticket::TicketDynamicFieldDefault###Element4**

Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (<http://otrs.github.io/doc/>), chapter "Ticket Event Module".

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element4'} = {  
  'Event' => '',  
  'Name' => '',  
  'Value' => ''  
};
```

## **Ticket::TicketDynamicFieldDefault###Element5**

Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (<http://otrs.github.io/doc/>), chapter "Ticket Event Module".

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element5'} = {  
  'Event' => '',  
  'Name' => '',  
  'Value' => ''  
};
```

```
};
```

### **Ticket::TicketDynamicFieldDefault###Element6**

Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (<http://otrs.github.io/doc/>), chapter "Ticket Event Module".

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element6'} = {  
  'Event' => '',  
  'Name' => '',  
  'Value' => ''  
};
```

### **Ticket::TicketDynamicFieldDefault###Element7**

Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (<http://otrs.github.io/doc/>), chapter "Ticket Event Module".

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element7'} = {  
  'Event' => '',  
  'Name' => '',  
  'Value' => ''  
};
```

### **Ticket::TicketDynamicFieldDefault###Element8**

Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (<http://otrs.github.io/doc/>), chapter "Ticket Event Module".

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element8'} = {  
  'Event' => '',  
  'Name' => '',  
  'Value' => ''  
};
```

### **Ticket::TicketDynamicFieldDefault###Element9**

Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (<http://otrs.github.io/doc/>), chapter "Ticket Event Module".

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element9'} = {  
  'Event' => '',
```

```
'Name' => '',  
'Value' => ''  
};
```

### **Ticket::TicketDynamicFieldDefault###Element10**

Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (<http://otrs.github.io/doc/>), chapter "Ticket Event Module".

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element10'} = {  
  'Event' => '',  
  'Name' => '',  
  'Value' => ''  
};
```

### **Ticket::TicketDynamicFieldDefault###Element11**

Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (<http://otrs.github.io/doc/>), chapter "Ticket Event Module".

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element11'} = {  
  'Event' => '',  
  'Name' => '',  
  'Value' => ''  
};
```

### **Ticket::TicketDynamicFieldDefault###Element12**

Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (<http://otrs.github.io/doc/>), chapter "Ticket Event Module".

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element12'} = {  
  'Event' => '',  
  'Name' => '',  
  'Value' => ''  
};
```

### **Ticket::TicketDynamicFieldDefault###Element13**

Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (<http://otrs.github.io/doc/>), chapter "Ticket Event Module".

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element13'} = {  
  'Event' => '',  
  'Name' => '',  
  'Value' => ''  
};
```

### **Ticket::TicketDynamicFieldDefault###Element14**

Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (<http://otrs.github.io/doc/>), chapter "Ticket Event Module".

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element14'} = {  
  'Event' => '',  
  'Name' => '',  
  'Value' => ''  
};
```

### **Ticket::TicketDynamicFieldDefault###Element15**

Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (<http://otrs.github.io/doc/>), chapter "Ticket Event Module".

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element15'} = {  
  'Event' => '',  
  'Name' => '',  
  'Value' => ''  
};
```

### **Ticket::TicketDynamicFieldDefault###Element16**

Configures a default TicketDynamicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that will be set, and "Event" defines the trigger event. Please check the developer manual (<http://otrs.github.io/doc/>), chapter "Ticket Event Module".

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element16'} = {  
  'Event' => '',  
  'Name' => '',  
  'Value' => ''  
};
```

## **Ticket → Core::TicketWatcher**

### **Ticket::Watcher**

Enables or disables the ticket watcher feature, to keep track of tickets without being the owner nor the responsible.

不能使这个设置失效。



默认值:

```
$Self->{'Ticket::Watcher'} = '0';
```

### **Ticket::WatcherGroup**

Enables ticket watcher feature only for the listed groups.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::WatcherGroup'} = [  
  'admin',  
  'users'  
];
```

## **Ticket → Frontend::Admin**

### **Events###Ticket**

List of all ticket events to be displayed in the GUI.

不能使这个设置失效。

默认值:

```
$Self->{'Events'}->{'Ticket'} = [  
  'TicketCreate',  
  'TicketDelete',  
  'TicketTitleUpdate',  
  'TicketUnlockTimeoutUpdate',  
  'TicketQueueUpdate',  
  'TicketTypeUpdate',  
  'TicketServiceUpdate',  
  'TicketSLAUpdate',  
  'TicketCustomerUpdate',  
  'TicketPendingTimeUpdate',  
  'TicketLockUpdate',  
  'TicketArchiveFlagUpdate',  
  'TicketStateUpdate',  
  'TicketOwnerUpdate',  
  'TicketResponsibleUpdate',  
  'TicketPriorityUpdate',  
  'HistoryAdd',  
  'HistoryDelete',  
  'TicketAccountTime',  
  'TicketMerge',  
  'TicketSubscribe',  
  'TicketUnsubscribe',  
  'TicketFlagSet',  
  'TicketFlagDelete',  
  'TicketSlaveLinkAdd',  
  'TicketSlaveLinkDelete',  
  'TicketMasterLinkDelete',  
  'EscalationResponseTimeNotifyBefore',  
  'EscalationUpdateTimeNotifyBefore',  
  'EscalationSolutionTimeNotifyBefore',  
  'EscalationResponseTimeStart',  
  'EscalationUpdateTimeStart',  
  'EscalationSolutionTimeStart',  
  'EscalationResponseTimeStop',  
  'EscalationUpdateTimeStop',  
  'EscalationSolutionTimeStop',  
  'NotificationNewTicket',  
  'NotificationFollowUp',  
  'NotificationLockTimeout',  
  'NotificationOwnerUpdate',  
  'NotificationResponsibleUpdate',  
  'NotificationAddNote',
```

```
'NotificationMove',
'NotificationPendingReminder',
'NotificationEscalation',
'NotificationEscalationNotifyBefore',
'NotificationServiceUpdate'
];
```

### Events###Article

List of all article events to be displayed in the GUI.

不能使这个设置失效。

默认值:

```
$Self->{'Events'}->{'Article'} = [
'ArticleCreate',
'ArticleUpdate',
'ArticleSend',
'ArticleBounce',
'ArticleAgentNotification',
'ArticleCustomerNotification',
'ArticleAutoResponse',
'ArticleFlagSet',
'ArticleFlagDelete',
'ArticleAgentNotification',
'ArticleCustomerNotification'
];
```

### Events###Queue

List of all queue events to be displayed in the GUI.

不能使这个设置失效。

默认值:

```
$Self->{'Events'}->{'Queue'} = [
'QueueCreate',
'QueueUpdate'
];
```

## Ticket → Frontend::Admin::AdminNotificationEvent

### Frontend::Admin::AdminNotificationEvent###RichText

Uses richtext for viewing and editing ticket notification.

默认值:

```
$Self->{'Frontend::Admin::AdminNotificationEvent'}->{'RichText'} = '1';
```

### Frontend::Admin::AdminNotificationEvent###RichTextWidth

Defines the width for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).

默认值:

```
$Self->{'Frontend::Admin::AdminNotificationEvent'}->{'RichTextWidth'} = '620';
```

### Frontend::Admin::AdminNotificationEvent###RichTextHeight

Defines the height for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).

默认值:

```
$Self->{'Frontend::Admin::AdminNotificationEvent'}->{'RichTextHeight'} = '320';
```

### Notification::Transport###Email

Defines all the parameters for this notification transport.

默认值:

```
$Self->{'Notification::Transport'}->{'Email'} = {
  'AgentEnabledByDefault' => '1',
  'Icon' => 'fa fa-envelope',
  'IsOTRSBusinessTransport' => '0',
  'Module' => 'Kernel::System::Ticket::Event::NotificationEvent::Transport::Email',
  'Name' => 'Email',
  'Prio' => '100'
};
```

### Notification::Transport###NotificationView

Defines all the parameters for this notification transport.

默认值:

```
$Self->{'Notification::Transport'}->{'NotificationView'} = {
  'AgentEnabledByDefault' => '0',
  'Icon' => 'fa fa-th-list',
  'IsOTRSBusinessTransport' => '1',
  'Module' =>
  'Kernel::System::Ticket::Event::NotificationEvent::Transport::NotificationView',
  'Name' => 'Web View',
  'Prio' => '110'
};
```

### Notification::Transport###SMS

Defines all the parameters for this notification transport.

默认值:

```
$Self->{'Notification::Transport'}->{'SMS'} = {
  'AgentEnabledByDefault' => '0',
  'Icon' => 'fa fa-mobile',
  'IsOTRSBusinessTransport' => '1',
  'Module' => 'Kernel::System::Ticket::Event::NotificationEvent::Transport::SMS',
  'Name' => 'SMS (Short Message Service)',
  'Prio' => '120'
};
```

### Notification::CharactersPerLine

Defines the number of character per line used in case an HTML article preview replacement on TemplateGenerator for EventNotifications.

不能使这个设置失效。

默认值:

```
$Self->{'Notification::CharactersPerLine'} = '80';
```

## Ticket → Frontend::Admin::ModuleRegistration

### Frontend::Module###AdminACL

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminACL'} = {
```

```
'Description' => 'This module is part of the admin area of OTRS.',
'Group' => [
  'admin'
],
'Loader' => {
  'CSS' => [
    'Core.Agent.Admin.ACL.css'
  ],
  'JavaScript' => [
    'Core.Agent.Admin.ACL.js'
  ]
},
'NavBarModule' => {
  'Block' => 'Ticket',
  'Description' => 'Configure and manage ACLs.',
  'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
  'Name' => 'Access Control Lists (ACL)',
  'Prio' => '750'
},
'NavBarName' => 'Admin',
'Title' => 'Access Control Lists (ACL)'
};
```

### Frontend::Module###AdminQueue

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminQueue'} = {
  'Description' => 'This module is part of the admin area of OTRS.',
  'Group' => [
    'admin'
  ],
  'NavBarModule' => {
    'Block' => 'Queue',
    'Description' => 'Create and manage queues.',
    'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
    'Name' => 'Queues',
    'Prio' => '100'
  },
  'NavBarName' => 'Admin',
  'Title' => 'Queues'
};
```

### Frontend::Module###AdminTemplate

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminTemplate'} = {
  'Description' => 'This module is part of the admin area of OTRS.',
  'Group' => [
    'admin'
  ],
  'NavBarModule' => {
    'Block' => 'Queue',
    'Description' => 'Create and manage templates.',
    'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
    'Name' => 'Templates',
    'Prio' => '200'
  },
  'NavBarName' => 'Admin',
  'Title' => 'Templates'
};
```

### Frontend::Module###AdminQueueTemplates

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminQueueTemplates'} = {
  'Description' => 'This module is part of the admin area of OTRS.',
  'Group' => [
    'admin'
  ],
  'NavBarModule' => {
    'Block' => 'Queue',
    'Description' => 'Link templates to queues.',
    'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
    'Name' => 'Templates <-> Queues',
    'Prio' => '300'
  },
  'NavBarName' => 'Admin',
  'Title' => 'Templates <-> Queues'
};
```

### Frontend::Module###AdminAutoResponse

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminAutoResponse'} = {
  'Description' => 'This module is part of the admin area of OTRS.',
  'Group' => [
    'admin'
  ],
  'NavBarModule' => {
    'Block' => 'Queue',
    'Description' => 'Create and manage responses that are automatically sent.',
    'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
    'Name' => 'Auto Responses',
    'Prio' => '400'
  },
  'NavBarName' => 'Admin',
  'Title' => 'Auto Responses'
};
```

### Frontend::Module###AdminQueueAutoResponse

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminQueueAutoResponse'} = {
  'Description' => 'This module is part of the admin area of OTRS.',
  'Group' => [
    'admin'
  ],
  'NavBarModule' => {
    'Block' => 'Queue',
    'Description' => 'Link queues to auto responses.',
    'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
    'Name' => 'Auto Responses <-> Queues',
    'Prio' => '500'
  },
  'NavBarName' => 'Admin',
  'Title' => 'Auto Responses <-> Queues'
};
```

### Frontend::Module###AdminAttachment

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminAttachment'} = {
  'Description' => 'This module is part of the admin area of OTRS.',
```

```
'Group' => [
  'admin'
],
'NavBarModule' => {
  'Block' => 'Queue',
  'Description' => 'Create and manage attachments.',
  'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
  'Name' => 'Attachments',
  'Prio' => '600'
},
'NavBarName' => 'Admin',
'Title' => 'Attachments'
};
```

### Frontend::Module###AdminTemplateAttachment

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminTemplateAttachment'} = {
  'Description' => 'This module is part of the admin area of OTRS.',
  'Group' => [
    'admin'
  ],
  'NavBarModule' => {
    'Block' => 'Queue',
    'Description' => 'Link attachments to templates.',
    'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
    'Name' => 'Attachments <-> Templates',
    'Prio' => '700'
  },
  'NavBarName' => 'Admin',
  'Title' => 'Attachments <-> Templates'
};
```

### Frontend::Module###AdminSalutation

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminSalutation'} = {
  'Description' => 'This module is part of the admin area of OTRS.',
  'Group' => [
    'admin'
  ],
  'NavBarModule' => {
    'Block' => 'Queue',
    'Description' => 'Create and manage salutations.',
    'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
    'Name' => 'Salutations',
    'Prio' => '800'
  },
  'NavBarName' => 'Admin',
  'Title' => 'Salutations'
};
```

### Frontend::Module###AdminSignature

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminSignature'} = {
  'Description' => 'This module is part of the admin area of OTRS.',
  'Group' => [
    'admin'
  ],
  'NavBarModule' => {
```

```
'Block' => 'Queue',
'Description' => 'Create and manage signatures.',
'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
'Name' => 'Signatures',
'Prio' => '900'
},
'NavBarName' => 'Admin',
'Title' => 'Signatures'
};
```

### Frontend::Module###AdminSystemAddress

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminSystemAddress'} = {
'Description' => 'This module is part of the admin area of OTRS.',
'Group' => [
'admin'
],
'NavBarModule' => {
'Block' => 'Email',
'Description' => 'Set sender email addresses for this system.',
'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
'Name' => 'Email Addresses',
'Prio' => '300'
},
'NavBarName' => 'Admin',
'Title' => 'Email Addresses'
};
```

### Frontend::Module###AdminNotificationEvent

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminNotificationEvent'} = {
'Description' => 'This module is part of the admin area of OTRS.',
'Group' => [
'admin'
],
'Loader' => {
'CSS' => [
'Core.Agent.Admin.NotificationEvent.css'
],
'JavaScript' => [
'Core.Agent.Admin.NotificationEvent.js'
]
},
'NavBarModule' => {
'Block' => 'Ticket',
'Description' => 'Create and manage ticket notifications.',
'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
'Name' => 'Ticket Notifications',
'Prio' => '400'
},
'NavBarName' => 'Admin',
'Title' => 'Ticket Notifications'
};
```

### Frontend::Module###AdminService

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminService'} = {
'Description' => 'This module is part of the admin area of OTRS.',
```

```
'Group' => [
  'admin'
],
'NavBarModule' => {
  'Block' => 'Ticket',
  'Description' => 'Create and manage services.',
  'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
  'Name' => 'Services',
  'Prio' => '900'
},
'NavBarName' => 'Admin',
'Title' => 'Services'
};
```

### Frontend::Module###AdminSLA

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminSLA'} = {
  'Description' => 'This module is part of the admin area of OTRS.',
  'Group' => [
    'admin'
  ],
  'NavBarModule' => {
    'Block' => 'Ticket',
    'Description' => 'Create and manage Service Level Agreements (SLAs).',
    'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
    'Name' => 'Service Level Agreements',
    'Prio' => '1000'
  },
  'NavBarName' => 'Admin',
  'Title' => 'Service Level Agreements'
};
```

### Frontend::Module###AdminType

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminType'} = {
  'Description' => 'This module is part of the admin area of OTRS.',
  'Group' => [
    'admin'
  ],
  'NavBarModule' => {
    'Block' => 'Ticket',
    'Description' => 'Create and manage ticket types.',
    'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
    'Name' => 'Types',
    'Prio' => '700'
  },
  'NavBarName' => 'Admin',
  'Title' => 'Types'
};
```

### Frontend::Module###AdminState

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminState'} = {
  'Description' => 'This module is part of the admin area of OTRS.',
  'Group' => [
    'admin'
  ],
  'NavBarModule' => {
```



```
'Block' => 'Ticket',
'Description' => 'Create and manage ticket states.',
'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
'Name' => 'States',
'Prio' => '800'
},
'NavBarName' => 'Admin',
'Title' => 'States'
};
```

### Frontend::Module###AdminPriority

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminPriority'} = {
'Description' => 'This module is part of the admin area of OTRS.',
'Group' => [
'admin'
],
'NavBarModule' => {
'Block' => 'Ticket',
'Description' => 'Create and manage ticket priorities.',
'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
'Name' => 'Priorities',
'Prio' => '850'
},
'NavBarName' => 'Admin',
'Title' => 'Priorities'
};
```

### Frontend::Module###AdminGenericAgent

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AdminGenericAgent'} = {
'Description' => 'This module is part of the admin area of OTRS.',
'Group' => [
'admin'
],
'Loader' => {
'JavaScript' => [
'Core.Agent.Admin.GenericAgent.js'
]
},
'NavBarModule' => {
'Block' => 'System',
'Description' => 'Manage tasks triggered by event or time based execution.',
'Module' => 'Kernel::Output::HTML::NavBar::ModuleAdmin',
'Name' => 'GenericAgent',
'Prio' => '300'
},
'NavBarName' => 'Admin',
'Title' => 'GenericAgent'
};
```

## Ticket → Frontend::Agent

### Ticket::Frontend::PendingDiffTime

Time in seconds that gets added to the actual time if setting a pending-state (default: 86400 = 1 day).

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::PendingDiffTime'} = '86400';
```

### **Ticket::Frontend::MaxQueueLevel**

Define the max depth of queues.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::MaxQueueLevel'} = '5';
```

### **Ticket::Frontend::ListType**

Shows existing parent/child queue lists in the system in the form of a tree or a list.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::ListType'} = 'tree';
```

### **Ticket::Frontend::TextAreaEmail**

Permitted width for compose email windows.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::TextAreaEmail'} = '82';
```

### **Ticket::Frontend::TextAreaNote**

Permitted width for compose note windows.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::TextAreaNote'} = '78';
```

### **Ticket::Frontend::InformAgentMaxSize**

Max size (in rows) of the informed agents box in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::InformAgentMaxSize'} = '3';
```

### **Ticket::Frontend::InvolvedAgentMaxSize**

Max size (in rows) of the involved agents box in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::InvolvedAgentMaxSize'} = '3';
```

### **Ticket::Frontend::BlockLoadingRemoteContent**

Makes the application block external content loading.

默认值:

```
$Self->{'Ticket::Frontend::BlockLoadingRemoteContent'} = '0';
```

### **Ticket::Frontend::CustomerInfoCompose**

Shows the customer user information (phone and email) in the compose screen.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::CustomerInfoCompose'} = '1';
```

### **Ticket::Frontend::CustomerInfoComposeMaxSize**

Max size (in characters) of the customer information table (phone and email) in the compose screen.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::CustomerInfoComposeMaxSize'} = '22';
```

### **Ticket::Frontend::CustomerInfoZoom**

Shows the customer user's info in the ticket zoom view.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::CustomerInfoZoom'} = '1';
```

### **Ticket::Frontend::CustomerInfoZoomMaxSize**

Maximum size (in characters) of the customer information table in the ticket zoom view.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::CustomerInfoZoomMaxSize'} = '22';
```

### **Ticket::Frontend::DynamicFieldsZoomMaxSizeSidebar**

Maximum length (in characters) of the dynamic field in the sidebar of the ticket zoom view.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::DynamicFieldsZoomMaxSizeSidebar'} = '18';
```

### **Ticket::Frontend::DynamicFieldsZoomMaxSizeArticle**

Maximum length (in characters) of the dynamic field in the article of the ticket zoom view.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::DynamicFieldsZoomMaxSizeArticle'} = '160';
```

### **Ticket::Frontend::AccountTime**

Activates time accounting.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AccountTime'} = '1';
```

### **Ticket::Frontend::TimeUnits**

Sets the preferred time units (e.g. work units, hours, minutes).

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::TimeUnits'} = '(work units)';
```

### **Ticket::Frontend::NeedAccountedTime**

Defines if time accounting is mandatory in the agent interface. If activated, a note must be entered for all ticket actions (no matter if the note itself is configured as active or is originally mandatory for the individual ticket action screen).

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::NeedAccountedTime'} = '0';
```

### **Ticket::Frontend::BulkAccountedTime**

Defines if time accounting must be set to all tickets in bulk action.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::BulkAccountedTime'} = '1';
```

### **Ticket::Frontend::NeedSpellCheck**

Defines if composed messages have to be spell checked in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::NeedSpellCheck'} = '0';
```

### **Ticket::Frontend::NewOwnerSelection**

Shows an owner selection in phone and email tickets in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::NewOwnerSelection'} = '1';
```

### **Ticket::Frontend::NewResponsibleSelection**

Show a responsible selection in phone and email tickets in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::NewResponsibleSelection'} = '1';
```

### **Ticket::Frontend::NewQueueSelectionType**

Defines the recipient target of the phone ticket and the sender of the email ticket ("Queue" shows all queues, "System address" displays all system addresses) in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::NewQueueSelectionType'} = 'Queue';
```

### **Ticket::Frontend::NewQueueSelectionString**

Determines the strings that will be shown as recipient (To:) of the phone ticket and as sender (From:) of the email ticket in the agent interface. For Queue as NewQueueSelectionType "<Queue>" shows the names of the queues and for SystemAddress "<Realname> <<Email>>" shows the name and email of the recipient.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::NewQueueSelectionString'} = '<Queue>';
```

### **Ticket::Frontend::NewQueueOwnSelection**

Determines which options will be valid of the recipient (phone ticket) and the sender (email ticket) in the agent interface.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::NewQueueOwnSelection'} = {  
  '1' => 'First Queue',  
  '2' => 'Second Queue'  
};
```

### **Ticket::Frontend::ShowCustomerTickets**

Shows customer history tickets in AgentTicketPhone, AgentTicketEmail and AgentTicketCustomer.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::ShowCustomerTickets'} = '1';
```

### **NewTicketInNewWindow::Enabled**

If enabled, TicketPhone and TicketEmail will be open in new windows.

不能使这个设置失效。

默认值:

```
$Self->{'NewTicketInNewWindow::Enabled'} = '0';
```

### CustomerDBLink

Defines an external link to the database of the customer (e.g. 'http://yourhost/customer.php?CID=[% Data.CustomerID %]' or '').

不能使这个设置失效。

默认值:

```
$Self->{'CustomerDBLink'} = "[% Env("CGIHandle") %]?  
Action=AgentCustomerInformationCenter;CustomerID=[% Data.CustomerID | uri %]";
```

### CustomerDBLinkTarget

Defines the target attribute in the link to external customer database. E.g. 'target="cdb"'.

不能使这个设置失效。

默认值:

```
$Self->{'CustomerDBLinkTarget'} = '';
```

### CustomerDBLinkClass

Defines the target attribute in the link to external customer database. E.g. 'AsPopup PopupType\_TicketAction'.

不能使这个设置失效。

默认值:

```
$Self->{'CustomerDBLinkClass'} = '';
```

### Frontend::CommonParam###Action

Defines the default used Frontend-Module if no Action parameter given in the url on the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Frontend::CommonParam'}->{'Action'} = 'AgentDashboard';
```

### Frontend::CommonParam###QueueID

Default queue ID used by the system in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Frontend::CommonParam'}->{'QueueID'} = '0';
```

### Frontend::CommonParam###TicketID

Default ticket ID used by the system in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Frontend::CommonParam'}->{'TicketID'} = '';
```

## DefaultOverviewColumns

General ticket data shown in the ticket overviews (fall-back). Possible settings: 0 = Disabled, 1 = Available, 2 = Enabled by default. Note that TicketNumber can not be disabled, because it is necessary.

不能使这个设置失效。

默认值:

```
$Self->{'DefaultOverviewColumns'} = {
  'Age' => '2',
  'Changed' => '1',
  'Created' => '1',
  'CustomerCompanyName' => '1',
  'CustomerID' => '2',
  'CustomerName' => '1',
  'CustomerUserID' => '1',
  'EscalationResponseTime' => '1',
  'EscalationSolutionTime' => '1',
  'EscalationTime' => '1',
  'EscalationUpdateTime' => '1',
  'Lock' => '2',
  'Owner' => '2',
  'PendingTime' => '1',
  'Priority' => '1',
  'Queue' => '2',
  'Responsible' => '1',
  'SLA' => '1',
  'Service' => '1',
  'State' => '2',
  'TicketNumber' => '2',
  'Title' => '2',
  'Type' => '1'
};
```

## Ticket → Frontend::Agent::Dashboard

### DashboardBackend###0100-TicketPendingReminder

Parameters for the dashboard backend of the ticket pending reminder overview of the agent interface . "Limit" is the number of entries shown by default. "Group" is used to restrict the access to the plugin (e. g. Group: admin;group1;group2;). "Default" determines if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" is the cache time in minutes for the plugin. Note: Only Ticket attributes and Dynamic Fields (DynamicField\_NameX) are allowed for DefaultColumns. Possible settings: 0 = Disabled, 1 = Available, 2 = Enabled by default.

默认值:

```
$Self->{'DashboardBackend'}->{'0100-TicketPendingReminder'} = {
  'Attributes' => 'TicketPendingTimeOlderMinutes=1;StateType=pending
reminder;SortBy=PendingTime;OrderBy=Down;',
  'Block' => 'ContentLarge',
  'CacheTTLLocal' => '0.5',
  'Default' => '1',
  'DefaultColumns' => {
    'Age' => '2',
    'Changed' => '1',
    'Created' => '1',
    'CustomerCompanyName' => '1',
```

```

'CustomerID' => '1',
'CustomerName' => '1',
'CustomerUserID' => '1',
'EscalationResponseTime' => '1',
'EscalationSolutionTime' => '1',
'EscalationTime' => '1',
'EscalationUpdateTime' => '1',
'Lock' => '1',
'Owner' => '1',
'PendingTime' => '1',
'Priority' => '1',
'Queue' => '1',
'Responsible' => '1',
'SLA' => '1',
'Service' => '1',
'State' => '1',
'TicketNumber' => '2',
'Title' => '2',
'Type' => '1'
},
'Description' => 'All tickets with a reminder set where the reminder date has been
reached',
'Filter' => 'Locked',
'Group' => '',
'Limit' => '10',
'Module' => 'Kernel::Output::HTML::Dashboard::TicketGeneric',
'Permission' => 'rw',
'Time' => 'UntilTime',
'Title' => 'Reminder Tickets'
};

```

## DashboardBackend###0110-TicketEscalation

Parameters for the dashboard backend of the ticket escalation overview of the agent interface. "Limit" is the number of entries shown by default. "Group" is used to restrict the access to the plugin (e. g. Group: admin;group1;group2;). "Default" determines if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" is the cache time in minutes for the plugin. Note: Only Ticket attributes and Dynamic Fields (DynamicField\_NameX) are allowed for DefaultColumns. Possible settings: 0 = Disabled, 1 = Available, 2 = Enabled by default.

默认值:

```

$self->{'DashboardBackend'}->{'0110-TicketEscalation'} = {
  'Attributes' =>
  'TicketEscalationTimeOlderMinutes=1;SortBy=EscalationTime;OrderBy=Down;',
  'Block' => 'ContentLarge',
  'CacheTTLLocal' => '0.5',
  'Default' => '1',
  'DefaultColumns' => {
    'Age' => '2',
    'Changed' => '1',
    'Created' => '1',
    'CustomerCompanyName' => '1',
    'CustomerID' => '1',
    'CustomerName' => '1',
    'CustomerUserID' => '1',
    'EscalationResponseTime' => '1',
    'EscalationSolutionTime' => '1',
    'EscalationTime' => '1',
    'EscalationUpdateTime' => '1',
    'Lock' => '1',
    'Owner' => '1',
    'PendingTime' => '1',
    'Priority' => '1',
    'Queue' => '1',
    'Responsible' => '1',
    'SLA' => '1',
    'Service' => '1',
    'State' => '1',

```



```

    'TicketNumber' => '2',
    'Title' => '2',
    'Type' => '1'
  },
  'Description' => 'All escalated tickets',
  'Filter' => 'All',
  'Group' => '',
  'Limit' => '10',
  'Module' => 'Kernel::Output::HTML::Dashboard::TicketGeneric',
  'Permission' => 'rw',
  'Time' => 'EscalationTime',
  'Title' => 'Escalated Tickets'
};

```

### DashboardBackend###0120-TicketNew

Parameters for the dashboard backend of the new tickets overview of the agent interface. "Limit" is the number of entries shown by default. "Group" is used to restrict the access to the plugin (e. g. Group: admin;group1;group2;). "Default" determines if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" is the cache time in minutes for the plugin. Note: Only Ticket attributes and Dynamic Fields (DynamicField\_NameX) are allowed for DefaultColumns. Possible settings: 0 = Disabled, 1 = Available, 2 = Enabled by default.

默认值:

```

$self->{'DashboardBackend'}->{'0120-TicketNew'} = {
  'Attributes' => 'StateType=new;',
  'Block' => 'ContentLarge',
  'CacheTTLLocal' => '0.5',
  'Default' => '1',
  'DefaultColumns' => {
    'Age' => '2',
    'Changed' => '1',
    'Created' => '1',
    'CustomerCompanyName' => '1',
    'CustomerID' => '1',
    'CustomerName' => '1',
    'CustomerUserID' => '1',
    'EscalationResponseTime' => '1',
    'EscalationSolutionTime' => '1',
    'EscalationTime' => '1',
    'EscalationUpdateTime' => '1',
    'Lock' => '1',
    'Owner' => '1',
    'PendingTime' => '1',
    'Priority' => '1',
    'Queue' => '1',
    'Responsible' => '1',
    'SLA' => '1',
    'Service' => '1',
    'State' => '1',
    'TicketNumber' => '2',
    'Title' => '2',
    'Type' => '1'
  },
  'Description' => 'All new tickets, these tickets have not been worked on yet',
  'Filter' => 'All',
  'Group' => '',
  'Limit' => '10',
  'Module' => 'Kernel::Output::HTML::Dashboard::TicketGeneric',
  'Permission' => 'rw',
  'Time' => 'Age',
  'Title' => 'New Tickets'
};

```

### DashboardBackend###0130-TicketOpen

Parameters for the dashboard backend of the open tickets overview of the agent interface. "Limit" is the number of entries shown by default. "Group" is used to restrict

the access to the plugin (e. g. Group: admin;group1;group2;). "Default" determines if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" is the cache time in minutes for the plugin. Note: Only Ticket attributes and Dynamic Fields (DynamicField\_NameX) are allowed for DefaultColumns. Possible settings: 0 = Disabled, 1 = Available, 2 = Enabled by default.

默认值:

```
$Self->{'DashboardBackend'}->{'0130-TicketOpen'} = {
  'Attributes' => 'StateType=open;',
  'Block' => 'ContentLarge',
  'CacheTTLLocal' => '0.5',
  'Default' => '1',
  'DefaultColumns' => {
    'Age' => '2',
    'Changed' => '1',
    'Created' => '1',
    'CustomerCompanyName' => '1',
    'CustomerID' => '1',
    'CustomerName' => '1',
    'CustomerUserID' => '1',
    'EscalationResponseTime' => '1',
    'EscalationSolutionTime' => '1',
    'EscalationTime' => '1',
    'EscalationUpdateTime' => '1',
    'Lock' => '1',
    'Owner' => '1',
    'PendingTime' => '1',
    'Priority' => '1',
    'Queue' => '1',
    'Responsible' => '1',
    'SLA' => '1',
    'Service' => '1',
    'State' => '1',
    'TicketNumber' => '2',
    'Title' => '2',
    'Type' => '1'
  },
  'Description' => 'All open tickets, these tickets have already been worked on, but
  need a response',
  'Filter' => 'All',
  'Group' => '',
  'Limit' => '10',
  'Module' => 'Kernel::Output::HTML::Dashboard::TicketGeneric',
  'Permission' => 'rw',
  'Time' => 'Age',
  'Title' => 'Open Tickets / Need to be answered'
};
```

### DashboardBackend###0250-TicketStats

Parameters for the dashboard backend of the ticket stats of the agent interface. "Limit" is the number of entries shown by default. "Group" is used to restrict the access to the plugin (e. g. Group: admin;group1;group2;). "Default" determines if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" is the cache time in minutes for the plugin.

默认值:

```
$Self->{'DashboardBackend'}->{'0250-TicketStats'} = {
  'Block' => 'ContentSmall',
  'CacheTTLLocal' => '30',
  'Changed' => '1',
  'Closed' => '1',
  'Default' => '1',
  'Group' => '',
  'Module' => 'Kernel::Output::HTML::Dashboard::TicketStatsGeneric',
  'Permission' => 'rw',
  'Title' => '7 Day Stats'
};
```

### DashboardBackend###0260-TicketCalendar

Parameters for the dashboard backend of the upcoming events widget of the agent interface. "Limit" is the number of entries shown by default. "Group" is used to restrict the access to the plugin (e. g. Group: admin;group1;group2;). "Default" determines if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" is the cache time in minutes for the plugin.

默认值:

```
$Self->{'DashboardBackend'}->{'0260-TicketCalendar'} = {  
  'Block' => 'ContentSmall',  
  'CacheTTL' => '2',  
  'Default' => '1',  
  'Group' => '',  
  'Limit' => '6',  
  'Module' => 'Kernel::Output::HTML::Dashboard::Calendar',  
  'OwnerOnly' => '',  
  'Permission' => 'rw',  
  'Title' => 'Upcoming Events'  
};
```

### DashboardBackend###0270-TicketQueueOverview

Parameters for the dashboard backend of the queue overview widget of the agent interface. "Group" is used to restrict the access to the plugin (e. g. Group: admin;group1;group2;). "QueuePermissionGroup" is not mandatory, queues are only listed if they belong to this permission group if you enable it. "States" is a list of states, the key is the sort order of the state in the widget. "Default" determines if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" is the cache time in minutes for the plugin.

默认值:

```
$Self->{'DashboardBackend'}->{'0270-TicketQueueOverview'} = {  
  'Block' => 'ContentLarge',  
  'CacheTTLLocal' => '2',  
  'Default' => '1',  
  'Description' => 'Provides a matrix overview of the tickets per state per queue.',  
  'Group' => '',  
  'Module' => 'Kernel::Output::HTML::Dashboard::TicketQueueOverview',  
  'Permission' => 'rw',  
  'QueuePermissionGroup' => 'users',  
  'Sort' => 'SortBy=Age;OrderBy=Up',  
  'States' => {  
    '1' => 'new',  
    '4' => 'open',  
    '6' => 'pending reminder'  
  },  
  'Title' => 'Ticket Queue Overview'  
};
```

### DashboardBackend###0280-DashboardEventsTicketCalendar

Parameters for the dashboard backend of the ticket events calendar of the agent interface. "Limit" is the number of entries shown by default. "Group" is used to restrict the access to the plugin (e. g. Group: admin;group1;group2;). "Default" determines if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" is the cache time in minutes for the plugin.

默认值:

```
$Self->{'DashboardBackend'}->{'0280-DashboardEventsTicketCalendar'} = {  
  'Block' => 'ContentLarge',  
  'CacheTTL' => '0',  
  'Default' => '0',  
  'Group' => '',
```

```
'Module' => 'Kernel::Output::HTML::Dashboard::EventsTicketCalendar',
'Title' => 'Events Ticket Calendar'
};
```

### AgentCustomerInformationCenter::Backend###0100-CIC-TicketPendingReminder

Parameters for the dashboard backend of the ticket pending reminder overview of the agent interface . "Limit" is the number of entries shown by default. "Group" is used to restrict the access to the plugin (e. g. Group: admin;group1;group2;). "Default" determines if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" is the cache time in minutes for the plugin. Note: Only Ticket attributes and Dynamic Fields (DynamicField\_NameX) are allowed for DefaultColumns. Possible settings: 0 = Disabled, 1 = Available, 2 = Enabled by default.

默认值:

```
$Self->{'AgentCustomerInformationCenter::Backend'}->{'0100-CIC-TicketPendingReminder'}
= {
  'Attributes' => 'TicketPendingTimeOlderMinutes=1;StateType=pending
reminder;SortBy=PendingTime;OrderBy=Down;',
  'Block' => 'ContentLarge',
  'CacheTTLLocal' => '0.5',
  'Default' => '1',
  'DefaultColumns' => {
    'Age' => '2',
    'Changed' => '1',
    'Created' => '1',
    'CustomerCompanyName' => '1',
    'CustomerID' => '1',
    'CustomerName' => '1',
    'CustomerUserID' => '1',
    'EscalationResponseTime' => '1',
    'EscalationSolutionTime' => '1',
    'EscalationTime' => '1',
    'EscalationUpdateTime' => '1',
    'Lock' => '1',
    'Owner' => '1',
    'PendingTime' => '1',
    'Priority' => '1',
    'Queue' => '1',
    'Responsible' => '1',
    'SLA' => '1',
    'Service' => '1',
    'State' => '1',
    'TicketNumber' => '2',
    'Title' => '2',
    'Type' => '1'
  },
  'Description' => 'All tickets with a reminder set where the reminder date has been
reached',
  'Filter' => 'Locked',
  'Group' => '',
  'Limit' => '10',
  'Module' => 'Kernel::Output::HTML::Dashboard::TicketGeneric',
  'Permission' => 'ro',
  'Time' => 'UntilTime',
  'Title' => 'Reminder Tickets'
};
```

### AgentCustomerInformationCenter::Backend###0110-CIC-TicketEscalation

Parameters for the dashboard backend of the ticket escalation overview of the agent interface . "Limit" is the number of entries shown by default. "Group" is used to restrict the access to the plugin (e. g. Group: admin;group1;group2;). "Default" determines if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" is the cache time in minutes for the plugin. Note: Only Ticket

attributes and Dynamic Fields (DynamicField\_NameX) are allowed for DefaultColumns. Possible settings: 0 = Disabled, 1 = Available, 2 = Enabled by default.

默认值:

```
$Self->{'AgentCustomerInformationCenter::Backend'}->{'0110-CIC-TicketEscalation'} = {
  'Attributes' =>
  'TicketEscalationTimeOlderMinutes=1;SortBy=EscalationTime;OrderBy=Down;',
  'Block' => 'ContentLarge',
  'CacheTTLLocal' => '0.5',
  'Default' => '1',
  'DefaultColumns' => {
    'Age' => '2',
    'Changed' => '1',
    'Created' => '1',
    'CustomerCompanyName' => '1',
    'CustomerID' => '1',
    'CustomerName' => '1',
    'CustomerUserID' => '1',
    'EscalationResponseTime' => '1',
    'EscalationSolutionTime' => '1',
    'EscalationTime' => '1',
    'EscalationUpdateTime' => '1',
    'Lock' => '1',
    'Owner' => '1',
    'PendingTime' => '1',
    'Priority' => '1',
    'Queue' => '1',
    'Responsible' => '1',
    'SLA' => '1',
    'Service' => '1',
    'State' => '1',
    'TicketNumber' => '2',
    'Title' => '2',
    'Type' => '1'
  },
  'Description' => 'All escalated tickets',
  'Filter' => 'All',
  'Group' => '',
  'Limit' => '10',
  'Module' => 'Kernel::Output::HTML::Dashboard::TicketGeneric',
  'Permission' => 'ro',
  'Time' => 'EscalationTime',
  'Title' => 'Escalated Tickets'
};
```

### AgentCustomerInformationCenter::Backend###0120-CIC-TicketNew

Parameters for the dashboard backend of the new tickets overview of the agent interface. "Limit" is the number of entries shown by default. "Group" is used to restrict the access to the plugin (e. g. Group: admin;group1;group2;). "Default" determines if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" is the cache time in minutes for the plugin. Note: Only Ticket attributes and Dynamic Fields (DynamicField\_NameX) are allowed for DefaultColumns. Possible settings: 0 = Disabled, 1 = Available, 2 = Enabled by default.

默认值:

```
$Self->{'AgentCustomerInformationCenter::Backend'}->{'0120-CIC-TicketNew'} = {
  'Attributes' => 'StateType=new;',
  'Block' => 'ContentLarge',
  'CacheTTLLocal' => '0.5',
  'Default' => '1',
  'DefaultColumns' => {
    'Age' => '2',
    'Changed' => '1',
    'Created' => '1',
    'CustomerCompanyName' => '1',
    'CustomerID' => '1',
    'CustomerName' => '1',
```

```

'CustomerUserID' => '1',
'EscalationResponseTime' => '1',
'EscalationSolutionTime' => '1',
'EscalationTime' => '1',
'EscalationUpdateTime' => '1',
'Lock' => '1',
'Owner' => '1',
'PendingTime' => '1',
'Priority' => '1',
'Queue' => '1',
'Responsible' => '1',
'SLA' => '1',
'Service' => '1',
'State' => '1',
'TicketNumber' => '2',
'Title' => '2',
'Type' => '1'
},
'Description' => 'All new tickets, these tickets have not been worked on yet',
'Filter' => 'All',
'Group' => '',
'Limit' => '10',
'Module' => 'Kernel::Output::HTML::Dashboard::TicketGeneric',
'Permission' => 'ro',
'Time' => 'Age',
'Title' => 'New Tickets'
};

```

### AgentCustomerInformationCenter::Backend###0130-CIC-TicketOpen

Parameters for the dashboard backend of the open tickets overview of the agent interface. "Limit" is the number of entries shown by default. "Group" is used to restrict the access to the plugin (e. g. Group: admin;group1;group2;). "Default" determines if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" is the cache time in minutes for the plugin. Note: Only Ticket attributes and Dynamic Fields (DynamicField\_NameX) are allowed for DefaultColumns. Possible settings: 0 = Disabled, 1 = Available, 2 = Enabled by default.

默认值:

```

$self->{'AgentCustomerInformationCenter::Backend'}->{'0130-CIC-TicketOpen'} = {
  'Attributes' => 'StateType=open;',
  'Block' => 'ContentLarge',
  'CacheTTLLocal' => '0.5',
  'Default' => '1',
  'DefaultColumns' => {
    'Age' => '2',
    'Changed' => '1',
    'Created' => '1',
    'CustomerCompanyName' => '1',
    'CustomerID' => '1',
    'CustomerName' => '1',
    'CustomerUserID' => '1',
    'EscalationResponseTime' => '1',
    'EscalationSolutionTime' => '1',
    'EscalationTime' => '1',
    'EscalationUpdateTime' => '1',
    'Lock' => '1',
    'Owner' => '1',
    'PendingTime' => '1',
    'Priority' => '1',
    'Queue' => '1',
    'Responsible' => '1',
    'SLA' => '1',
    'Service' => '1',
    'State' => '1',
    'TicketNumber' => '2',
    'Title' => '2',
    'Type' => '1'
  },
};

```

```
'Description' => 'All open tickets, these tickets have already been worked on, but
need a response',
'Filter' => 'All',
'Group' => '',
'Limit' => '10',
'Module' => 'Kernel::Output::HTML::Dashboard::TicketGeneric',
'Permission' => 'ro',
'Time' => 'Age',
'Title' => 'Open Tickets / Need to be answered'
};
```

### AgentCustomerInformationCenter::Backend###0500-CIC-CustomerIDStatus

Parameters for the dashboard backend of the customer id status widget of the agent interface. "Group" is used to restrict the access to the plugin (e. g. Group: admin;group1;group2;). "Default" determines if the plugin is enabled by default or if the user needs to enable it manually. "CacheTTLLocal" is the cache time in minutes for the plugin.

默认值:

```
$Self->{'AgentCustomerInformationCenter::Backend'}->{'0500-CIC-CustomerIDStatus'} = {
  'Attributes' => '',
  'Block' => 'ContentSmall',
  'CacheTTLLocal' => '0.5',
  'Default' => '1',
  'Description' => 'Company Status',
  'Group' => '',
  'Module' => 'Kernel::Output::HTML::Dashboard::CustomerIDStatus',
  'Permission' => 'ro',
  'Title' => 'Company Status'
};
```

### Ticket → Frontend::Agent::Dashboard::EventsTicketCalendar

#### DashboardEventsTicketCalendar###CalendarWidth

Defines the calendar width in percent. Default is 95%.

不能使这个设置失效。

默认值:

```
$Self->{'DashboardEventsTicketCalendar'}->{'CalendarWidth'} = '95';
```

#### DashboardEventsTicketCalendar###Queues

Defines queues that's tickets are used for displaying as calendar events.

不能使这个设置失效。

默认值:

```
$Self->{'DashboardEventsTicketCalendar'}->{'Queues'} = [
  'Raw'
];
```

#### DashboardEventsTicketCalendar::DynamicFieldStartTime

Define dynamic field name for start time. This field has to be manually added to the system as Ticket: "Date / Time" and must be activated in ticket creation screens and/or in any other ticket action screens.

默认值:

```
$Self->{'DashboardEventsTicketCalendar::DynamicFieldStartTime'} =
'TicketCalendarStartTime';
```

## DashboardEventsTicketCalendar::DynamicFieldEndTime

Define dynamic field name for end time. This field has to be manually added to the system as Ticket: "Date / Time" and must be activated in ticket creation screens and/or in any other ticket action screens.

默认值:

```
$Self->{'DashboardEventsTicketCalendar::DynamicFieldEndTime'} =
'TicketCalendarEndTime';
```

## DashboardEventsTicketCalendar::DynamicFieldsForEvents

Defines the dynamic fields that are used for displaying on calendar events.

不能使这个设置失效。

默认值:

```
$Self->{'DashboardEventsTicketCalendar::DynamicFieldsForEvents'} = [
'TicketCalendarStartTime',
'TicketCalendarEndTime'
];
```

## DashboardEventsTicketCalendar::TicketFieldsForEvents

Defines the ticket fields that are going to be displayed calendar events. The "Key" defines the field or ticket attribute and the "Content" defines the display name.

不能使这个设置失效。

默认值:

```
$Self->{'DashboardEventsTicketCalendar::TicketFieldsForEvents'} = {
'CustomerID' => 'Customer ID',
'CustomerUserID' => 'Customer user',
'Priority' => 'Priority',
'Queue' => 'Queue',
'SLA' => 'SLA',
'Service' => 'Service',
'State' => 'State',
'Title' => 'Title',
'Type' => 'Type'
};
```

## Ticket → Frontend::Agent::Dashboard::TicketFilters

### OnlyValuesOnTicket

Defines if the values for filters should be retrieved from all available tickets. If set to "Yes", only values which are actually used in any ticket will be available for filtering. Please note: The list of customers will always be retrieved like this.

不能使这个设置失效。

默认值:

```
$Self->{'OnlyValuesOnTicket'} = '1';
```

## Ticket → Frontend::Agent::LinkObject

### LinkObject::ComplexTable::SettingsVisibility###Ticket

Define Actions where a settings button is available in the linked objects widget (LinkObject::ViewMode = "complex"). Please note that these Actions



must have registered the following JS and CSS files: Core.AllocationList.css, Core.UI.AllocationList.js, Core.UI.Table.Sort.js, Core.Agent.TableFilters.js.

默认值:

```
$Self->{'LinkObject::ComplexTable::SettingsVisibility'}->{'Ticket'} = [
  'AgentTicketZoom'
];
```

### LinkObject::ComplexTable###Ticket

Define which columns are shown in the linked tickets widget (LinkObject::ViewMode = "complex"). Note: Only Ticket attributes and Dynamic Fields (DynamicField\_NameX) are allowed for DefaultColumns. Possible settings: 0 = Disabled, 1 = Available, 2 = Enabled by default.

默认值:

```
$Self->{'LinkObject::ComplexTable'}->{'Ticket'} = {
  'DefaultColumns' => {
    'Age' => '1',
    'Changed' => '1',
    'Created' => '2',
    'CustomerCompanyName' => '1',
    'CustomerID' => '1',
    'CustomerName' => '1',
    'CustomerUserID' => '1',
    'EscalationResponseTime' => '1',
    'EscalationSolutionTime' => '1',
    'EscalationTime' => '1',
    'EscalationUpdateTime' => '1',
    'Lock' => '1',
    'Owner' => '1',
    'PendingTime' => '1',
    'Priority' => '1',
    'Queue' => '2',
    'Responsible' => '1',
    'SLA' => '1',
    'Service' => '1',
    'State' => '2',
    'TicketNumber' => '2',
    'Title' => '2',
    'Type' => '1'
  },
  'Module' => 'Kernel::Output::HTML::LinkObject::Ticket.pm',
  'Priority' => {
    'Age' => '110',
    'Changed' => '120',
    'Created' => '310',
    'CustomerCompanyName' => '320',
    'CustomerID' => '240',
    'CustomerName' => '250',
    'CustomerUserID' => '260',
    'EscalationResponseTime' => '160',
    'EscalationSolutionTime' => '150',
    'EscalationTime' => '140',
    'EscalationUpdateTime' => '170',
    'Lock' => '200',
    'Owner' => '220',
    'PendingTime' => '130',
    'Priority' => '300',
    'Queue' => '210',
    'Responsible' => '230',
    'SLA' => '290',
    'Service' => '280',
    'State' => '190',
    'TicketNumber' => '100',
    'Title' => '180',
    'Type' => '270'
  }
}
```

```
};
```

## Ticket → Frontend::Agent::ModuleMetaHead

### Frontend::HeaderMetaModule###2-TicketSearch

Module to generate html OpenSearch profile for short ticket search in the agent interface.

默认值:

```
$Self->{'Frontend::HeaderMetaModule'}->{'2-TicketSearch'} = {  
  'Action' => 'AgentTicketSearch',  
  'Module' => 'Kernel::Output::HTML::HeaderMeta::AgentTicketSearch'  
};
```

## Ticket → Frontend::Agent::ModuleNotify

### Frontend::NotifyModule###5-Ticket::TicketEscalation

Module to show notifications and escalations (ShownMax: max. shown escalations, EscalationInMinutes: Show ticket which will escalation in, CacheTime: Cache of calculated escalations in seconds).

这个设置默认没有激活。

默认值:

```
$Self->{'Frontend::NotifyModule'}->{'5-Ticket::TicketEscalation'} = {  
  'CacheTime' => '40',  
  'EscalationInMinutes' => '120',  
  'Module' => 'Kernel::Output::HTML::Notification::AgentTicketEscalation',  
  'ShownMax' => '25'  
};
```

## Ticket → Frontend::Agent::ModuleRegistration

### Frontend::Module###AgentTicketQueue

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AgentTicketQueue'} = {  
  'Description' => 'Overview of all open Tickets.',  
  'Loader' => {  
    'CSS' => [  
      'Core.AgentTicketQueue.css',  
      'Core.AllocationList.css'  
    ],  
    'JavaScript' => [  
      'Core.UI.AllocationList.js',  
      'Core.Agent.TableFilters.js'  
    ]  
  },  
  'NavBar' => [  
    {  
      'AccessKey' => 'o',  
      'Block' => '',  
      'Description' => 'Overview of all open Tickets.',  
      'Link' => 'Action=AgentTicketQueue',  
      'LinkOption' => '',  
      'Name' => 'Queue view',  
      'NavBar' => 'Ticket',  
      'Prio' => '100',  
      'Type' => ''  
    },  
  ],  
};
```

```

{
  'AccessKey' => 't',
  'Block' => 'ItemArea',
  'Description' => '',
  'Link' => 'Action=AgentTicketQueue',
  'LinkOption' => '',
  'Name' => 'Tickets',
  'NavBar' => 'Ticket',
  'Prio' => '200',
  'Type' => 'Menu'
}
],
'NavBarName' => 'Ticket',
'Title' => 'QueueView'
};

```

### Frontend::Module###AgentTicketService

Frontend module registration for the agent interface.

默认值:

```

$self->{'Frontend::Module'}->{'AgentTicketService'} = {
  'Description' => 'Overview of all open Tickets.',
  'Loader' => {
    'CSS' => [
      'Core.AgentTicketService.css',
      'Core.AllocationList.css'
    ],
    'JavaScript' => [
      'Core.UI.AllocationList.js',
      'Core.Agent.TableFilters.js'
    ]
  },
  'NavBar' => [
    {
      'AccessKey' => '0',
      'Block' => '',
      'Description' => 'Overview of all open Tickets.',
      'Link' => 'Action=AgentTicketService',
      'LinkOption' => '',
      'Name' => 'Service view',
      'NavBar' => 'Ticket',
      'Prio' => '105',
      'Type' => ''
    }
  ],
  'NavBarName' => 'Ticket',
  'Title' => 'ServiceView'
};

```

### Frontend::Module###AgentTicketPhone

Frontend module registration for the agent interface.

默认值:

```

$self->{'Frontend::Module'}->{'AgentTicketPhone'} = {
  'Description' => 'Create new phone ticket.',
  'Loader' => {
    'JavaScript' => [
      'Core.Agent.CustomerSearch.js',
      'Core.Agent.TicketAction.js'
    ]
  },
  'NavBar' => [
    {
      'AccessKey' => 'n',
      'Block' => '',
      'Description' => 'Create new phone ticket (inbound).',
      'Link' => 'Action=AgentTicketPhone',

```

```

    'LinkOption' => '',
    'Name' => 'New phone ticket',
    'NavBar' => 'Ticket',
    'Prio' => '200',
    'Type' => ''
  }
],
'NavBarName' => 'Ticket',
'Title' => 'New phone ticket'
};

```

### Frontend::Module###AgentTicketPhoneOutbound

Frontend module registration for the agent interface.

默认值:

```

$self->{'Frontend::Module'}->{'AgentTicketPhoneOutbound'} = {
  'Description' => 'Phone Call.',
  'Loader' => {
    'JavaScript' => [
      'Core.Agent.TicketAction.js'
    ]
  },
  'NavBarName' => 'Ticket',
  'Title' => 'Phone-Ticket'
};

```

### Frontend::Module###AgentTicketPhoneInbound

Frontend module registration for the agent interface.

默认值:

```

$self->{'Frontend::Module'}->{'AgentTicketPhoneInbound'} = {
  'Description' => 'Incoming Phone Call.',
  'Loader' => {
    'JavaScript' => [
      'Core.Agent.TicketAction.js'
    ]
  },
  'NavBarName' => 'Ticket',
  'Title' => 'Phone-Ticket'
};

```

### Frontend::Module###AgentTicketEmail

Frontend module registration for the agent interface.

默认值:

```

$self->{'Frontend::Module'}->{'AgentTicketEmail'} = {
  'Description' => 'Create new email ticket.',
  'Loader' => {
    'JavaScript' => [
      'Core.Agent.CustomerSearch.js',
      'Core.Agent.TicketAction.js'
    ]
  },
  'NavBar' => [
    {
      'AccessKey' => 'm',
      'Block' => '',
      'Description' => 'Create new email ticket and send this out (outbound).',
      'Link' => 'Action=AgentTicketEmail',
      'LinkOption' => '',
      'Name' => 'New email ticket',
      'NavBar' => 'Ticket',
      'Prio' => '210',
      'Type' => ''
    }
  ]
};

```

```

    }
  ],
  'NavBarName' => 'Ticket',
  'Title' => 'New email ticket'
};

```

### Frontend::Module###AgentTicketSearch

Frontend module registration for the agent interface.

默认值:

```

$self->{'Frontend::Module'}->{'AgentTicketSearch'} = {
  'Description' => 'Search Ticket.',
  'Loader' => {
    'JavaScript' => [
      'Core.UI.AllocationList.js',
      'Core.Agent.TableFilters.js'
    ]
  },
  'NavBar' => [
    {
      'AccessKey' => 's',
      'Block' => '',
      'Description' => 'Search Tickets.',
      'Link' => 'Action=AgentTicketSearch',
      'LinkOption' => 'onclick="window.setTimeout(function()
{Core.Agent.Search.OpenSearchDialog(\'AgentTicketSearch\');}, 0); return false;"',
      'Name' => 'Search',
      'NavBar' => 'Ticket',
      'Prio' => '300',
      'Type' => ''
    }
  ],
  'NavBarName' => 'Ticket',
  'Title' => 'Search'
};

```

### Frontend::Module###AgentTicketLockedView

Frontend module registration for the agent interface.

默认值:

```

$self->{'Frontend::Module'}->{'AgentTicketLockedView'} = {
  'Description' => 'Locked Tickets.',
  'Loader' => {
    'CSS' => [
      'Core.AgentTicketQueue.css',
      'Core.AllocationList.css'
    ],
    'JavaScript' => [
      'Core.UI.AllocationList.js',
      'Core.Agent.TableFilters.js'
    ]
  },
  'NavBarName' => 'Ticket',
  'Title' => 'Locked Tickets'
};

```

### Frontend::Module###AgentTicketResponsibleView

Frontend module registration for the agent interface.

默认值:

```

$self->{'Frontend::Module'}->{'AgentTicketResponsibleView'} = {
  'Description' => 'Responsible Tickets.',
  'Loader' => {
    'CSS' => [

```

```

    'Core.AllocationList.css'
  ],
  'JavaScript' => [
    'Core.UI.AllocationList.js',
    'Core.Agent.TableFilters.js'
  ]
},
'NavBarName' => 'Ticket',
'Title' => 'Responsible Tickets'
};

```

### Frontend::Module###AgentTicketWatchView

Frontend module registration for the agent interface.

默认值:

```

$self->{'Frontend::Module'}->{'AgentTicketWatchView'} = {
  'Description' => 'Watched Tickets.',
  'Loader' => {
    'CSS' => [
      'Core.AgentTicketQueue.css',
      'Core.AllocationList.css'
    ],
    'JavaScript' => [
      'Core.UI.AllocationList.js',
      'Core.Agent.TableFilters.js'
    ]
  },
  'NavBarName' => 'Ticket',
  'Title' => 'Watched Tickets'
};

```

### Frontend::Module###AgentCustomerSearch

Frontend module registration for the agent interface.

默认值:

```

$self->{'Frontend::Module'}->{'AgentCustomerSearch'} = {
  'Description' => 'AgentCustomerSearch.',
  'NavBarName' => 'Ticket',
  'Title' => 'AgentCustomerSearch'
};

```

### Frontend::Module###AgentUserSearch

Frontend module registration for the agent interface.

默认值:

```

$self->{'Frontend::Module'}->{'AgentUserSearch'} = {
  'Description' => 'AgentUserSearch.',
  'NavBarName' => 'Ticket',
  'Title' => 'AgentUserSearch'
};

```

### Frontend::Module###AgentTicketStatusView

Frontend module registration for the agent interface.

默认值:

```

$self->{'Frontend::Module'}->{'AgentTicketStatusView'} = {
  'Description' => 'Overview of all open tickets.',
  'Loader' => {
    'CSS' => [
      'Core.AllocationList.css'
    ],
    'JavaScript' => [

```

```

    'Core.UI.AllocationList.js',
    'Core.Agent.TableFilters.js'
  ]
},
'NavBar' => [
  {
    'AccessKey' => 'v',
    'Block' => '',
    'Description' => 'Overview of all open Tickets.',
    'Link' => 'Action=AgentTicketStatusView',
    'LinkOption' => '',
    'Name' => 'Status view',
    'NavBar' => 'Ticket',
    'Prio' => '110',
    'Type' => ''
  }
],
'NavBarName' => 'Ticket',
'Title' => 'Status view'
};

```

### Frontend::Module###AgentTicketEscalationView

Frontend module registration for the agent interface.

默认值:

```

$self->{'Frontend::Module'}->{'AgentTicketEscalationView'} = {
  'Description' => 'Overview of all escalated tickets.',
  'Loader' => {
    'CSS' => [
      'Core.AllocationList.css'
    ],
    'JavaScript' => [
      'Core.UI.AllocationList.js',
      'Core.Agent.TableFilters.js'
    ]
  },
  'NavBar' => [
    {
      'AccessKey' => 'e',
      'Block' => '',
      'Description' => 'Overview Escalated Tickets.',
      'Link' => 'Action=AgentTicketEscalationView',
      'LinkOption' => '',
      'Name' => 'Escalation view',
      'NavBar' => 'Ticket',
      'Prio' => '120',
      'Type' => ''
    }
  ],
  'NavBarName' => 'Ticket',
  'Title' => 'Escalation view'
};

```

### Frontend::Module###AgentZoom

Frontend module registration for the agent interface.

默认值:

```

$self->{'Frontend::Module'}->{'AgentZoom'} = {
  'Description' => 'Compat module for AgentZoom to AgentTicketZoom.',
  'NavBarName' => 'Ticket',
  'Title' => ''
};

```

### Frontend::Module###AgentTicketZoom

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AgentTicketZoom'} = {  
  'Description' => 'Ticket Zoom.',  
  'Loader' => {  
    'CSS' => [  
      'Core.Agent.TicketProcess.css',  
      'Core.Agent.TicketMenuModuleCluster.css',  
      'Core.AllocationList.css'  
    ],  
    'JavaScript' => [  
      'thirdparty/jquery-tablesorter-2.0.5/jquery.tablesorter.js',  
      'Core.Agent.TicketZoom.js',  
      'Core.UI.AllocationList.js',  
      'Core.UI.Table.Sort.js',  
      'Core.Agent.TableFilters.js',  
      'Core.Agent.LinkObject.js'  
    ]  
  },  
  'NavBarName' => 'Ticket',  
  'Title' => 'Zoom'  
};
```

### Frontend::Module###AgentTicketAttachment

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AgentTicketAttachment'} = {  
  'Description' => 'To download attachments.',  
  'NavBarName' => 'Ticket',  
  'Title' => ''  
};
```

### Frontend::Module###AgentTicketPlain

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AgentTicketPlain'} = {  
  'Description' => 'Ticket plain view of an email.',  
  'NavBarName' => 'Ticket',  
  'Title' => 'Plain'  
};
```

### Frontend::Module###AgentTicketNote

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AgentTicketNote'} = {  
  'Description' => 'Ticket Note.',  
  'Loader' => {  
    'JavaScript' => [  
      'Core.Agent.TicketAction.js'  
    ]  
  },  
  'NavBarName' => 'Ticket',  
  'Title' => 'Note'  
};
```

### Frontend::Module###AgentTicketMerge

Frontend module registration for the agent interface.

默认值:



```
$Self->{'Frontend::Module'}->{'AgentTicketMerge'} = {  
  'Description' => 'Ticket Merge.',  
  'Loader' => {  
    'JavaScript' => [  
      'Core.Agent.TicketMerge.js'  
    ]  
  },  
  'NavBarName' => 'Ticket',  
  'Title' => 'Merge'  
};
```

### Frontend::Module###AgentTicketPending

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AgentTicketPending'} = {  
  'Description' => 'Ticket Pending.',  
  'Loader' => {  
    'JavaScript' => [  
      'Core.Agent.TicketAction.js'  
    ]  
  },  
  'NavBarName' => 'Ticket',  
  'Title' => 'Pending'  
};
```

### Frontend::Module###AgentTicketWatcher

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AgentTicketWatcher'} = {  
  'Description' => 'A TicketWatcher Module.',  
  'NavBarName' => 'Ticket-Watcher',  
  'Title' => 'Ticket Watcher'  
};
```

### Frontend::Module###AgentTicketPriority

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AgentTicketPriority'} = {  
  'Description' => 'Ticket Priority.',  
  'Loader' => {  
    'JavaScript' => [  
      'Core.Agent.TicketAction.js'  
    ]  
  },  
  'NavBarName' => 'Ticket',  
  'Title' => 'Priority'  
};
```

### Frontend::Module###AgentTicketLock

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AgentTicketLock'} = {  
  'Description' => 'Ticket Lock.',  
  'NavBarName' => 'Ticket',  
  'Title' => 'Lock'  
};
```

### Frontend::Module###AgentTicketMove

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AgentTicketMove'} = {  
  'Description' => 'Ticket Move.',  
  'Loader' => {  
    'JavaScript' => [  
      'Core.Agent.TicketAction.js'  
    ]  
  },  
  'NavBarName' => 'Ticket',  
  'Title' => 'Move'  
};
```

### Frontend::Module###AgentTicketHistory

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AgentTicketHistory'} = {  
  'Description' => 'Ticket History.',  
  'NavBarName' => 'Ticket',  
  'Title' => 'History'  
};
```

### Frontend::Module###AgentTicketOwner

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AgentTicketOwner'} = {  
  'Description' => 'Ticket Owner.',  
  'Loader' => {  
    'JavaScript' => [  
      'Core.Agent.TicketAction.js'  
    ]  
  },  
  'NavBarName' => 'Ticket',  
  'Title' => 'Owner'  
};
```

### Frontend::Module###AgentTicketResponsible

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AgentTicketResponsible'} = {  
  'Description' => 'Ticket Responsible.',  
  'Loader' => {  
    'JavaScript' => [  
      'Core.Agent.TicketAction.js'  
    ]  
  },  
  'NavBarName' => 'Ticket',  
  'Title' => 'Responsible'  
};
```

### Frontend::Module###AgentTicketCompose

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AgentTicketCompose'} = {  
  'Description' => 'Ticket Compose email Answer.',  
  'Loader' => {  
    'JavaScript' => [  
      'Core.Agent.CustomerSearch.js',  
      'Core.Agent.TicketAction.js'  
    ]  
  },  
  'NavBarName' => 'Ticket',  
  'Title' => 'Compose'  
};
```

### Frontend::Module###AgentTicketBounce

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AgentTicketBounce'} = {  
  'Description' => 'Ticket Compose Bounce Email.',  
  'NavBarName' => 'Ticket',  
  'Title' => 'Bounce'  
};
```

### Frontend::Module###AgentTicketForward

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AgentTicketForward'} = {  
  'Description' => 'Ticket Forward Email.',  
  'Loader' => {  
    'JavaScript' => [  
      'Core.Agent.CustomerSearch.js',  
      'Core.Agent.TicketAction.js'  
    ]  
  },  
  'NavBarName' => 'Ticket',  
  'Title' => 'Forward'  
};
```

### Frontend::Module###AgentTicketEmailOutbound

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AgentTicketEmailOutbound'} = {  
  'Description' => 'Ticket Outbound Email.',  
  'Loader' => {  
    'JavaScript' => [  
      'Core.Agent.CustomerSearch.js',  
      'Core.Agent.TicketAction.js'  
    ]  
  },  
  'NavBarName' => 'Ticket',  
  'Title' => 'Email Outbound'  
};
```

### Frontend::Module###AgentTicketCustomer

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AgentTicketCustomer'} = {  
  'Description' => 'Ticket Customer.',  
  'Loader' => {
```

```
'JavaScript' => [  
  'Core.Agent.CustomerSearch.js',  
  'Core.Agent.TicketAction.js'  
],  
,  
'NavBarName' => 'Ticket',  
'Title' => 'Customer'  
};
```

### Frontend::Module###AgentTicketClose

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AgentTicketClose'} = {  
  'Description' => 'Ticket Close.',  
  'Loader' => {  
    'JavaScript' => [  
      'Core.Agent.TicketAction.js'  
    ]  
  },  
  'NavBarName' => 'Ticket',  
  'Title' => 'Close'  
};
```

### Frontend::Module###AgentTicketFreeText

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AgentTicketFreeText'} = {  
  'Description' => 'Ticket FreeText.',  
  'Loader' => {  
    'JavaScript' => [  
      'Core.Agent.TicketAction.js'  
    ]  
  },  
  'NavBarName' => 'Ticket',  
  'Title' => 'Free Fields'  
};
```

### Frontend::Module###AgentTicketPrint

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AgentTicketPrint'} = {  
  'Description' => 'Ticket Print.',  
  'NavBarName' => 'Ticket',  
  'Title' => 'Print'  
};
```

### Frontend::Module###AgentTicketBulk

Frontend module registration for the agent interface.

默认值:

```
$Self->{'Frontend::Module'}->{'AgentTicketBulk'} = {  
  'Description' => 'Ticket bulk module.',  
  'Loader' => {  
    'JavaScript' => [  
      'Core.Agent.TicketAction.js'  
    ]  
  },  
  'NavBarName' => 'Ticket',  
  'Title' => 'Bulk Action'  
};
```

```
};
```

## Ticket → Frontend::Agent::Preferences

### PreferencesGroups###CustomQueue

Parameters for the CustomQueue object in the preference view of the agent interface.

默认值:

```
$Self->{'PreferencesGroups'}->{'CustomQueue'} = {
  'Active' => '1',
  'Column' => 'Notification Settings',
  'Desc' => 'Your queue selection of your preferred queues. You also get notified about
those queues via email if enabled.',
  'Key' => '',
  'Label' => 'My Queues',
  'Module' => 'Kernel::Output::HTML::Preferences::CustomQueue',
  'Permission' => 'ro',
  'Prio' => '1000'
};
```

### PreferencesGroups###CustomService

Parameters for the CustomService object in the preference view of the agent interface.

默认值:

```
$Self->{'PreferencesGroups'}->{'CustomService'} = {
  'Active' => '1',
  'Column' => 'Notification Settings',
  'Desc' => 'Your service selection of your preferred services. You also get notified
about those services via email if enabled.',
  'Key' => '',
  'Label' => 'My Services',
  'Module' => 'Kernel::Output::HTML::Preferences::CustomService',
  'Prio' => '1000'
};
```

### PreferencesGroups###RefreshTime

Parameters for the RefreshTime object in the preference view of the agent interface.

默认值:

```
$Self->{'PreferencesGroups'}->{'RefreshTime'} = {
  'Active' => '1',
  'Column' => 'Other Settings',
  'Data' => {
    '0' => 'off',
    '10' => '10 minutes',
    '15' => '15 minutes',
    '2' => ' 2 minutes',
    '5' => ' 5 minutes',
    '7' => ' 7 minutes'
  },
  'DataSelected' => '0',
  'Desc' => 'If enabled, the different overviews (Dashboard, LockedView, QueueView) will
automatically refresh after the specified time.',
  'Key' => 'After',
  'Label' => 'Overview Refresh Time',
  'Module' => 'Kernel::Output::HTML::Preferences::Generic',
  'PrefKey' => 'UserRefreshTime',
  'Prio' => '2000'
};
```

### PreferencesGroups###TicketOverviewSmallPageShown

Parameters for the pages (in which the tickets are shown) of the small ticket overview.

默认值:

```
$Self->{'PreferencesGroups'}->{'TicketOverviewSmallPageShown'} = {
  'Active' => '0',
  'Column' => 'Other Settings',
  'Data' => {
    '10' => '10',
    '15' => '15',
    '20' => '20',
    '25' => '25',
    '30' => '30',
    '35' => '35'
  },
  'DataSelected' => '25',
  'Key' => 'Ticket limit per page for Ticket Overview "Small"',
  'Label' => 'Ticket Overview "Small" Limit',
  'Module' => 'Kernel::Output::HTML::Preferences::Generic',
  'PrefKey' => 'UserTicketOverviewSmallPageShown',
  'Prio' => '8000'
};
```

### PreferencesGroups###TicketOverviewFilterSettings

Parameters for the column filters of the small ticket overview.

默认值:

```
$Self->{'PreferencesGroups'}->{'TicketOverviewFilterSettings'} = {
  'Active' => '0',
  'Column' => 'Other Settings',
  'Key' => 'Column ticket filters for Ticket Overviews type "Small".',
  'Label' => 'Enabled filters.',
  'Module' => 'Kernel::Output::HTML::Preferences::ColumnFilters',
  'PrefKey' => 'UserFilterColumnsEnabled',
  'Prio' => '8100'
};
```

### PreferencesGroups###TicketOverviewMediumPageShown

Parameters for the pages (in which the tickets are shown) of the medium ticket overview.

默认值:

```
$Self->{'PreferencesGroups'}->{'TicketOverviewMediumPageShown'} = {
  'Active' => '0',
  'Column' => 'Other Settings',
  'Data' => {
    '10' => '10',
    '15' => '15',
    '20' => '20',
    '25' => '25',
    '30' => '30',
    '35' => '35'
  },
  'DataSelected' => '20',
  'Key' => 'Ticket limit per page for Ticket Overview "Medium"',
  'Label' => 'Ticket Overview "Medium" Limit',
  'Module' => 'Kernel::Output::HTML::Preferences::Generic',
  'PrefKey' => 'UserTicketOverviewMediumPageShown',
  'Prio' => '8100'
};
```

### PreferencesGroups###TicketOverviewPreviewPageShown

Parameters for the pages (in which the tickets are shown) of the ticket preview overview.

默认值:

```
$Self->{'PreferencesGroups'}->{'TicketOverviewPreviewPageShown'} = {
  'Active' => '0',
  'Column' => 'Other Settings',
  'Data' => {
    '10' => '10',
    '15' => '15',
    '20' => '20',
    '25' => '25',
    '30' => '30',
    '35' => '35'
  },
  'DataSelected' => '15',
  'Key' => 'Ticket limit per page for Ticket Overview "Preview"',
  'Label' => 'Ticket Overview "Preview" Limit',
  'Module' => 'Kernel::Output::HTML::Preferences::Generic',
  'PrefKey' => 'UserTicketOverviewPreviewPageShown',
  'Prio' => '8200'
};
```

### PreferencesGroups###CreateNextMask

Parameters for the CreateNextMask object in the preference view of the agent interface.

默认值:

```
$Self->{'PreferencesGroups'}->{'CreateNextMask'} = {
  'Active' => '1',
  'Column' => 'Other Settings',
  'Data' => {
    '0' => 'Create Ticket',
    'AgentTicketZoom' => 'Ticket Zoom'
  },
  'DataSelected' => '',
  'Desc' => 'Configure which screen should be shown after a new ticket has been created.',
  'Key' => 'Screen',
  'Label' => 'Screen after new ticket',
  'Module' => 'Kernel::Output::HTML::Preferences::Generic',
  'PrefKey' => 'UserCreateNextMask',
  'Prio' => '3000'
};
```

### PreferencesGroups###NotificationEvent

Transport selection for ticket notifications.

默认值:

```
$Self->{'PreferencesGroups'}->{'NotificationEvent'} = {
  'Active' => '1',
  'Column' => 'Notification Settings',
  'Desc' => 'Choose for which kind of ticket changes you want to receive notifications.',
  'Label' => 'Ticket notifications',
  'Module' => 'Kernel::Output::HTML::Preferences::NotificationEvent',
  'PrefKey' => 'AdminNotificationEventTransport',
  'Prio' => '8000'
};
```

## Ticket → Frontend::Agent::SearchRouter

### Frontend::Search###AgentCustomerInformationCenter

Search backend router.

默认值:

```
$Self->{'Frontend::Search'}->{'AgentCustomerInformationCenter'} = {
```

```
'^AgentCustomerInformationCenter' => 'Action=AgentCustomerInformationCenterSearch'
};
```

## Frontend::Search::JavaScript###AgentCustomerInformationCenter

JavaScript function for the search frontend.

默认值:

```
$Self->{'Frontend::Search::JavaScript'}->{'AgentCustomerInformationCenter'} = {
  '^AgentCustomerInformationCenter' =>
  'Core.Agent.CustomerInformationCenterSearch.OpenSearchDialog()'
};
```

## Frontend::Search###Ticket

Search backend router.

默认值:

```
$Self->{'Frontend::Search'}->{'Ticket'} = {
  '^AgentTicket' => 'Action=AgentTicketSearch;Subaction=AJAX'
};
```

## Ticket → Frontend::Agent::Ticket::ArticleAttachmentModule

### Ticket::Frontend::ArticleAttachmentModule###1-Download

Shows a link to download article attachments in the zoom view of the article in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::ArticleAttachmentModule'}->{'1-Download'} = {
  'Module' => 'Kernel::Output::HTML::ArticleAttachment::Download'
};
```

### Ticket::Frontend::ArticleAttachmentModule###2-HTML-Viewer

Shows a link to access article attachments via a html online viewer in the zoom view of the article in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::ArticleAttachmentModule'}->{'2-HTML-Viewer'} = {
  'Module' => 'Kernel::Output::HTML::ArticleAttachment::HTMLViewer'
};
```

## Ticket → Frontend::Agent::Ticket::ArticleComposeModule

### Ticket::Frontend::ArticleComposeModule###1-SignEmail

Module to compose signed messages (PGP or S/MIME).

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::ArticleComposeModule'}->{'1-SignEmail'} = {
  'Module' => 'Kernel::Output::HTML::ArticleCompose::Sign'
};
```



```
};
```

### **Ticket::Frontend::ArticleComposeModule###2-CryptEmail**

Module to crypt composed messages (PGP or S/MIME).

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::ArticleComposeModule'}->{'2-CryptEmail'} = {  
  'Module' => 'Kernel::Output::HTML::ArticleCompose::Crypt'  
};
```

### **Ticket → Frontend::Agent::Ticket::ArticleViewModule**

#### **Ticket::Frontend::ArticleViewModule###1-PGP**

Agent interface article notification module to check PGP.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::ArticleViewModule'}->{'1-PGP'} = {  
  'Module' => 'Kernel::Output::HTML::ArticleCheck::PGP'  
};
```

#### **Ticket::Frontend::ArticleViewModule###1-SMIME**

Agent interface module to check incoming emails in the Ticket-Zoom-View if the S/MIME-key is available and true.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::ArticleViewModule'}->{'1-SMIME'} = {  
  'Module' => 'Kernel::Output::HTML::ArticleCheck::SMIME'  
};
```

### **Ticket → Frontend::Agent::Ticket::ArticleViewModulePre**

#### **Ticket::Frontend::ArticlePreViewModule###1-PGP**

Agent interface article notification module to check PGP.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::ArticlePreViewModule'}->{'1-PGP'} = {  
  'Module' => 'Kernel::Output::HTML::ArticleCheck::PGP'  
};
```

#### **Ticket::Frontend::ArticlePreViewModule###1-SMIME**

Agent interface article notification module to check S/MIME.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::ArticlePreViewModule'}->{'1-SMIME'} = {
```

```
'Module' => 'Kernel::Output::HTML::ArticleCheck::SMIME'
};
```

## Ticket → Frontend::Agent::Ticket::MenuModule

### Ticket::Frontend::MenuModule###000-Back

Shows a link in the menu to go back in the ticket zoom view of the agent interface. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move\_into:group2". To cluster menu items use for Key "ClusterName" and for the Content any name you want to see in the UI. Use "ClusterPriority" to configure the order of a certain cluster within the toolbar.

默认值:

```
$Self->{'Ticket::Frontend::MenuModule'}->{'000-Back'} = {
  'Action' => '',
  'ClusterName' => '',
  'ClusterPriority' => '',
  'Description' => 'Go back',
  'Link' => "[% Env('LastScreenOverview') %];TicketID=[% Data.TicketID | html %]",
  'Module' => 'Kernel::Output::HTML::TicketMenu::Generic',
  'Name' => 'Back',
  'PopupType' => '',
  'Target' => ''
};
```

### Ticket::Frontend::MenuModule###100-Lock

Shows a link in the menu to lock/unlock tickets in the ticket zoom view of the agent interface. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move\_into:group2". To cluster menu items use for Key "ClusterName" and for the Content any name you want to see in the UI. Use "ClusterPriority" to configure the order of a certain cluster within the toolbar.

默认值:

```
$Self->{'Ticket::Frontend::MenuModule'}->{'100-Lock'} = {
  'Action' => 'AgentTicketLock',
  'ClusterName' => 'Miscellaneous',
  'ClusterPriority' => '800',
  'Description' => 'Lock / unlock this ticket',
  'Module' => 'Kernel::Output::HTML::TicketMenu::Lock',
  'Name' => 'Lock',
  'Target' => ''
};
```

### Ticket::Frontend::MenuModule###200-History

Shows a link in the menu to access the history of a ticket in the ticket zoom view of the agent interface. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move\_into:group2". To cluster menu items use for Key "ClusterName" and for the Content any name you want to see in the UI. Use "ClusterPriority" to configure the order of a certain cluster within the toolbar.

默认值:

```
$Self->{'Ticket::Frontend::MenuModule'}->{'200-History'} = {
  'Action' => 'AgentTicketHistory',
  'ClusterName' => 'Miscellaneous',
  'ClusterPriority' => '800',
  'Description' => 'Show the history for this ticket',
  'Link' => 'Action=AgentTicketHistory;TicketID=[% Data.TicketID | html %]',
  'Module' => 'Kernel::Output::HTML::TicketMenu::Generic',
  'Name' => 'History',
  'PopupType' => 'TicketHistory',
};
```

```
'Target' => ''
};
```

### **Ticket::Frontend::MenuModule###210-Print**

Shows a link in the menu to print a ticket or an article in the ticket zoom view of the agent interface. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move\_into:group2". To cluster menu items use for Key "ClusterName" and for the Content any name you want to see in the UI. Use "ClusterPriority" to configure the order of a certain cluster within the toolbar.

默认值:

```
$Self->{'Ticket::Frontend::MenuModule'}->{'210-Print'} = {
  'Action' => 'AgentTicketPrint',
  'ClusterName' => '',
  'ClusterPriority' => '',
  'Description' => 'Print this ticket',
  'Link' => 'Action=AgentTicketPrint;TicketID=[% Data.TicketID | html %]',
  'LinkParam' => 'target="print"',
  'Module' => 'Kernel::Output::HTML::TicketMenu::Generic',
  'Name' => 'Print',
  'PopupType' => 'TicketAction',
  'Target' => ''
};
```

### **Ticket::Frontend::MenuModule###300-Priority**

Shows a link in the menu to see the priority of a ticket in the ticket zoom view of the agent interface. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move\_into:group2". To cluster menu items use for Key "ClusterName" and for the Content any name you want to see in the UI. Use "ClusterPriority" to configure the order of a certain cluster within the toolbar.

默认值:

```
$Self->{'Ticket::Frontend::MenuModule'}->{'300-Priority'} = {
  'Action' => 'AgentTicketPriority',
  'ClusterName' => '',
  'ClusterPriority' => '',
  'Description' => 'Change the priority for this ticket',
  'Link' => 'Action=AgentTicketPriority;TicketID=[% Data.TicketID | html %]',
  'Module' => 'Kernel::Output::HTML::TicketMenu::Generic',
  'Name' => 'Priority',
  'PopupType' => 'TicketAction',
  'Target' => ''
};
```

### **Ticket::Frontend::MenuModule###310-FreeText**

Shows a link in the menu to add a free text field in the ticket zoom view of the agent interface. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move\_into:group2". To cluster menu items use for Key "ClusterName" and for the Content any name you want to see in the UI. Use "ClusterPriority" to configure the order of a certain cluster within the toolbar.

默认值:

```
$Self->{'Ticket::Frontend::MenuModule'}->{'310-FreeText'} = {
  'Action' => 'AgentTicketFreeText',
  'ClusterName' => 'Miscellaneous',
  'ClusterPriority' => '800',
  'Description' => 'Change the free fields for this ticket',
  'Link' => 'Action=AgentTicketFreeText;TicketID=[% Data.TicketID | html %]',
  'Module' => 'Kernel::Output::HTML::TicketMenu::Generic',
  'Name' => 'Free Fields',
  'PopupType' => 'TicketAction',
};
```

```
'Target' => ''
};
```

### **Ticket::Frontend::MenuModule###320-Link**

Shows a link in the menu that allows linking a ticket with another object in the ticket zoom view of the agent interface. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move\_into:group2". To cluster menu items use for Key "ClusterName" and for the Content any name you want to see in the UI. Use "ClusterPriority" to configure the order of a certain cluster within the toolbar.

默认值:

```
$Self->{'Ticket::Frontend::MenuModule'}->{'320-Link'} = {
  'Action' => 'AgentLinkObject',
  'ClusterName' => 'Miscellaneous',
  'ClusterPriority' => '800',
  'Description' => 'Link this ticket to other objects',
  'Link' => 'Action=AgentLinkObject;SourceObject=Ticket;SourceKey=[% Data.TicketID |
html %]',
  'Module' => 'Kernel::Output::HTML::TicketMenu::Generic',
  'Name' => 'Link',
  'PopupType' => 'TicketAction',
  'Target' => ''
};
```

### **Ticket::Frontend::MenuModule###400-Owner**

Shows a link in the menu to change the owner of a ticket in the ticket zoom view of the agent interface. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move\_into:group2". To cluster menu items use for Key "ClusterName" and for the Content any name you want to see in the UI. Use "ClusterPriority" to configure the order of a certain cluster within the toolbar.

默认值:

```
$Self->{'Ticket::Frontend::MenuModule'}->{'400-Owner'} = {
  'Action' => 'AgentTicketOwner',
  'ClusterName' => 'People',
  'ClusterPriority' => '430',
  'Description' => 'Change the owner for this ticket',
  'Link' => 'Action=AgentTicketOwner;TicketID=[% Data.TicketID | html %]',
  'Module' => 'Kernel::Output::HTML::TicketMenu::Generic',
  'Name' => 'Owner',
  'PopupType' => 'TicketAction',
  'Target' => ''
};
```

### **Ticket::Frontend::MenuModule###410-Responsible**

Shows a link in the menu to change the responsible agent of a ticket in the ticket zoom view of the agent interface. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move\_into:group2". To cluster menu items use for Key "ClusterName" and for the Content any name you want to see in the UI. Use "ClusterPriority" to configure the order of a certain cluster within the toolbar.

默认值:

```
$Self->{'Ticket::Frontend::MenuModule'}->{'410-Responsible'} = {
  'Action' => 'AgentTicketResponsible',
  'ClusterName' => 'People',
  'ClusterPriority' => '430',
  'Description' => 'Change the responsible for this ticket',
  'Link' => 'Action=AgentTicketResponsible;TicketID=[% Data.TicketID | html %]',
};
```

```
'Module' => 'Kernel::Output::HTML::TicketMenu::Responsible',
'Name' => 'Responsible',
'PopupType' => 'TicketAction',
'Target' => ''
};
```

### **Ticket::Frontend::MenuModule###420-Customer**

Shows a link in the menu to change the customer who requested the ticket in the ticket zoom view of the agent interface. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move\_into:group2". To cluster menu items use for Key "ClusterName" and for the Content any name you want to see in the UI. Use "ClusterPriority" to configure the order of a certain cluster within the toolbar.

默认值:

```
$Self->{'Ticket::Frontend::MenuModule'}->{'420-Customer'} = {
'Action' => 'AgentTicketCustomer',
'ClusterName' => 'People',
'ClusterPriority' => '430',
'Description' => 'Change the customer for this ticket',
'Link' => 'Action=AgentTicketCustomer;TicketID=[% Data.TicketID | html %]',
'Module' => 'Kernel::Output::HTML::TicketMenu::Generic',
'Name' => 'Customer',
'PopupType' => 'TicketAction',
'Target' => ''
};
```

### **Ticket::Frontend::MenuModule###420-Note**

Shows a link in the menu to add a note in the ticket zoom view of the agent interface. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move\_into:group2". To cluster menu items use for Key "ClusterName" and for the Content any name you want to see in the UI. Use "ClusterPriority" to configure the order of a certain cluster within the toolbar.

默认值:

```
$Self->{'Ticket::Frontend::MenuModule'}->{'420-Note'} = {
'Action' => 'AgentTicketNote',
'ClusterName' => 'Communication',
'ClusterPriority' => '435',
'Description' => 'Add a note to this ticket',
'Link' => 'Action=AgentTicketNote;TicketID=[% Data.TicketID | html %]',
'Module' => 'Kernel::Output::HTML::TicketMenu::Generic',
'Name' => 'Note',
'PopupType' => 'TicketAction',
'Target' => ''
};
```

### **Ticket::Frontend::MenuModule###425-Phone Call Outbound**

Shows a link in the menu to add a phone call outbound in the ticket zoom view of the agent interface. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move\_into:group2". To cluster menu items use for Key "ClusterName" and for the Content any name you want to see in the UI. Use "ClusterPriority" to configure the order of a certain cluster within the toolbar.

默认值:

```
$Self->{'Ticket::Frontend::MenuModule'}->{'425-Phone Call Outbound'} = {
'Action' => 'AgentTicketPhoneOutbound',
'ClusterName' => 'Communication',
'ClusterPriority' => '435',
'Description' => 'Add an outbound phone call to this ticket',
'Link' => 'Action=AgentTicketPhoneOutbound;TicketID=[% Data.TicketID | html %]',
```

```
'Module' => 'Kernel::Output::HTML::TicketMenu::Generic',
'Name' => 'Phone Call Outbound',
'PopupType' => 'TicketAction',
'Target' => ''
};
```

### **Ticket::Frontend::MenuModule###426-Phone Call Inbound**

Shows a link in the menu to add a phone call inbound in the ticket zoom view of the agent interface. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move\_into:group2". To cluster menu items use for Key "ClusterName" and for the Content any name you want to see in the UI. Use "ClusterPriority" to configure the order of a certain cluster within the toolbar.

默认值:

```
$Self->{'Ticket::Frontend::MenuModule'}->{'426-Phone Call Inbound'} = {
'Action' => 'AgentTicketPhoneInbound',
'ClusterName' => 'Communication',
'ClusterPriority' => '435',
'Description' => 'Add an inbound phone call to this ticket',
'Link' => 'Action=AgentTicketPhoneInbound;TicketID=[% Data.TicketID | html %]',
'Module' => 'Kernel::Output::HTML::TicketMenu::Generic',
'Name' => 'Phone Call Inbound',
'PopupType' => 'TicketAction',
'Target' => ''
};
```

### **Ticket::Frontend::MenuModule###427-Email Outbound**

Shows a link in the menu to send an outbound email in the ticket zoom view of the agent interface. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move\_into:group2". To cluster menu items use for Key "ClusterName" and for the Content any name you want to see in the UI. Use "ClusterPriority" to configure the order of a certain cluster within the toolbar.

默认值:

```
$Self->{'Ticket::Frontend::MenuModule'}->{'427-Email Outbound'} = {
'Action' => 'AgentTicketEmailOutbound',
'ClusterName' => 'Communication',
'ClusterPriority' => '435',
'Description' => 'Send new outgoing mail from this ticket',
'Link' => 'Action=AgentTicketEmailOutbound;TicketID=[% Data.TicketID | html %]',
'Module' => 'Kernel::Output::HTML::TicketMenu::Generic',
'Name' => 'E-Mail Outbound',
'PopupType' => 'TicketAction',
'Target' => ''
};
```

### **Ticket::Frontend::MenuModule###430-Merge**

Shows a link in the menu that allows merging tickets in the ticket zoom view of the agent interface. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move\_into:group2". To cluster menu items use for Key "ClusterName" and for the Content any name you want to see in the UI. Use "ClusterPriority" to configure the order of a certain cluster within the toolbar.

默认值:

```
$Self->{'Ticket::Frontend::MenuModule'}->{'430-Merge'} = {
'Action' => 'AgentTicketMerge',
'ClusterName' => 'Miscellaneous',
'ClusterPriority' => '800',
'Description' => 'Merge this ticket and all articles into another ticket',
'Link' => 'Action=AgentTicketMerge;TicketID=[% Data.TicketID | html %]',
'Module' => 'Kernel::Output::HTML::TicketMenu::Generic',
};
```

```
'Name' => 'Merge',
'PopupType' => 'TicketAction',
'Target' => ''
};
```

### **Ticket::Frontend::MenuModule###440-Pending**

Shows a link in the menu to set a ticket as pending in the ticket zoom view of the agent interface. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move\_into:group2". To cluster menu items use for Key "ClusterName" and for the Content any name you want to see in the UI. Use "ClusterPriority" to configure the order of a certain cluster within the toolbar.

默认值:

```
$Self->{'Ticket::Frontend::MenuModule'}->{'440-Pending'} = {
  'Action' => 'AgentTicketPending',
  'ClusterName' => '',
  'ClusterPriority' => '',
  'Description' => 'Set this ticket to pending',
  'Link' => 'Action=AgentTicketPending;TicketID=[% Data.TicketID | html %]',
  'Module' => 'Kernel::Output::HTML::TicketMenu::Generic',
  'Name' => 'Pending',
  'PopupType' => 'TicketAction',
  'Target' => ''
};
```

### **Ticket::Frontend::MenuModule###448-Watch**

Shows a link in the menu for subscribing / unsubscribing from a ticket in the ticket zoom view of the agent interface. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move\_into:group2". To cluster menu items use for Key "ClusterName" and for the Content any name you want to see in the UI. Use "ClusterPriority" to configure the order of a certain cluster within the toolbar.

默认值:

```
$Self->{'Ticket::Frontend::MenuModule'}->{'448-Watch'} = {
  'Action' => 'AgentTicketWatcher',
  'ClusterName' => '',
  'ClusterPriority' => '',
  'Description' => 'Watch this ticket',
  'Module' => 'Kernel::Output::HTML::TicketMenu::TicketWatcher',
  'Name' => 'Watch',
  'Target' => ''
};
```

### **Ticket::Frontend::MenuModule###450-Close**

Shows a link in the menu to close a ticket in the ticket zoom view of the agent interface. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move\_into:group2". To cluster menu items use for Key "ClusterName" and for the Content any name you want to see in the UI. Use "ClusterPriority" to configure the order of a certain cluster within the toolbar.

默认值:

```
$Self->{'Ticket::Frontend::MenuModule'}->{'450-Close'} = {
  'Action' => 'AgentTicketClose',
  'ClusterName' => '',
  'ClusterPriority' => '',
  'Description' => 'Close this ticket',
  'Link' => 'Action=AgentTicketClose;TicketID=[% Data.TicketID | html %]',
  'Module' => 'Kernel::Output::HTML::TicketMenu::Generic',
  'Name' => 'Close',
  'PopupType' => 'TicketAction',
};
```

```
'Target' => ''
};
```

### **Ticket::Frontend::MenuModule###460-Delete**

Shows a link in the menu to delete a ticket in the ticket zoom view of the agent interface. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move\_into:group2". To cluster menu items use for Key "ClusterName" and for the Content any name you want to see in the UI. Use "ClusterPriority" to configure the order of a certain cluster within the toolbar.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::MenuModule'}->{'460-Delete'} = {
  'Action' => 'AgentTicketMove',
  'ClusterName' => '',
  'ClusterPriority' => '',
  'Description' => 'Delete this ticket',
  'Link' => 'Action=AgentTicketMove;TicketID=[% Data.TicketID %];DestQueue=Delete',
  'Module' => 'Kernel::Output::HTML::TicketMenu::Generic',
  'Name' => 'Delete',
  'PopupType' => '',
  'Target' => ''
};
```

### **Ticket::Frontend::MenuModule###470-Junk**

Shows a link to set a ticket as junk in the ticket zoom view of the agent interface. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move\_into:group2". To cluster menu items use for Key "ClusterName" and for the Content any name you want to see in the UI. Use "ClusterPriority" to configure the order of a certain cluster within the toolbar.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::MenuModule'}->{'470-Junk'} = {
  'Action' => 'AgentTicketMove',
  'ClusterName' => '',
  'ClusterPriority' => '',
  'Description' => 'Mark this ticket as junk!',
  'Link' => 'Action=AgentTicketMove;TicketID=[% Data.TicketID %];DestQueue=Junk',
  'Module' => 'Kernel::Output::HTML::TicketMenu::Generic',
  'Name' => 'Spam',
  'PopupType' => '',
  'Target' => ''
};
```

## **Ticket → Frontend::Agent::Ticket::MenuModulePre**

### **Ticket::Frontend::PreMenuModule###100-Lock**

Shows a link in the menu to lock / unlock a ticket in the ticket overviews of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::PreMenuModule'}->{'100-Lock'} = {
  'Action' => 'AgentTicketLock',
  'Module' => 'Kernel::Output::HTML::TicketMenu::Lock',
  'Name' => 'Lock',
  'PopupType' => '',
  'Target' => ''
};
```



```
};
```

### **Ticket::Frontend::PreMenuModule###200-Zoom**

Shows a link in the menu to zoom a ticket in the ticket overviews of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::PreMenuModule'}->{'200-Zoom'} = {  
  'Action' => 'AgentTicketZoom',  
  'Description' => 'Look into a ticket!',  
  'Link' => 'Action=AgentTicketZoom;TicketID=[% Data.TicketID | html %]',  
  'Module' => 'Kernel::Output::HTML::TicketMenu::Generic',  
  'Name' => 'Zoom',  
  'PopupType' => '',  
  'Target' => ''  
};
```

### **Ticket::Frontend::PreMenuModule###210-History**

Shows a link in the menu to see the history of a ticket in every ticket overview of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::PreMenuModule'}->{'210-History'} = {  
  'Action' => 'AgentTicketHistory',  
  'Description' => 'Show the ticket history',  
  'Link' => 'Action=AgentTicketHistory;TicketID=[% Data.TicketID | html %]',  
  'Module' => 'Kernel::Output::HTML::TicketMenu::Generic',  
  'Name' => 'History',  
  'PopupType' => 'TicketHistory',  
  'Target' => ''  
};
```

### **Ticket::Frontend::PreMenuModule###300-Priority**

Shows a link in the menu to set the priority of a ticket in every ticket overview of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::PreMenuModule'}->{'300-Priority'} = {  
  'Action' => 'AgentTicketPriority',  
  'Description' => 'Change the priority for this ticket',  
  'Link' => 'Action=AgentTicketPriority;TicketID=[% Data.TicketID | html %]',  
  'Module' => 'Kernel::Output::HTML::TicketMenu::Generic',  
  'Name' => 'Priority',  
  'PopupType' => 'TicketAction',  
  'Target' => ''  
};
```

### **Ticket::Frontend::PreMenuModule###420-Note**

Shows a link in the menu to add a note to a ticket in every ticket overview of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::PreMenuModule'}->{'420-Note'} = {  
  'Action' => 'AgentTicketNote',  
  'Description' => 'Add a note to this ticket',  
  'Link' => 'Action=AgentTicketNote;TicketID=[% Data.TicketID | html %]',  
  'Module' => 'Kernel::Output::HTML::TicketMenu::Generic',  
  'Name' => 'Note',  
  'PopupType' => 'TicketAction',  
  'Target' => ''  
};
```

### **Ticket::Frontend::PreMenuModule###440-Close**

Shows a link in the menu to close a ticket in every ticket overview of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::PreMenuModule'}->{'440-Close'} = {  
  'Action' => 'AgentTicketClose',  
  'Description' => 'Close this ticket',  
  'Link' => 'Action=AgentTicketClose;TicketID=[% Data.TicketID | html %]',  
  'Module' => 'Kernel::Output::HTML::TicketMenu::Generic',  
  'Name' => 'Close',  
  'PopupType' => 'TicketAction',  
  'Target' => ''  
};
```

### **Ticket::Frontend::PreMenuModule###445-Move**

Shows a link in the menu to move a ticket in every ticket overview of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::PreMenuModule'}->{'445-Move'} = {  
  'Action' => 'AgentTicketMove',  
  'Description' => 'Change queue!',  
  'Module' => 'Kernel::Output::HTML::TicketMenu::Move',  
  'Name' => 'Move'  
};
```

### **Ticket::Frontend::PreMenuModule###450-Delete**

Shows a link in the menu to delete a ticket in every ticket overview of the agent interface. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move\_into:group2".

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::PreMenuModule'}->{'450-Delete'} = {  
  'Action' => 'AgentTicketMove',  
  'Description' => 'Delete this ticket',  
  'Link' => 'Action=AgentTicketMove;TicketID=[% Data.TicketID %];DestQueue=Delete',  
  'Module' => 'Kernel::Output::HTML::TicketMenu::Generic',  
  'Name' => 'Delete',  
  'PopupType' => '',  
  'Target' => ''  
};
```

### **Ticket::Frontend::PreMenuModule###460-Junk**

Shows a link in the menu to set a ticket as junk in every ticket overview of the agent interface. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move\_into:group2".

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::PreMenuModule'}->{'460-Junk'} = {  
  'Action' => 'AgentTicketMove',  
  'Description' => 'Mark as Spam!',  
  'Link' => 'Action=AgentTicketMove;TicketID=[% Data.TicketID %];DestQueue=Junk',  
  'Module' => 'Kernel::Output::HTML::TicketMenu::Generic',  
  'Name' => 'Spam',  
  'PopupType' => '',  
};
```

```
'Target' => ''
};
```

## Ticket → Frontend::Agent::Ticket::OverviewMenuModule

### Ticket::Frontend::OverviewMenuModule###001-Sort

Shows a select of ticket attributes to order the queue view ticket list. The possible selections can be configured via 'TicketOverviewMenuSort###SortAttributes'.

默认值:

```
$Self->{'Ticket::Frontend::OverviewMenuModule'}->{'001-Sort'} = {
  'Module' => 'Kernel::Output::HTML::TicketOverviewMenu::Sort'
};
```

### TicketOverviewMenuSort###SortAttributes

Defines from which ticket attributes the agent can select the result order.

默认值:

```
$Self->{'TicketOverviewMenuSort'}->{'SortAttributes'} = {
  'Age' => '1',
  'Title' => '1'
};
```

## Ticket → Frontend::Agent::Ticket::ViewBounce

### Ticket::Frontend::AgentTicketBounce###Permission

Required permissions to use the ticket bounce screen in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketBounce'}->{'Permission'} = 'bounce';
```

### Ticket::Frontend::AgentTicketBounce###RequiredLock

Defines if a ticket lock is required in the ticket bounce screen of the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketBounce'}->{'RequiredLock'} = '1';
```

### Ticket::Frontend::AgentTicketBounce###StateDefault

Defines the default next state of a ticket after being bounced, in the ticket bounce screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketBounce'}->{'StateDefault'} = 'closed
successful';
```

### Ticket::Frontend::AgentTicketBounce###StateType

Defines the next state of a ticket after being bounced, in the ticket bounce screen of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketBounce'}->{'StateType'} = [  
    'open',  
    'closed'  
];
```

### **Ticket::Frontend::BounceText**

Defines the default ticket bounced notification for customer/sender in the ticket bounce screen of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::BounceText'} = 'Your email with ticket number  
"<OTRS_TICKET>" is bounced to "<OTRS_BOUNCE_TO>". Contact this address for further  
information.';
```

### **Ticket → Frontend::Agent::Ticket::ViewBulk**

#### **Ticket::Frontend::AgentTicketBulk###RequiredLock**

Automatically lock and set owner to current Agent after selecting for an Bulk Action.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'RequiredLock'} = '1';
```

#### **Ticket::Frontend::AgentTicketBulk###TicketType**

Sets the ticket type in the ticket bulk screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'TicketType'} = '1';
```

#### **Ticket::Frontend::AgentTicketBulk###Owner**

Sets the ticket owner in the ticket bulk screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'Owner'} = '1';
```

#### **Ticket::Frontend::AgentTicketBulk###Responsible**

Sets the responsible agent of the ticket in the ticket bulk screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'Responsible'} = '1';
```

#### **Ticket::Frontend::AgentTicketBulk###State**

Sets the state of a ticket in the ticket bulk screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'State'} = '1';
```

#### **Ticket::Frontend::AgentTicketBulk###StateType**

Defines the next state of a ticket after adding a note, in the ticket bulk screen of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'StateType'} = [
  'open',
  'closed',
  'pending reminder',
  'pending auto'
];
```

### **Ticket::Frontend::AgentTicketBulk###StateDefault**

Defines the default next state of a ticket after adding a note, in the ticket bulk screen of the agent interface.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'StateDefault'} = 'open';
```

### **Ticket::Frontend::AgentTicketBulk###Priority**

Shows the ticket priority options in the ticket bulk screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'Priority'} = '1';
```

### **Ticket::Frontend::AgentTicketBulk###PriorityDefault**

Defines the default ticket priority in the ticket bulk screen of the agent interface.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'PriorityDefault'} = '3 normal';
```

### **Ticket::Frontend::AgentTicketBulk###ArticleTypeDefault**

Defines the default type of the note in the ticket bulk screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'ArticleTypeDefault'} = 'note-internal';
```

### **Ticket::Frontend::AgentTicketBulk###ArticleTypes**

Specifies the different note types that will be used in the system.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'ArticleTypes'} = {
  'note-external' => '1',
  'note-internal' => '1',
  'note-report' => '0'
};
```

## **Ticket → Frontend::Agent::Ticket::ViewClose**

### **Ticket::Frontend::AgentTicketClose###Permission**

Required permissions to use the close ticket screen in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketClose'}->{'Permission'} = 'close';
```

### **Ticket::Frontend::AgentTicketClose###RequiredLock**

Defines if a ticket lock is required in the close ticket screen of the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketClose'}->{'RequiredLock'} = '1';
```

### **Ticket::Frontend::AgentTicketClose###TicketType**

Sets the ticket type in the close ticket screen of the agent interface (Ticket::Type needs to be activated).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketClose'}->{'TicketType'} = '0';
```

### **Ticket::Frontend::AgentTicketClose###Service**

Sets the service in the close ticket screen of the agent interface (Ticket::Service needs to be activated).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketClose'}->{'Service'} = '0';
```

### **Ticket::Frontend::AgentTicketClose###ServiceMandatory**

Sets if service must be selected by the agent.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketClose'}->{'ServiceMandatory'} = '0';
```

### **Ticket::Frontend::AgentTicketClose###SLAMandatory**

Sets if SLA must be selected by the agent.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketClose'}->{'SLAMandatory'} = '0';
```

### **Ticket::Frontend::AgentTicketClose###Queue**

Sets the queue in the ticket close screen of a zoomed ticket in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketClose'}->{'Queue'} = '0';
```

### **Ticket::Frontend::AgentTicketClose###Owner**

Sets the ticket owner in the close ticket screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketClose'}->{'Owner'} = '0';
```

### **Ticket::Frontend::AgentTicketClose###OwnerMandatory**

Sets if ticket owner must be selected by the agent.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketClose'}->{'OwnerMandatory'} = '0';
```

### **Ticket::Frontend::AgentTicketClose###Responsible**

Sets the responsible agent of the ticket in the close ticket screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketClose'}->{'Responsible'} = '0';
```

### **Ticket::Frontend::AgentTicketClose###State**

Sets the state of a ticket in the close ticket screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketClose'}->{'State'} = '1';
```

### **Ticket::Frontend::AgentTicketClose###StateType**

Defines the next state of a ticket after adding a note, in the close ticket screen of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketClose'}->{'StateType'} = [  
  'closed'  
];
```

### **Ticket::Frontend::AgentTicketClose###StateDefault**

Defines the default next state of a ticket after adding a note, in the close ticket screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketClose'}->{'StateDefault'} = 'closed successful';
```

### **Ticket::Frontend::AgentTicketClose###Note**

Allows adding notes in the close ticket screen of the agent interface. Can be overwritten by Ticket::Frontend::NeedAccountedTime.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketClose'}->{'Note'} = '1';
```

### **Ticket::Frontend::AgentTicketClose###NoteMandatory**

Sets if note must be filled in by the agent. Can be overwritten by Ticket::Frontend::NeedAccountedTime.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketClose'}->{'NoteMandatory'} = '1';
```

### **Ticket::Frontend::AgentTicketClose###Subject**

Sets the default subject for notes added in the close ticket screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketClose'}->{'Subject'} = '';
```

### **Ticket::Frontend::AgentTicketClose###Body**

Sets the default body text for notes added in the close ticket screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketClose'}->{'Body'} = '';
```

### **Ticket::Frontend::AgentTicketClose###InvolvedAgent**

Shows a list of all the involved agents on this ticket, in the close ticket screen of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketClose'}->{'InvolvedAgent'} = '0';
```

### **Ticket::Frontend::AgentTicketClose###InformAgent**

Shows a list of all the possible agents (all agents with note permissions on the queue/ticket) to determine who should be informed about this note, in the close ticket screen of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketClose'}->{'InformAgent'} = '0';
```

### **Ticket::Frontend::AgentTicketClose###ArticleTypeDefault**

Defines the default type of the note in the close ticket screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketClose'}->{'ArticleTypeDefault'} = 'note-internal';
```

### **Ticket::Frontend::AgentTicketClose###ArticleTypes**

Specifies the available note types for this ticket mask. If the option is deselected, ArticleTypeDefault is used and the option is removed from the mask.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketClose'}->{'ArticleTypes'} = {  
  'note-external' => '0',  
  'note-internal' => '1',  
  'note-report' => '0'  
};
```

### **Ticket::Frontend::AgentTicketClose###Priority**

Shows the ticket priority options in the close ticket screen of the agent interface.



不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketClose'}->{'Priority'} = '0';
```

### **Ticket::Frontend::AgentTicketClose###PriorityDefault**

Defines the default ticket priority in the close ticket screen of the agent interface.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketClose'}->{'PriorityDefault'} = '3 normal';
```

### **Ticket::Frontend::AgentTicketClose###Title**

Shows the title fields in the close ticket screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketClose'}->{'Title'} = '0';
```

### **Ticket::Frontend::AgentTicketClose###HistoryType**

Defines the history type for the close ticket screen action, which gets used for ticket history in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketClose'}->{'HistoryType'} = 'AddNote';
```

### **Ticket::Frontend::AgentTicketClose###HistoryComment**

Defines the history comment for the close ticket screen action, which gets used for ticket history in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketClose'}->{'HistoryComment'} = '%Close';
```

### **Ticket::Frontend::AgentTicketClose###DynamicField**

Dynamic fields shown in the ticket close screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketClose'}->{'DynamicField'} = {};
```

### **Ticket::Frontend::AgentTicketClose###RichTextWidth**

Defines the width for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketClose'}->{'RichTextWidth'} = '620';
```

### **Ticket::Frontend::AgentTicketClose###RichTextHeight**

Defines the height for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketClose'}->{'RichTextHeight'} = '100';
```

## **Ticket → Frontend::Agent::Ticket::ViewCompose**

### **Ticket::Frontend::AgentTicketCompose###Permission**

Required permissions to use the ticket compose screen in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketCompose'}->{'Permission'} = 'compose';
```

### **Ticket::Frontend::AgentTicketCompose###RequiredLock**

Defines if a ticket lock is required in the ticket compose screen of the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketCompose'}->{'RequiredLock'} = '1';
```

### **Ticket::Frontend::AgentTicketCompose###StateDefault**

Defines the default next state of a ticket if it is composed / answered in the ticket compose screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketCompose'}->{'StateDefault'} = 'open';
```

### **Ticket::Frontend::AgentTicketCompose###StateType**

Defines the next possible states after composing / answering a ticket in the ticket compose screen of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketCompose'}->{'StateType'} = [  
  'open',  
  'closed',  
  'pending auto',  
  'pending reminder'  
];
```

### **Ticket::Frontend::AgentTicketCompose###ArticleTypes**

Specifies the different article types that will be used in the system.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketCompose'}->{'ArticleTypes'} = [  
  'email-external',  
  'email-internal'  
];
```

### **Ticket::Frontend::AgentTicketCompose###DefaultArticleType**

Specifies the default article type for the ticket compose screen in the agent interface if the article type cannot be automatically detected.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketCompose'}->{'DefaultArticleType'} = 'email-external';
```

### **Ticket::Frontend::ResponseFormat**

Defines the format of responses in the ticket compose screen of the agent interface ([% Data.OrigFrom | html %] is From 1:1, [% Data.OrigFromName | html %] is only realname of From).

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::ResponseFormat'} = "[% Data.Salutation | html %]  
[% Data.StdResponse | html %]  
[% Data.Signature | html %]  
  
[% Data.Created | Localize("TimeShort") %] - [% Data.OrigFromName | html %] [%  
Translate("wrote") | html %]:  
[% Data.Body | html %]  
';
```

### **Ticket::Frontend::Quote**

Defines the used character for plaintext email quotes in the ticket compose screen of the agent interface. If this is empty or inactive, original emails will not be quoted but appended to the response.

默认值:

```
$Self->{'Ticket::Frontend::Quote'} = '>';
```

### **Ticket::Frontend::ResponseQuoteMaxLines**

Defines the maximum number of quoted lines to be added to responses.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::ResponseQuoteMaxLines'} = '99';
```

### **Ticket::Frontend::ComposeAddCustomerAddress**

Adds customers email addresses to recipients in the ticket compose screen of the agent interface. The customers email address won't be added if the article type is email-internal.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::ComposeAddCustomerAddress'} = '1';
```

### **Ticket::Frontend::ComposeReplaceSenderAddress**

Replaces the original sender with current customer's email address on compose answer in the ticket compose screen of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::ComposeReplaceSenderAddress'} = '0';
```

### **Ticket::Frontend::ComposeExcludeCcRecipients**

Uses Cc recipients in reply Cc list on compose an email answer in the ticket compose screen of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::ComposeExcludeCcRecipients'} = '0';
```

### **Ticket::Frontend::AgentTicketCompose###DynamicField**

Dynamic fields shown in the ticket compose screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketCompose'}->{'DynamicField'} = {};
```

## **Ticket → Frontend::Agent::Ticket::ViewCustomer**

### **Ticket::Frontend::AgentTicketCustomer###Permission**

Required permissions to change the customer of a ticket in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketCustomer'}->{'Permission'} = 'customer';
```

### **Ticket::Frontend::AgentTicketCustomer###RequiredLock**

Defines if a ticket lock is required to change the customer of a ticket in the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketCustomer'}->{'RequiredLock'} = '0';
```

### **Ticket::Frontend::AgentTicketCustomer::CustomerIDReadOnly**

Controls if CustomerID is read-only in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketCustomer::CustomerIDReadOnly'} = '1';
```

## **Ticket → Frontend::Agent::Ticket::ViewEmailNew**

### **Ticket::Frontend::AgentTicketEmail###Priority**

Sets the default priority for new email tickets in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'Priority'} = '3 normal';
```

### **Ticket::Frontend::AgentTicketEmail###ArticleType**

Sets the default article type for new email tickets in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'ArticleType'} = 'email-external';
```

### **Ticket::Frontend::AgentTicketEmail###SenderType**

Sets the default sender type for new email tickets in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'SenderType'} = 'agent';
```

### **Ticket::Frontend::AgentTicketEmail::CustomerIDReadOnly**

Controls if CustomerID is read-only in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketEmail::CustomerIDReadOnly'} = '1';
```

### **Ticket::Frontend::AgentTicketEmail###Subject**

Sets the default subject for new email tickets (e.g. 'email Outbound') in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'Subject'} = '';
```

### **Ticket::Frontend::AgentTicketEmail###Body**

Sets the default text for new email tickets in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'Body'} = '';
```

### **Ticket::Frontend::AgentTicketEmail###StateDefault**

Sets the default next ticket state, after the creation of an email ticket in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'StateDefault'} = 'open';
```

### **Ticket::Frontend::AgentTicketEmail###StateType**

Determines the next possible ticket states, after the creation of a new email ticket in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'StateType'} = [  
  'open',  
  'pending auto',  
  'pending reminder',  
  'closed'  
];
```

### **Ticket::Frontend::AgentTicketEmail###HistoryType**

Defines the history type for the email ticket screen action, which gets used for ticket history in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'HistoryType'} = 'EmailAgent';
```

### **Ticket::Frontend::AgentTicketEmail###HistoryComment**

Defines the history comment for the email ticket screen action, which gets used for ticket history in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'HistoryComment'} = '';
```

### **Ticket::Frontend::AgentTicketEmail###ServiceMandatory**

Sets if service must be selected by the agent.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'ServiceMandatory'} = '0';
```

### **Ticket::Frontend::AgentTicketEmail###SLAMandatory**

Sets if SLA must be selected by the agent.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'SLAMandatory'} = '0';
```

### **Ticket::Frontend::AgentTicketEmail###DynamicField**

Dynamic fields shown in the ticket email screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'DynamicField'} = {};
```

### **Ticket::Frontend::AgentTicketEmail###RichTextWidth**

Defines the width for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'RichTextWidth'} = '620';
```

### **Ticket::Frontend::AgentTicketEmail###RichTextHeight**

Defines the height for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'RichTextHeight'} = '320';
```

## **Ticket → Frontend::Agent::Ticket::ViewEmailOutbound**

### **Ticket::Frontend::AgentTicketEmailOutbound###Permission**

Required permissions to use the email outbound screen in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketEmailOutbound'}->{'Permission'} = 'compose';
```

### **Ticket::Frontend::AgentTicketEmailOutbound###RequiredLock**

Defines if a ticket lock is required in the email outbound screen of the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketEmailOutbound'}->{'RequiredLock'} = '1';
```

### **Ticket::Frontend::AgentTicketEmailOutbound###StateDefault**

Defines the default next state of a ticket after the message has been sent, in the email outbound screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketEmailOutbound'}->{'StateDefault'} = 'open';
```

### **Ticket::Frontend::AgentTicketEmailOutbound###StateType**

Defines the next possible states after sending a message in the email outbound screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketEmailOutbound'}->{'StateType'} = [
  'open',
  'closed',
  'pending reminder',
  'pending auto'
];
```

### **Ticket::Frontend::AgentTicketEmailOutbound###ArticleTypeDefault**

Defines the default type of the message in the email outbound screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketEmailOutbound'}->{'ArticleTypeDefault'} =  
'email-internal';
```

### **Ticket::Frontend::AgentTicketEmailOutbound###ArticleTypes**

Specifies the different article types that will be used in the system.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketEmailOutbound'}->{'ArticleTypes'} = [  
  'email-external',  
  'email-internal'  
];
```

### **Ticket::Frontend::AgentTicketEmailOutbound###DynamicField**

Dynamic fields shown in the email outbound screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketEmailOutbound'}->{'DynamicField'} = {};
```

### **Ticket::Frontend::AgentTicketEmailOutbound###RichTextWidth**

Defines the width for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketEmailOutbound'}->{'RichTextWidth'} = '620';
```

### **Ticket::Frontend::AgentTicketEmailOutbound###RichTextHeight**

Defines the height for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketEmailOutbound'}->{'RichTextHeight'} = '300';
```

## **Ticket → Frontend::Agent::Ticket::ViewEscalation**

### **Ticket::Frontend::AgentTicketEscalationView###TicketPermission**

Defines the required permission to show a ticket in the escalation view of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketEscalationView'}->{'TicketPermission'} = 'rw';
```

### **Ticket::Frontend::AgentTicketEscalationView###ViewableTicketsPage**

Shows all open tickets (even if they are locked) in the escalation view of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketEscalationView'}->{'ViewableTicketsPage'} =  
'50';
```



### **Ticket::Frontend::AgentTicketEscalationView###SortBy::Default**

Defines the default ticket attribute for ticket sorting in the escalation view of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketEscalationView'}->{'SortBy::Default'} = 'EscalationTime';
```

### **Ticket::Frontend::AgentTicketEscalationView###Order::Default**

Defines the default ticket order (after priority sort) in the escalation view of the agent interface. Up: oldest on top. Down: latest on top.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketEscalationView'}->{'Order::Default'} = 'Up';
```

### **Ticket::Frontend::AgentTicketEscalationView###DefaultColumns**

Columns that can be filtered in the escalation view of the agent interface. Possible settings: 0 = Disabled, 1 = Available, 2 = Enabled by default. Note: Only Ticket attributes, Dynamic Fields (DynamicField\_NameX) and Customer attributes (e.g. CustomerUserPhone, CustomerCompanyName, ...) are allowed.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketEscalationView'}->{'DefaultColumns'} = {  
  'Age' => '2',  
  'Changed' => '1',  
  'Created' => '1',  
  'CustomerCompanyName' => '1',  
  'CustomerID' => '2',  
  'CustomerName' => '1',  
  'CustomerUserID' => '1',  
  'EscalationResponseTime' => '1',  
  'EscalationSolutionTime' => '1',  
  'EscalationTime' => '2',  
  'EscalationUpdateTime' => '1',  
  'Lock' => '2',  
  'Owner' => '2',  
  'PendingTime' => '1',  
  'Priority' => '1',  
  'Queue' => '2',  
  'Responsible' => '1',  
  'SLA' => '1',  
  'Service' => '1',  
  'State' => '2',  
  'TicketNumber' => '2',  
  'Title' => '2',  
  'Type' => '1'  
};
```

## **Ticket → Frontend::Agent::Ticket::ViewForward**

### **Ticket::Frontend::AgentTicketForward###Permission**

Required permissions to use the ticket forward screen in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketForward'}->{'Permission'} = 'forward';
```

### **Ticket::Frontend::AgentTicketForward###RequiredLock**

Defines if a ticket lock is required in the ticket forward screen of the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketForward'}->{'RequiredLock'} = '1';
```

### **Ticket::Frontend::AgentTicketForward###StateDefault**

Defines the default next state of a ticket after being forwarded, in the ticket forward screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketForward'}->{'StateDefault'} = 'closed  
successful';
```

### **Ticket::Frontend::AgentTicketForward###StateType**

Defines the next possible states after forwarding a ticket in the ticket forward screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketForward'}->{'StateType'} = [  
  'open',  
  'closed',  
  'pending reminder',  
  'pending auto'  
];
```

### **Ticket::Frontend::AgentTicketForward###ArticleTypeDefault**

Defines the default type of forwarded message in the ticket forward screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketForward'}->{'ArticleTypeDefault'} = 'email-  
external';
```

### **Ticket::Frontend::AgentTicketForward###ArticleTypes**

Specifies the different article types that will be used in the system.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketForward'}->{'ArticleTypes'} = [  
  'email-external',  
  'email-internal'  
];
```

### **Ticket::Frontend::AgentTicketForward###DynamicField**

Dynamic fields shown in the ticket forward screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketForward'}->{'DynamicField'} = {};
```

### **Ticket::Frontend::AgentTicketForward###RichTextWidth**

Defines the width for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketForward'}->{'RichTextWidth'} = '620';
```

### **Ticket::Frontend::AgentTicketForward###RichTextHeight**

Defines the height for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketForward'}->{'RichTextHeight'} = '100';
```

## **Ticket → Frontend::Agent::Ticket::ViewFreeText**

### **Ticket::Frontend::AgentTicketFreeText###Permission**

Required permissions to use the ticket free text screen in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'Permission'} = 'rw';
```

### **Ticket::Frontend::AgentTicketFreeText###RequiredLock**

Defines if a ticket lock is required in the ticket free text screen of the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'RequiredLock'} = '0';
```

### **Ticket::Frontend::AgentTicketFreeText###TicketType**

Sets the ticket type in the ticket free text screen of the agent interface (Ticket::Type needs to be activated).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'TicketType'} = '1';
```

### **Ticket::Frontend::AgentTicketFreeText###Service**

Sets the service in the ticket free text screen of the agent interface (Ticket::Service needs to be activated).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'Service'} = '1';
```

### **Ticket::Frontend::AgentTicketFreeText###ServiceMandatory**

Sets if service must be selected by the agent.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'ServiceMandatory'} = '0';
```

### **Ticket::Frontend::AgentTicketFreeText###SLAMandatory**

Sets if SLA must be selected by the agent.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'SLAMandatory'} = '0';
```

### **Ticket::Frontend::AgentTicketFreeText###Queue**

Sets the queue in the ticket free text screen of a zoomed ticket in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'Queue'} = '0';
```

### **Ticket::Frontend::AgentTicketFreeText###Owner**

Sets the ticket owner in the ticket free text screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'Owner'} = '0';
```

### **Ticket::Frontend::AgentTicketFreeText###OwnerMandatory**

Sets if ticket owner must be selected by the agent.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'OwnerMandatory'} = '0';
```

### **Ticket::Frontend::AgentTicketFreeText###Responsible**

Sets the responsible agent of the ticket in the ticket free text screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'Responsible'} = '0';
```

### **Ticket::Frontend::AgentTicketFreeText###State**

Sets the state of a ticket in the ticket free text screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'State'} = '0';
```

### **Ticket::Frontend::AgentTicketFreeText###StateType**

Defines the next state of a ticket after adding a note, in the ticket free text screen of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'StateType'} = [
    'open',
    'closed',
    'pending reminder',
    'pending auto'
];
```

### **Ticket::Frontend::AgentTicketFreeText###StateDefault**

Defines the default next state of a ticket after adding a note, in the ticket free text screen of the agent interface.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'StateDefault'} = 'open';
```

### **Ticket::Frontend::AgentTicketFreeText###Note**

Allows adding notes in the ticket free text screen of the agent interface. Can be overwritten by Ticket::Frontend::NeedAccountedTime.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'Note'} = '0';
```

### **Ticket::Frontend::AgentTicketFreeText###NoteMandatory**

Sets if note must be filled in by the agent. Can be overwritten by Ticket::Frontend::NeedAccountedTime.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'NoteMandatory'} = '0';
```

### **Ticket::Frontend::AgentTicketFreeText###Subject**

Defines the default subject of a note in the ticket free text screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'Subject'} = '';
```

### **Ticket::Frontend::AgentTicketFreeText###Body**

Defines the default body of a note in the ticket free text screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'Body'} = '';
```

### **Ticket::Frontend::AgentTicketFreeText###InvolvedAgent**

Shows a list of all the involved agents on this ticket, in the ticket free text screen of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'InvolvedAgent'} = '0';
```

### **Ticket::Frontend::AgentTicketFreeText###InformAgent**

Shows a list of all the possible agents (all agents with note permissions on the queue/ticket) to determine who should be informed about this note, in the ticket free text screen of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'InformAgent'} = '0';
```

### **Ticket::Frontend::AgentTicketFreeText###ArticleTypeDefault**

Defines the default type of the note in the ticket free text screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'ArticleTypeDefault'} = 'note-internal';
```

### **Ticket::Frontend::AgentTicketFreeText###ArticleTypes**

Specifies the available note types for this ticket mask. If the option is deselected, ArticleTypeDefault is used and the option is removed from the mask.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'ArticleTypes'} = {  
  'note-external' => '1',  
  'note-internal' => '1',  
  'note-report' => '0'  
};
```

### **Ticket::Frontend::AgentTicketFreeText###Priority**

Shows the ticket priority options in the ticket free text screen of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'Priority'} = '0';
```

### **Ticket::Frontend::AgentTicketFreeText###PriorityDefault**

Defines the default ticket priority in the ticket free text screen of the agent interface.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'PriorityDefault'} = '3 normal';
```

### **Ticket::Frontend::AgentTicketFreeText###Title**

Shows the title field in the ticket free text screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'Title'} = '1';
```

### **Ticket::Frontend::AgentTicketFreeText###HistoryType**

Defines the history type for the ticket free text screen action, which gets used for ticket history.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'HistoryType'} = 'AddNote';
```

### **Ticket::Frontend::AgentTicketFreeText###HistoryComment**

Defines the history comment for the ticket free text screen action, which gets used for ticket history.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'HistoryComment'} = '%FreeText';
```

### **Ticket::Frontend::AgentTicketFreeText###DynamicField**

Dynamic fields shown in the ticket free text screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'DynamicField'} = {};
```

### **Ticket::Frontend::AgentTicketFreeText###RichTextWidth**

Defines the width for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'RichTextWidth'} = '620';
```

### **Ticket::Frontend::AgentTicketFreeText###RichTextHeight**

Defines the height for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'RichTextHeight'} = '100';
```

## **Ticket → Frontend::Agent::Ticket::ViewHistory**

### **Ticket::Frontend::HistoryOrder**

Shows the ticket history (reverse ordered) in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::HistoryOrder'} = 'normal';
```

### **Ticket::Frontend::HistoryTypes###000-Framework**

Controls how to display the ticket history entries as readable values.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::HistoryTypes'}->{'000-Framework'} = {
  'AddNote' => 'Added note (%s)',
  'ArchiveFlagUpdate' => 'Archive state changed: "%s"',
  'Bounce' => 'Bounced to "%s".',
  'CustomerUpdate' => 'Updated: %s',
  'EmailAgent' => 'Email sent to customer.',
  'EmailCustomer' => 'Added email. %s',
  'EscalationResponseTimeNotifyBefore' => 'Escalation response time forewarned',
  'EscalationResponseTimeStart' => 'Escalation response time in effect',
  'EscalationResponseTimeStop' => 'Escalation response time finished',
  'EscalationSolutionTimeNotifyBefore' => 'Escalation solution time forewarned',
  'EscalationSolutionTimeStart' => 'Escalation solution time in effect',
  'EscalationSolutionTimeStop' => 'Escalation solution time finished',
  'EscalationUpdateTimeNotifyBefore' => 'Escalation update time forewarned',
  'EscalationUpdateTimeStart' => 'Escalation update time in effect',
  'EscalationUpdateTimeStop' => 'Escalation update time finished',
}
```

```
'FollowUp' => 'FollowUp for [%s]. %s',
'Forward' => 'Forwarded to "%s".',
'Lock' => 'Locked ticket.',
'LoopProtection' => 'Loop-Protection! No auto-response sent to "%s".',
'Misc' => '%s',
'Move' => 'Ticket moved into Queue "%s" (%s) from Queue "%s" (%s).',
'NewTicket' => 'New Ticket [%s] created (Q=%s;P=%s;S=%s).',
'OwnerUpdate' => 'New owner is "%s" (ID=%s).',
'PhoneCallAgent' => 'Agent called customer.',
'PhoneCallCustomer' => 'Customer called us.',
'PriorityUpdate' => 'Changed priority from "%s" (%s) to "%s" (%s).',
'Remove' => '%s',
'ResponsibleUpdate' => 'New responsible is "%s" (ID=%s).',
'SLAUpdate' => 'Updated SLA to %s (ID=%s).',
'SendAgentNotification' => '"%s" notification was sent to "%s" by "%s".',
'SendAnswer' => 'Email sent to "%s".',
'SendAutoFollowUp' => 'AutoFollowUp sent to "%s".',
'SendAutoReject' => 'AutoReject sent to "%s".',
'SendAutoReply' => 'AutoReply sent to "%s".',
'SendCustomerNotification' => 'Notification sent to "%s".',
'ServiceUpdate' => 'Updated Service to %s (ID=%s).',
'SetPendingTime' => 'Updated: %s',
'StateUpdate' => 'Old: "%s" New: "%s"',
'Subscribe' => 'Added subscription for user "%s".',
'SystemRequest' => 'System Request (%s).',
'TicketDynamicFieldUpdate' => 'Updated: %s=%s;%s=%s;%s=%s;',
'TicketLinkAdd' => 'Added link to ticket "%s".',
'TicketLinkDelete' => 'Deleted link to ticket "%s".',
'TimeAccounting' => '%s time unit(s) accounted. Now total %s time unit(s).',
'TitleUpdate' => 'Title updated: Old: "%s", New: "%s"',
'TypeUpdate' => 'Updated Type to %s (ID=%s).',
'Unlock' => 'Unlocked ticket.',
'Unsubscribe' => 'Removed subscription for user "%s".',
'WebRequestCustomer' => 'Customer request via web.'
};
```

## Ticket → Frontend::Agent::Ticket::ViewLocked

### Ticket::Frontend::AgentTicketLockedView###SortBy::Default

Defines the default ticket attribute for ticket sorting in the locked ticket view of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketLockedView'}->{'SortBy::Default'} = 'Age';
```

### Ticket::Frontend::AgentTicketLockedView###Order::Default

Defines the default ticket order in the ticket locked view of the agent interface. Up: oldest on top. Down: latest on top.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketLockedView'}->{'Order::Default'} = 'Up';
```

### Ticket::Frontend::AgentTicketLockedView###DefaultColumns

Columns that can be filtered in the locked view of the agent interface. Possible settings: 0 = Disabled, 1 = Available, 2 = Enabled by default. Note: Only Ticket attributes, Dynamic Fields (DynamicField\_NameX) and Customer attributes (e.g. CustomerUserPhone, CustomerCompanyName, ...) are allowed.

默认值:



```
$Self->{'Ticket::Frontend::AgentTicketLockedView'}->{'DefaultColumns'} = {
  'Age' => '2',
  'Changed' => '1',
  'Created' => '1',
  'CustomerCompanyName' => '1',
  'CustomerID' => '2',
  'CustomerName' => '1',
  'CustomerUserID' => '1',
  'EscalationResponseTime' => '1',
  'EscalationSolutionTime' => '1',
  'EscalationTime' => '1',
  'EscalationUpdateTime' => '1',
  'Lock' => '2',
  'Owner' => '2',
  'PendingTime' => '1',
  'Priority' => '1',
  'Queue' => '2',
  'Responsible' => '1',
  'SLA' => '1',
  'Service' => '1',
  'State' => '2',
  'TicketNumber' => '2',
  'Title' => '2',
  'Type' => '1'
};
```

## Ticket → Frontend::Agent::Ticket::ViewMerge

### Ticket::Frontend::AgentTicketMerge###Permission

Required permissions to use the ticket merge screen of a zoomed ticket in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketMerge'}->{'Permission'} = 'rw';
```

### Ticket::Frontend::AgentTicketMerge###RequiredLock

Defines if a ticket lock is required in the ticket merge screen of a zoomed ticket in the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketMerge'}->{'RequiredLock'} = '1';
```

### Ticket::Frontend::MergeText

When tickets are merged, the customer can be informed per email by setting the check box "Inform Sender". In this text area, you can define a pre-formatted text which can later be modified by the agents.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::MergeText'} = 'Your email with ticket number "<OTRS_TICKET>" is merged to "<OTRS_MERGE_TO_TICKET>";
```

### Ticket::Frontend::AutomaticMergeSubject

When tickets are merged, a note will be added automatically to the ticket which is no longer active. Here you can define the subject of this note (this subject cannot be changed by the agent).

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AutomaticMergeSubject'} = 'Ticket Merged';
```

### **Ticket::Frontend::AutomaticMergeText**

When tickets are merged, a note will be added automatically to the ticket which is no longer active. Here you can define the body of this note (this text cannot be changed by the agent).

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AutomaticMergeText'} = 'Merged Ticket <OTRS_TICKET> to <OTRS_MERGE_TO_TICKET>.';
```

### **Ticket::Frontend::AgentTicketMerge###RichTextWidth**

Defines the width for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketMerge'}->{'RichTextWidth'} = '620';
```

### **Ticket::Frontend::AgentTicketMerge###RichTextHeight**

Defines the height for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketMerge'}->{'RichTextHeight'} = '100';
```

## **Ticket → Frontend::Agent::Ticket::ViewMove**

### **Ticket::Frontend::MoveType**

Determines if the list of possible queues to move to ticket into should be displayed in a dropdown list or in a new window in the agent interface. If "New Window" is set you can add a move note to the ticket.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::MoveType'} = 'form';
```

### **Ticket::Frontend::AgentTicketMove###RequiredLock**

Automatically lock and set owner to current Agent after opening the move ticket screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketMove'}->{'RequiredLock'} = '1';
```

### **Ticket::Frontend::AgentTicketMove###State**

Allows to set a new ticket state in the move ticket screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketMove'}->{'State'} = '1';
```

### **Ticket::Frontend::AgentTicketMove###StateType**

Defines the next state of a ticket after being moved to another queue, in the move ticket screen of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketMove'}->{'StateType'} = [  
  'open',  
  'closed'  
];
```

### **Ticket::Frontend::AgentTicketMove###Priority**

Shows the ticket priority options in the move ticket screen of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketMove'}->{'Priority'} = '0';
```

### **Ticket::Frontend::AgentTicketMove###Note**

Allows adding notes in the ticket free text screen of the agent interface. Can be overwritten by Ticket::Frontend::NeedAccountedTime.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketMove'}->{'Note'} = '0';
```

### **Ticket::Frontend::AgentTicketMove###NoteMandatory**

Sets if note must be filled in by the agent. Can be overwritten by Ticket::Frontend::NeedAccountedTime.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketMove'}->{'NoteMandatory'} = '0';
```

### **Ticket::Frontend::AgentTicketMove###NextScreen**

Determines the next screen after the ticket is moved. LastScreenOverview will return the last overview screen (e.g. search results, queueview, dashboard). TicketZoom will return to the TicketZoom.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketMove'}->{'NextScreen'} = 'TicketZoom';
```

### **Ticket::Frontend::AgentTicketMove###Subject**

Sets the default subject for notes added in the ticket move screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketMove'}->{'Subject'} = '';
```

### **Ticket::Frontend::AgentTicketMove###Body**

Sets the default body text for notes added in the ticket move screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketMove'}->{'Body'} = '';
```

### **Ticket::Frontend::AgentTicketMove###RichTextWidth**

Defines the width for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketMove'}->{'RichTextWidth'} = '620';
```

### **Ticket::Frontend::AgentTicketMove###RichTextHeight**

Defines the height for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketMove'}->{'RichTextHeight'} = '100';
```

### **Ticket::Frontend::AgentTicketMove###DynamicField**

Dynamic fields shown in the ticket move screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketMove'}->{'DynamicField'} = {};
```

## **Ticket → Frontend::Agent::Ticket::ViewNote**

### **Ticket::Frontend::AgentTicketNote###Permission**

Required permissions to use the ticket note screen in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketNote'}->{'Permission'} = 'note';
```

### **Ticket::Frontend::AgentTicketNote###RequiredLock**

Defines if a ticket lock is required in the ticket note screen of the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketNote'}->{'RequiredLock'} = '0';
```

### **Ticket::Frontend::AgentTicketNote###TicketType**

Sets the ticket type in the ticket note screen of the agent interface (Ticket::Type needs to be activated).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketNote'}->{'TicketType'} = '0';
```

### **Ticket::Frontend::AgentTicketNote###Service**

Sets the service in the ticket note screen of the agent interface (Ticket::Service needs to be activated).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketNote'}->{'Service'} = '0';
```

### **Ticket::Frontend::AgentTicketNote###ServiceMandatory**

Sets if service must be selected by the agent.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketNote'}->{'ServiceMandatory'} = '0';
```

### **Ticket::Frontend::AgentTicketNote###SLAMandatory**

Sets if SLA must be selected by the agent.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketNote'}->{'SLAMandatory'} = '0';
```

### **Ticket::Frontend::AgentTicketNote###Queue**

Sets the queue in the ticket note screen of a zoomed ticket in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketNote'}->{'Queue'} = '0';
```

### **Ticket::Frontend::AgentTicketNote###Owner**

Sets the ticket owner in the ticket note screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketNote'}->{'Owner'} = '0';
```

### **Ticket::Frontend::AgentTicketNote###OwnerMandatory**

Sets if ticket owner must be selected by the agent.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketNote'}->{'OwnerMandatory'} = '0';
```

### **Ticket::Frontend::AgentTicketNote###Responsible**

Sets the responsible agent of the ticket in the ticket note screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketNote'}->{'Responsible'} = '0';
```

### **Ticket::Frontend::AgentTicketNote###State**

Sets the state of a ticket in the ticket note screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketNote'}->{'State'} = '0';
```

### **Ticket::Frontend::AgentTicketNote###StateType**

Defines the next state of a ticket after adding a note, in the ticket note screen of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketNote'}->{'StateType'} = [  
  'open',  
  'closed',  
  'pending reminder',  
  'pending auto'  
];
```

### **Ticket::Frontend::AgentTicketNote###StateDefault**

Defines the default next state of a ticket after adding a note, in the ticket note screen of the agent interface.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketNote'}->{'StateDefault'} = 'open';
```

### **Ticket::Frontend::AgentTicketNote###Note**

Allows adding notes in the ticket note screen of the agent interface. Can be overwritten by Ticket::Frontend::NeedAccountedTime.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketNote'}->{'Note'} = '1';
```

### **Ticket::Frontend::AgentTicketNote###NoteMandatory**

Sets if note must be filled in by the agent. Can be overwritten by Ticket::Frontend::NeedAccountedTime.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketNote'}->{'NoteMandatory'} = '1';
```

### **Ticket::Frontend::AgentTicketNote###Subject**

Sets the default subject for notes added in the ticket note screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketNote'}->{'Subject'} = '';
```

### **Ticket::Frontend::AgentTicketNote###Body**

Sets the default body text for notes added in the ticket note screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketNote'}->{'Body'} = '';
```

### **Ticket::Frontend::AgentTicketNote###InvolvedAgent**

Shows a list of all the involved agents on this ticket, in the ticket note screen of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketNote'}->{'InvolvedAgent'} = '0';
```

### **Ticket::Frontend::AgentTicketNote###InformAgent**

Shows a list of all the possible agents (all agents with note permissions on the queue/ticket) to determine who should be informed about this note, in the ticket note screen of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketNote'}->{'InformAgent'} = '0';
```

### **Ticket::Frontend::AgentTicketNote###ArticleTypeDefault**

Defines the default type of the note in the ticket note screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketNote'}->{'ArticleTypeDefault'} = 'note-internal';
```

### **Ticket::Frontend::AgentTicketNote###ArticleTypes**

Specifies the available note types for this ticket mask. If the option is deselected, ArticleTypeDefault is used and the option is removed from the mask.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketNote'}->{'ArticleTypes'} = {  
  'note-external' => '1',  
  'note-internal' => '1',  
  'note-report' => '0'  
};
```

### **Ticket::Frontend::AgentTicketNote###Priority**

Shows the ticket priority options in the ticket note screen of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketNote'}->{'Priority'} = '0';
```

### **Ticket::Frontend::AgentTicketNote###PriorityDefault**

Defines the default ticket priority in the ticket note screen of the agent interface.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketNote'}->{'PriorityDefault'} = '3 normal';
```

### **Ticket::Frontend::AgentTicketNote###Title**

Shows the title fields in the ticket note screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketNote'}->{'Title'} = '0';
```

### **Ticket::Frontend::AgentTicketNote###HistoryType**

Defines the history type for the ticket note screen action, which gets used for ticket history in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketNote'}->{'HistoryType'} = 'AddNote';
```

### **Ticket::Frontend::AgentTicketNote###HistoryComment**

Defines the history comment for the ticket note screen action, which gets used for ticket history in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketNote'}->{'HistoryComment'} = '%Note';
```

### **Ticket::Frontend::AgentTicketNote###DynamicField**

Dynamic fields shown in the ticket note screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketNote'}->{'DynamicField'} = {};
```

### **Ticket::Frontend::AgentTicketNote###RichTextWidth**

Defines the width for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketNote'}->{'RichTextWidth'} = '620';
```

### **Ticket::Frontend::AgentTicketNote###RichTextHeight**

Defines the height for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketNote'}->{'RichTextHeight'} = '100';
```

## **Ticket → Frontend::Agent::Ticket::ViewOwner**

### **Ticket::Frontend::AgentTicketOwner###Permission**

Required permissions to use the ticket owner screen of a zoomed ticket in the agent interface.

不能使这个设置失效。



默认值:

```
$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'Permission'} = 'owner';
```

### **Ticket::Frontend::AgentTicketOwner###RequiredLock**

Defines if a ticket lock is required in the ticket owner screen of a zoomed ticket in the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'RequiredLock'} = '0';
```

### **Ticket::Frontend::AgentTicketOwner###TicketType**

Sets the ticket type in the ticket owner screen of a zoomed ticket in the agent interface (Ticket::Type needs to be activated).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'TicketType'} = '0';
```

### **Ticket::Frontend::AgentTicketOwner###Service**

Sets the service in the ticket owner screen of a zoomed ticket in the agent interface (Ticket::Service needs to be activated).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'Service'} = '0';
```

### **Ticket::Frontend::AgentTicketOwner###ServiceMandatory**

Sets if service must be selected by the agent.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'ServiceMandatory'} = '0';
```

### **Ticket::Frontend::AgentTicketOwner###SLAMandatory**

Sets if SLA must be selected by the agent.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'SLAMandatory'} = '0';
```

### **Ticket::Frontend::AgentTicketOwner###Queue**

Sets the queue in the ticket owner screen of a zoomed ticket in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'Queue'} = '0';
```

### **Ticket::Frontend::AgentTicketOwner###Owner**

Sets the ticket owner in the ticket owner screen of a zoomed ticket in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'Owner'} = '1';
```

### **Ticket::Frontend::AgentTicketOwner###OwnerMandatory**

Sets if ticket owner must be selected by the agent.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'OwnerMandatory'} = '1';
```

### **Ticket::Frontend::AgentTicketOwner###Responsible**

Sets the responsible agent of the ticket in the ticket owner screen of a zoomed ticket in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'Responsible'} = '0';
```

### **Ticket::Frontend::AgentTicketOwner###State**

Sets the state of the ticket in the ticket owner screen of a zoomed ticket in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'State'} = '0';
```

### **Ticket::Frontend::AgentTicketOwner###StateType**

Defines the next state of a ticket after adding a note, in the ticket owner screen of a zoomed ticket in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'StateType'} = [  
  'open',  
  'pending reminder',  
  'pending auto'  
];
```

### **Ticket::Frontend::AgentTicketOwner###StateDefault**

Defines the default next state of a ticket after adding a note, in the ticket owner screen of a zoomed ticket in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'StateDefault'} = 'open';
```

### **Ticket::Frontend::AgentTicketOwner###Note**

Allows adding notes in the ticket owner screen of a zoomed ticket in the agent interface. Can be overwritten by Ticket::Frontend::NeedAccountedTime.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'Note'} = '1';
```

### **Ticket::Frontend::AgentTicketOwner###NoteMandatory**

Sets if note must be filled in by the agent. Can be overwritten by Ticket::Frontend::NeedAccountedTime.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'NoteMandatory'} = '1';
```

### **Ticket::Frontend::AgentTicketOwner###Subject**

Sets the default subject for notes added in the ticket owner screen of a zoomed ticket in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'Subject'} = '';
```

### **Ticket::Frontend::AgentTicketOwner###Body**

Sets the default body text for notes added in the ticket owner screen of a zoomed ticket in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'Body'} = '';
```

### **Ticket::Frontend::AgentTicketOwner###InvolvedAgent**

Shows a list of all the involved agents on this ticket, in the ticket owner screen of a zoomed ticket in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'InvolvedAgent'} = '0';
```

### **Ticket::Frontend::AgentTicketOwner###InformAgent**

Shows a list of all the possible agents (all agents with note permissions on the queue/ticket) to determine who should be informed about this note, in the ticket owner screen of a zoomed ticket in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'InformAgent'} = '0';
```

### **Ticket::Frontend::AgentTicketOwner###ArticleTypeDefault**

Defines the default type of the note in the ticket owner screen of a zoomed ticket in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'ArticleTypeDefault'} = 'note-internal';
```

### **Ticket::Frontend::AgentTicketOwner###ArticleTypes**

Specifies the available note types for this ticket mask. If the option is deselected, ArticleTypeDefault is used and the option is removed from the mask.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'ArticleTypes'} = {
  'note-external' => '0',
  'note-internal' => '1',
  'note-report' => '0'
};
```

### **Ticket::Frontend::AgentTicketOwner###Priority**

Shows the ticket priority options in the ticket owner screen of a zoomed ticket in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'Priority'} = '0';
```

### **Ticket::Frontend::AgentTicketOwner###PriorityDefault**

Defines the default ticket priority in the ticket owner screen of a zoomed ticket in the agent interface.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'PriorityDefault'} = '3 normal';
```

### **Ticket::Frontend::AgentTicketOwner###Title**

Shows the title fields in the ticket owner screen of a zoomed ticket in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'Title'} = '0';
```

### **Ticket::Frontend::AgentTicketOwner###HistoryType**

Defines the history type for the ticket owner screen action, which gets used for ticket history in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'HistoryType'} = 'AddNote';
```

### **Ticket::Frontend::AgentTicketOwner###HistoryComment**

Defines the history comment for the ticket owner screen action, which gets used for ticket history in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'HistoryComment'} = '%Owner';
```

### **Ticket::Frontend::AgentTicketOwner###DynamicField**

Dynamic fields shown in the ticket owner screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'DynamicField'} = {};
```

### **Ticket::Frontend::AgentTicketOwner###RichTextWidth**

Defines the width for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'RichTextWidth'} = '620';
```

### **Ticket::Frontend::AgentTicketOwner###RichTextHeight**

Defines the height for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'RichTextHeight'} = '100';
```

### **Ticket → Frontend::Agent::Ticket::ViewPending**

#### **Ticket::Frontend::AgentTicketPending###Permission**

Required permissions to use the ticket pending screen of a zoomed ticket in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPending'}->{'Permission'} = 'pending';
```

#### **Ticket::Frontend::AgentTicketPending###RequiredLock**

Defines if a ticket lock is required in the ticket pending screen of a zoomed ticket in the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPending'}->{'RequiredLock'} = '1';
```

#### **Ticket::Frontend::AgentTicketPending###TicketType**

Sets the ticket type in the ticket pending screen of a zoomed ticket in the agent interface (Ticket::Type needs to be activated).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPending'}->{'TicketType'} = '0';
```

#### **Ticket::Frontend::AgentTicketPending###Service**

Sets the service in the ticket pending screen of a zoomed ticket in the agent interface (Ticket::Service needs to be activated).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPending'}->{'Service'} = '0';
```

#### **Ticket::Frontend::AgentTicketPending###ServiceMandatory**

Sets if service must be selected by the agent.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPending'}->{'ServiceMandatory'} = '0';
```

#### **Ticket::Frontend::AgentTicketPending###SLAMandatory**

Sets if SLA must be selected by the agent.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPending'}->{'SLAMandatory'} = '0';
```

### **Ticket::Frontend::AgentTicketPending###Queue**

Sets the queue in the ticket pending screen of a zoomed ticket in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPending'}->{'Queue'} = '0';
```

### **Ticket::Frontend::AgentTicketPending###Owner**

Sets the ticket owner in the ticket pending screen of a zoomed ticket in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPending'}->{'Owner'} = '0';
```

### **Ticket::Frontend::AgentTicketPending###OwnerMandatory**

Sets if ticket owner must be selected by the agent.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPending'}->{'OwnerMandatory'} = '0';
```

### **Ticket::Frontend::AgentTicketPending###Responsible**

Sets the responsible agent of the ticket in the ticket pending screen of a zoomed ticket in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPending'}->{'Responsible'} = '0';
```

### **Ticket::Frontend::AgentTicketPending###State**

Sets the state of the ticket in the ticket pending screen of a zoomed ticket in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPending'}->{'State'} = '1';
```

### **Ticket::Frontend::AgentTicketPending###StateType**

Defines the next state of a ticket after adding a note, in the ticket pending screen of a zoomed ticket in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPending'}->{'StateType'} = [  
  'pending reminder',  
  'pending auto'  
];
```

### **Ticket::Frontend::AgentTicketPending###StateDefault**

Defines the default next state of a ticket after adding a note, in the ticket pending screen of a zoomed ticket in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPending'}->{'StateDefault'} = 'pending reminder';
```

### **Ticket::Frontend::AgentTicketPending###Note**

Allows adding notes in the ticket pending screen of a zoomed ticket in the agent interface. Can be overwritten by Ticket::Frontend::NeedAccountedTime.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPending'}->{'Note'} = '1';
```

### **Ticket::Frontend::AgentTicketPending###NoteMandatory**

Sets if note must be filled in by the agent. Can be overwritten by Ticket::Frontend::NeedAccountedTime.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPending'}->{'NoteMandatory'} = '1';
```

### **Ticket::Frontend::AgentTicketPending###Subject**

Sets the default subject for notes added in the ticket pending screen of a zoomed ticket in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPending'}->{'Subject'} = '';
```

### **Ticket::Frontend::AgentTicketPending###Body**

Sets the default body text for notes added in the ticket pending screen of a zoomed ticket in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPending'}->{'Body'} = '';
```

### **Ticket::Frontend::AgentTicketPending###InvolvedAgent**

Shows a list of all the involved agents on this ticket, in the ticket pending screen of a zoomed ticket in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPending'}->{'InvolvedAgent'} = '0';
```

### **Ticket::Frontend::AgentTicketPending###InformAgent**

Shows a list of all the possible agents (all agents with note permissions on the queue/ticket) to determine who should be informed about this note, in the ticket pending screen of a zoomed ticket in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPending'}->{'InformAgent'} = '0';
```

### **Ticket::Frontend::AgentTicketPending###ArticleTypeDefault**

Defines the default type of the note in the ticket pending screen of a zoomed ticket in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPending'}->{'ArticleTypeDefault'} = 'note-internal';
```

### **Ticket::Frontend::AgentTicketPending###ArticleTypes**

Specifies the available note types for this ticket mask. If the option is deselected, ArticleTypeDefault is used and the option is removed from the mask.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPending'}->{'ArticleTypes'} = {  
  'note-external' => '0',  
  'note-internal' => '1',  
  'note-report' => '0'  
};
```

### **Ticket::Frontend::AgentTicketPending###Priority**

Shows the ticket priority options in the ticket pending screen of a zoomed ticket in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPending'}->{'Priority'} = '0';
```

### **Ticket::Frontend::AgentTicketPending###PriorityDefault**

Defines the default ticket priority in the ticket pending screen of a zoomed ticket in the agent interface.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPending'}->{'PriorityDefault'} = '3 normal';
```

### **Ticket::Frontend::AgentTicketPending###Title**

Shows the title fields in the ticket pending screen of a zoomed ticket in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPending'}->{'Title'} = '0';
```

### **Ticket::Frontend::AgentTicketPending###HistoryType**

Defines the history type for the ticket pending screen action, which gets used for ticket history in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPending'}->{'HistoryType'} = 'AddNote';
```

### **Ticket::Frontend::AgentTicketPending###HistoryComment**

Defines the history comment for the ticket pending screen action, which gets used for ticket history in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPending'}->{'HistoryComment'} = '%Pending';
```



### **Ticket::Frontend::AgentTicketPending###DynamicField**

Dynamic fields shown in the ticket pending screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPending'}->{'DynamicField'} = {};
```

### **Ticket::Frontend::AgentTicketPending###RichTextWidth**

Defines the width for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPending'}->{'RichTextWidth'} = '620';
```

### **Ticket::Frontend::AgentTicketPending###RichTextHeight**

Defines the height for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPending'}->{'RichTextHeight'} = '100';
```

## **Ticket → Frontend::Agent::Ticket::ViewPhoneInbound**

### **Ticket::Frontend::AgentTicketPhoneInbound###Permission**

Required permissions to use the ticket phone inbound screen in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhoneInbound'}->{'Permission'} = 'phone';
```

### **Ticket::Frontend::AgentTicketPhoneInbound###RequiredLock**

Defines if a ticket lock is required in the ticket phone inbound screen of the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhoneInbound'}->{'RequiredLock'} = '0';
```

### **Ticket::Frontend::AgentTicketPhoneInbound###ArticleType**

Defines the default type of the note in the ticket phone inbound screen of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhoneInbound'}->{'ArticleType'} = 'phone';
```

### **Ticket::Frontend::AgentTicketPhoneInbound###SenderType**

Defines the default sender type for phone tickets in the ticket phone inbound screen of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhoneInbound'}->{'SenderType'} = 'customer';
```

### **Ticket::Frontend::AgentTicketPhoneInbound###Subject**

Defines the default subject for phone tickets in the ticket phone inbound screen of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhoneInbound'}->{'Subject'} = '';
```

### **Ticket::Frontend::AgentTicketPhoneInbound###Body**

Defines the default note body text for phone tickets in the ticket phone inbound screen of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhoneInbound'}->{'Body'} = '';
```

### **Ticket::Frontend::AgentTicketPhoneInbound###State**

Defines the default ticket next state after adding a phone note in the ticket phone inbound screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhoneInbound'}->{'State'} = 'open';
```

### **Ticket::Frontend::AgentTicketPhoneInbound###StateType**

Next possible ticket states after adding a phone note in the ticket phone inbound screen of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhoneInbound'}->{'StateType'} = [  
  'open',  
  'pending auto',  
  'pending reminder',  
  'closed'  
];
```

### **Ticket::Frontend::AgentTicketPhoneInbound###HistoryType**

Defines the history type for the ticket phone inbound screen action, which gets used for ticket history in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhoneInbound'}->{'HistoryType'} =  
'PhoneCallCustomer';
```

### **Ticket::Frontend::AgentTicketPhoneInbound###HistoryComment**

Defines the history comment for the ticket phone inbound screen action, which gets used for ticket history in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhoneInbound'}->{'HistoryComment'} = '';
```

### **Ticket::Frontend::AgentTicketPhoneInbound###DynamicField**

Dynamic fields shown in the ticket phone inbound screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhoneInbound'}->{'DynamicField'} = {};
```

### **Ticket::Frontend::AgentTicketPhoneInbound###RichTextWidth**

Defines the width for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhoneInbound'}->{'RichTextWidth'} = '475';
```

### **Ticket::Frontend::AgentTicketPhoneInbound###RichTextHeight**

Defines the height for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhoneInbound'}->{'RichTextHeight'} = '200';
```

## **Ticket → Frontend::Agent::Ticket::ViewPhoneNew**

### **Ticket::Frontend::AgentTicketPhone###Priority**

Sets the default priority for new phone tickets in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'Priority'} = '3 normal';
```

### **Ticket::Frontend::AgentTicketPhone###ArticleType**

Sets the default article type for new phone tickets in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'ArticleType'} = 'phone';
```

### **Ticket::Frontend::AgentTicketPhone###SenderType**

Sets the default sender type for new phone ticket in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'SenderType'} = 'customer';
```

### **Ticket::Frontend::AgentTicketPhone::CustomerIDReadOnly**

Controls if CustomerID is read-only in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhone::CustomerIDReadOnly'} = '1';
```

### **Ticket::Frontend::AgentTicketPhone::AllowMultipleFrom**

Controls if more than one from entry can be set in the new phone ticket in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhone::AllowMultipleFrom'} = '1';
```

### **Ticket::Frontend::AgentTicketPhone###Subject**

Sets the default subject for new phone tickets (e.g. 'Phone call') in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'Subject'} = '';
```

### **Ticket::Frontend::AgentTicketPhone###Body**

Sets the default note text for new telephone tickets. E.g 'New ticket via call' in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'Body'} = '';
```

### **Ticket::Frontend::AgentTicketPhone###StateDefault**

Sets the default next state for new phone tickets in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'StateDefault'} = 'open';
```

### **Ticket::Frontend::AgentTicketPhone###StateType**

Determines the next possible ticket states, after the creation of a new phone ticket in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'StateType'} = [  
'open',
```

```
'pending auto',
'pending reminder',
'closed'
];
```

### **Ticket::Frontend::AgentTicketPhone###HistoryType**

Defines the history type for the phone ticket screen action, which gets used for ticket history in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'HistoryType'} = 'PhoneCallCustomer';
```

### **Ticket::Frontend::AgentTicketPhone###HistoryComment**

Defines the history comment for the phone ticket screen action, which gets used for ticket history in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'HistoryComment'} = '';
```

### **Ticket::Frontend::AgentTicketPhone###SplitLinkType**

Sets the default link type of splitted tickets in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'SplitLinkType'} = {
  'Direction' => 'Target',
  'LinkType' => 'ParentChild'
};
```

### **Ticket::Frontend::AgentTicketPhone###ServiceMandatory**

Sets if service must be selected by the agent.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'ServiceMandatory'} = '0';
```

### **Ticket::Frontend::AgentTicketPhone###SLAMandatory**

Sets if SLA must be selected by the agent.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'SLAMandatory'} = '0';
```

### **Ticket::Frontend::AgentTicketPhone###DynamicField**

Dynamic fields shown in the ticket phone screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'DynamicField'} = {};
```

### **Ticket::Frontend::AgentTicketPhone###RichTextWidth**

Defines the width for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'RichTextWidth'} = '620';
```

### **Ticket::Frontend::AgentTicketPhone###RichTextHeight**

Defines the height for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'RichTextHeight'} = '320';
```

## **Ticket → Frontend::Agent::Ticket::ViewPhoneOutbound**

### **Ticket::Frontend::AgentTicketPhoneOutbound###Permission**

Required permissions to use the ticket phone outbound screen in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}->{'Permission'} = 'phone';
```

### **Ticket::Frontend::AgentTicketPhoneOutbound###RequiredLock**

Defines if a ticket lock is required in the ticket phone outbound screen of the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}->{'RequiredLock'} = '1';
```

### **Ticket::Frontend::AgentTicketPhoneOutbound###ArticleType**

Defines the default type of the note in the ticket phone outbound screen of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}->{'ArticleType'} = 'phone';
```

### **Ticket::Frontend::AgentTicketPhoneOutbound###SenderType**

Defines the default sender type for phone tickets in the ticket phone outbound screen of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}->{'SenderType'} = 'agent';
```

### **Ticket::Frontend::AgentTicketPhoneOutbound###Subject**

Defines the default subject for phone tickets in the ticket phone outbound screen of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}->{'Subject'} = '';
```

### **Ticket::Frontend::AgentTicketPhoneOutbound###Body**

Defines the default note body text for phone tickets in the ticket phone outbound screen of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}->{'Body'} = '';
```

### **Ticket::Frontend::AgentTicketPhoneOutbound###State**

Defines the default ticket next state after adding a phone note in the ticket phone outbound screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}->{'State'} = 'closed  
successful';
```

### **Ticket::Frontend::AgentTicketPhoneOutbound###StateType**

Next possible ticket states after adding a phone note in the ticket phone outbound screen of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}->{'StateType'} = [  
  'open',  
  'pending auto',  
  'pending reminder',  
  'closed'  
];
```

### **Ticket::Frontend::AgentTicketPhoneOutbound###HistoryType**

Defines the history type for the ticket phone outbound screen action, which gets used for ticket history in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}->{'HistoryType'} =  
'PhoneCallAgent';
```

### **Ticket::Frontend::AgentTicketPhoneOutbound###HistoryComment**

Defines the history comment for the ticket phone outbound screen action, which gets used for ticket history in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}->{'HistoryComment'} = '';
```

### **Ticket::Frontend::AgentTicketPhoneOutbound###DynamicField**

Dynamic fields shown in the ticket phone outbound screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}->{'DynamicField'} = {};
```

### **Ticket::Frontend::AgentTicketPhoneOutbound###RichTextWidth**

Defines the width for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}->{'RichTextWidth'} = '475';
```

### **Ticket::Frontend::AgentTicketPhoneOutbound###RichTextHeight**

Defines the height for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}->{'RichTextHeight'} = '200';
```

## **Ticket → Frontend::Agent::Ticket::ViewPrint**

### **Ticket::Frontend::AgentTicketPrint###DynamicField**

Dynamic fields shown in the ticket print screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPrint'}->{'DynamicField'} = {};
```

## **Ticket → Frontend::Agent::Ticket::ViewPriority**

### **Ticket::Frontend::AgentTicketPriority###Permission**

Required permissions to use the ticket priority screen of a zoomed ticket in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'Permission'} = 'priority';
```

### **Ticket::Frontend::AgentTicketPriority###RequiredLock**

Defines if a ticket lock is required in the ticket priority screen of a zoomed ticket in the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'RequiredLock'} = '1';
```

### **Ticket::Frontend::AgentTicketPriority###TicketType**

Sets the ticket type in the ticket priority screen of a zoomed ticket in the agent interface (Ticket::Type needs to be activated).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'TicketType'} = '0';
```



### **Ticket::Frontend::AgentTicketPriority###Service**

Sets the service in the ticket priority screen of a zoomed ticket in the agent interface (Ticket::Service needs to be activated).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'Service'} = '0';
```

### **Ticket::Frontend::AgentTicketPriority###ServiceMandatory**

Sets if service must be selected by the agent.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'ServiceMandatory'} = '0';
```

### **Ticket::Frontend::AgentTicketPriority###SLAMandatory**

Sets if SLA must be selected by the agent.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'SLAMandatory'} = '0';
```

### **Ticket::Frontend::AgentTicketPriority###Queue**

Sets the queue in the ticket priority screen of a zoomed ticket in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'Queue'} = '0';
```

### **Ticket::Frontend::AgentTicketPriority###Owner**

Sets the ticket owner in the ticket priority screen of a zoomed ticket in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'Owner'} = '0';
```

### **Ticket::Frontend::AgentTicketPriority###OwnerMandatory**

Sets if ticket owner must be selected by the agent.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'OwnerMandatory'} = '0';
```

### **Ticket::Frontend::AgentTicketPriority###Responsible**

Sets the responsible agent of the ticket in the ticket priority screen of a zoomed ticket in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'Responsible'} = '0';
```

### **Ticket::Frontend::AgentTicketPriority###State**

Sets the state of the ticket in the ticket priority screen of a zoomed ticket in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'State'} = '0';
```

### **Ticket::Frontend::AgentTicketPriority###StateType**

Defines the next state of a ticket after adding a note, in the ticket priority screen of a zoomed ticket in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'StateType'} = [  
  'open',  
  'pending reminder',  
  'pending auto'  
];
```

### **Ticket::Frontend::AgentTicketPriority###StateDefault**

Defines the default next state of a ticket after adding a note, in the ticket priority screen of a zoomed ticket in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'StateDefault'} = 'open';
```

### **Ticket::Frontend::AgentTicketPriority###Note**

Allows adding notes in the ticket priority screen of a zoomed ticket in the agent interface. Can be overwritten by Ticket::Frontend::NeedAccountedTime.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'Note'} = '1';
```

### **Ticket::Frontend::AgentTicketPriority###NoteMandatory**

Sets if note must be filled in by the agent. Can be overwritten by Ticket::Frontend::NeedAccountedTime.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'NoteMandatory'} = '1';
```

### **Ticket::Frontend::AgentTicketPriority###Subject**

Sets the default subject for notes added in the ticket priority screen of a zoomed ticket in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'Subject'} = '';
```

### **Ticket::Frontend::AgentTicketPriority###Body**

Sets the default body text for notes added in the ticket priority screen of a zoomed ticket in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'Body'} = '';
```

### **Ticket::Frontend::AgentTicketPriority###InvolvedAgent**

Shows a list of all the involved agents on this ticket, in the ticket priority screen of a zoomed ticket in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'InvolvedAgent'} = '0';
```

### **Ticket::Frontend::AgentTicketPriority###InformAgent**

Shows a list of all the possible agents (all agents with note permissions on the queue/ticket) to determine who should be informed about this note, in the ticket priority screen of a zoomed ticket in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'InformAgent'} = '0';
```

### **Ticket::Frontend::AgentTicketPriority###ArticleTypeDefault**

Defines the default type of the note in the ticket priority screen of a zoomed ticket in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'ArticleTypeDefault'} = 'note-internal';
```

### **Ticket::Frontend::AgentTicketPriority###ArticleTypes**

Specifies the available note types for this ticket mask. If the option is deselected, ArticleTypeDefault is used and the option is removed from the mask.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'ArticleTypes'} = {  
  'note-external' => '0',  
  'note-internal' => '1',  
  'note-report' => '0'  
};
```

### **Ticket::Frontend::AgentTicketPriority###Priority**

Shows the ticket priority options in the ticket priority screen of a zoomed ticket in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'Priority'} = '1';
```

### **Ticket::Frontend::AgentTicketPriority###PriorityDefault**

Defines the default ticket priority in the ticket priority screen of a zoomed ticket in the agent interface.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'PriorityDefault'} = '3 normal';
```

### **Ticket::Frontend::AgentTicketPriority###Title**

Shows the title fields in the ticket priority screen of a zoomed ticket in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'Title'} = '0';
```

### **Ticket::Frontend::AgentTicketPriority###HistoryType**

Defines the history type for the ticket priority screen action, which gets used for ticket history in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'HistoryType'} = 'AddNote';
```

### **Ticket::Frontend::AgentTicketPriority###HistoryComment**

Defines the history comment for the ticket priority screen action, which gets used for ticket history in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'HistoryComment'} = '%Priority';
```

### **Ticket::Frontend::AgentTicketPriority###DynamicField**

Dynamic fields shown in the ticket priority screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'DynamicField'} = {};
```

### **Ticket::Frontend::AgentTicketPriority###RichTextWidth**

Defines the width for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'RichTextWidth'} = '620';
```

### **Ticket::Frontend::AgentTicketPriority###RichTextHeight**

Defines the height for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'RichTextHeight'} = '100';
```

## **Ticket → Frontend::Agent::Ticket::ViewQueue**

### **Ticket::Frontend::AgentTicketQueue###StripEmptyLines**

Strips empty lines on the ticket preview in the queue view.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketQueue'}->{'StripEmptyLines'} = '0';
```

### **Ticket::Frontend::AgentTicketQueue###ViewAllPossibleTickets**

Shows all both ro and rw queues in the queue view.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketQueue'}->{'ViewAllPossibleTickets'} = '0';
```

### **Ticket::Frontend::AgentTicketQueue###HideEmptyQueues**

Show queues even when only locked tickets are in.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketQueue'}->{'HideEmptyQueues'} = '0';
```

### **Ticket::Frontend::AgentTicketQueue###HighlightAge1**

Sets the age in minutes (first level) for highlighting queues that contain untouched tickets.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketQueue'}->{'HighlightAge1'} = '1440';
```

### **Ticket::Frontend::AgentTicketQueue###HighlightAge2**

Sets the age in minutes (second level) for highlighting queues that contain untouched tickets.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketQueue'}->{'HighlightAge2'} = '2880';
```

### **Ticket::Frontend::AgentTicketQueue###Blink**

Activates a blinking mechanism of the queue that contains the oldest ticket.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketQueue'}->{'Blink'} = '1';
```

### **Ticket::Frontend::AgentTicketQueue###UseSubQueues**

Include tickets of subqueues per default when selecting a queue.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketQueue'}->{'UseSubQueues'} = '0';
```

### **Ticket::Frontend::AgentTicketQueue###QueueSort**

Sorts the tickets (ascendingly or descendingly) when a single queue is selected in the queue view and after the tickets are sorted by priority. Values: 0 = ascending (oldest on top, default), 1 = descending (youngest on top). Use the QueueID for the key and 0 or 1 for value.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketQueue'}->{'QueueSort'} = {
```

```
'3' => '0',
'7' => '1'
};
```

### **Ticket::Frontend::AgentTicketQueue###SortBy::Default**

Defines the default sort criteria for all queues displayed in the queue view.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketQueue'}->{'SortBy::Default'} = 'Age';
```

### **Ticket::Frontend::AgentTicketQueue###PreSort::ByPriority**

Defines if a pre-sorting by priority should be done in the queue view.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketQueue'}->{'PreSort::ByPriority'} = '1';
```

### **Ticket::Frontend::AgentTicketQueue###Order::Default**

Defines the default sort order for all queues in the queue view, after priority sort.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketQueue'}->{'Order::Default'} = 'Up';
```

### **Ticket::Frontend::AgentTicketQueue###DefaultColumns**

Columns that can be filtered in the queue view of the agent interface. Possible settings: 0 = Disabled, 1 = Available, 2 = Enabled by default. Note: Only Ticket attributes, Dynamic Fields (DynamicField\_NameX) and Customer attributes (e.g. CustomerUserPhone, CustomerCompanyName, ...) are allowed.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketQueue'}->{'DefaultColumns'} = {
  'Age' => '2',
  'Changed' => '1',
  'Created' => '1',
  'CustomerCompanyName' => '1',
  'CustomerID' => '2',
  'CustomerName' => '1',
  'CustomerUserID' => '1',
  'EscalationResponseTime' => '1',
  'EscalationSolutionTime' => '1',
  'EscalationTime' => '1',
  'EscalationUpdateTime' => '1',
  'Lock' => '2',
  'Owner' => '2',
  'PendingTime' => '1',
  'Priority' => '1',
  'Queue' => '2',
  'Responsible' => '1',
  'SLA' => '1',
  'Service' => '1',
  'State' => '2',
  'TicketNumber' => '2',
  'Title' => '2',
  'Type' => '1'
};
```

## **Ticket → Frontend::Agent::Ticket::ViewResponsible**

### **Ticket::Frontend::AgentTicketResponsibleView###SortBy::Default**

Defines the default ticket attribute for ticket sorting in the responsible view of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketResponsibleView'}->{'SortBy::Default'} = 'Age';
```

### **Ticket::Frontend::AgentTicketResponsibleView###Order::Default**

Defines the default ticket order in the responsible view of the agent interface. Up: oldest on top. Down: latest on top.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketResponsibleView'}->{'Order::Default'} = 'Up';
```

### **Ticket::Frontend::AgentTicketResponsible###Permission**

Required permissions to use the ticket responsible screen in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'Permission'} = 'responsible';
```

### **Ticket::Frontend::AgentTicketResponsible###RequiredLock**

Defines if a ticket lock is required in the ticket responsible screen of the agent interface (if the ticket isn't locked yet, the ticket gets locked and the current agent will be set automatically as its owner).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'RequiredLock'} = '0';
```

### **Ticket::Frontend::AgentTicketResponsible###TicketType**

Sets the ticket type in the ticket responsible screen of the agent interface (Ticket::Type needs to be activated).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'TicketType'} = '0';
```

### **Ticket::Frontend::AgentTicketResponsible###Service**

Sets the service in the ticket responsible screen of the agent interface (Ticket::Service needs to be activated).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'Service'} = '0';
```

### **Ticket::Frontend::AgentTicketResponsible###ServiceMandatory**

Sets if service must be selected by the agent.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'ServiceMandatory'} = '0';
```

### **Ticket::Frontend::AgentTicketResponsible###SLAMandatory**

Sets if SLA must be selected by the agent.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'SLAMandatory'} = '0';
```

### **Ticket::Frontend::AgentTicketResponsible###Queue**

Sets the queue in the ticket responsible screen of a zoomed ticket in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'Queue'} = '0';
```

### **Ticket::Frontend::AgentTicketResponsible###Owner**

Sets the ticket owner in the ticket responsible screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'Owner'} = '0';
```

### **Ticket::Frontend::AgentTicketResponsible###OwnerMandatory**

Sets if ticket owner must be selected by the agent.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'OwnerMandatory'} = '0';
```

### **Ticket::Frontend::AgentTicketResponsible###Responsible**

Sets the responsible agent of the ticket in the ticket responsible screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'Responsible'} = '1';
```

### **Ticket::Frontend::AgentTicketResponsible###State**

Sets the state of a ticket in the ticket responsible screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'State'} = '0';
```

### **Ticket::Frontend::AgentTicketResponsible###StateType**

Defines the next state of a ticket after adding a note, in the ticket responsible screen of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'StateType'} = [  
'open',
```



```
'pending reminder',  
'pending auto'  
];
```

### **Ticket::Frontend::AgentTicketResponsible###StateDefault**

Defines the default next state of a ticket after adding a note, in the ticket responsible screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'StateDefault'} = 'open';
```

### **Ticket::Frontend::AgentTicketResponsible###Note**

Allows adding notes in the ticket responsible screen of the agent interface. Can be overwritten by Ticket::Frontend::NeedAccountedTime.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'Note'} = '1';
```

### **Ticket::Frontend::AgentTicketResponsible###NoteMandatory**

Sets if note must be filled in by the agent. Can be overwritten by Ticket::Frontend::NeedAccountedTime.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'NoteMandatory'} = '1';
```

### **Ticket::Frontend::AgentTicketResponsible###Subject**

Sets the default subject for notes added in the ticket responsible screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'Subject'} = '';
```

### **Ticket::Frontend::AgentTicketResponsible###Body**

Sets the default body text for notes added in the ticket responsible screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'Body'} = '';
```

### **Ticket::Frontend::AgentTicketResponsible###InvolvedAgent**

Shows a list of all the involved agents on this ticket, in the ticket responsible screen of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'InvolvedAgent'} = '0';
```

### **Ticket::Frontend::AgentTicketResponsible###InformAgent**

Shows a list of all the possible agents (all agents with note permissions on the queue/ticket) to determine who should be informed about this note, in the ticket responsible screen of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'InformAgent'} = '0';
```

### **Ticket::Frontend::AgentTicketResponsible###ArticleTypeDefault**

Defines the default type of the note in the ticket responsible screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'ArticleTypeDefault'} = 'note-internal';
```

### **Ticket::Frontend::AgentTicketResponsible###ArticleTypes**

Specifies the available note types for this ticket mask. If the option is deselected, ArticleTypeDefault is used and the option is removed from the mask.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'ArticleTypes'} = {  
  'note-external' => '0',  
  'note-internal' => '1',  
  'note-report' => '0'  
};
```

### **Ticket::Frontend::AgentTicketResponsible###Priority**

Shows the ticket priority options in the ticket responsible screen of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'Priority'} = '0';
```

### **Ticket::Frontend::AgentTicketResponsible###PriorityDefault**

Defines the default ticket priority in the ticket responsible screen of the agent interface.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'PriorityDefault'} = '3 normal';
```

### **Ticket::Frontend::AgentTicketResponsible###Title**

Shows the title fields in the ticket responsible screen of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'Title'} = '1';
```

### **Ticket::Frontend::AgentTicketResponsible###HistoryType**

Defines the history type for the ticket responsible screen action, which gets used for ticket history in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'HistoryType'} = 'AddNote';
```

### **Ticket::Frontend::AgentTicketResponsible###HistoryComment**

Defines the history comment for the ticket responsible screen action, which gets used for ticket history in the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'HistoryComment'} = '%Responsible';
```

### **Ticket::Frontend::AgentTicketResponsible###DynamicField**

Dynamic fields shown in the ticket responsible screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'DynamicField'} = {};
```

### **Ticket::Frontend::AgentTicketResponsible###RichTextWidth**

Defines the width for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'RichTextWidth'} = '620';
```

### **Ticket::Frontend::AgentTicketResponsible###RichTextHeight**

Defines the height for the rich text editor component for this screen. Enter number (pixels) or percent value (relative).

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'RichTextHeight'} = '100';
```

### **Ticket::Frontend::AgentTicketResponsibleView###DefaultColumns**

Columns that can be filtered in the responsible view of the agent interface. Possible settings: 0 = Disabled, 1 = Available, 2 = Enabled by default. Note: Only Ticket attributes, Dynamic Fields (DynamicField\_NameX) and Customer attributes (e.g. CustomerUserPhone, CustomerCompanyName, ...) are allowed.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketResponsibleView'}->{'DefaultColumns'} = {
  'Age' => '2',
  'Changed' => '1',
  'Created' => '1',
  'CustomerCompanyName' => '1',
  'CustomerID' => '2',
  'CustomerName' => '1',
  'CustomerUserID' => '1',
  'EscalationResponseTime' => '1',
  'EscalationSolutionTime' => '1',
  'EscalationTime' => '1',
  'EscalationUpdateTime' => '1',
  'Lock' => '2',
  'Owner' => '2',
  'PendingTime' => '1',
  'Priority' => '1',
  'Queue' => '2',
  'Responsible' => '1',
  'SLA' => '1',
  'Service' => '1',
  'State' => '2',
  'TicketNumber' => '2',
```

```
'Title' => '2',  
'Type' => '1'  
};
```

## Ticket → Frontend::Agent::Ticket::ViewSearch

### Ticket::Frontend::AgentTicketSearch###ExtendedSearchCondition

Allows extended search conditions in ticket search of the agent interface. With this feature you can search e. g. ticket title with this kind of conditions like "(\*key1\*&&\*key2\*)" or "(\*key1\*||\*key2\*)".

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'ExtendedSearchCondition'} = '1';
```

### Ticket::Frontend::AgentTicketSearch###SearchLimit

Maximum number of tickets to be displayed in the result of a search in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'SearchLimit'} = '2000';
```

### Ticket::Frontend::AgentTicketSearch###SearchPageShown

Number of tickets to be displayed in each page of a search result in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'SearchPageShown'} = '40';
```

### Ticket::Frontend::AgentTicketSearch###SearchViewableTicketLines

Number of lines (per ticket) that are shown by the search utility in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'SearchViewableTicketLines'} = '10';
```

### Ticket::Frontend::AgentTicketSearch###SortBy::Default

Defines the default ticket attribute for ticket sorting of the ticket search result of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'SortBy::Default'} = 'Age';
```

### Ticket::Frontend::AgentTicketSearch###Order::Default

Defines the default ticket order in the ticket search result of the agent interface. Up: oldest on top. Down: latest on top.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Order::Default'} = 'Down';
```

### **Ticket::Frontend::AgentTicketSearch###SearchArticleCSVTree**

Exports the whole article tree in search result (it can affect the system performance).

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'SearchArticleCSVTree'} = '0';
```

### **Ticket::Frontend::AgentTicketSearch###SearchCSVData**

Data used to export the search result in CSV format.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'SearchCSVData'} = [  
  'TicketNumber',  
  'Age',  
  'Created',  
  'Closed',  
  'FirstLock',  
  'FirstResponse',  
  'State',  
  'Priority',  
  'Queue',  
  'Lock',  
  'Owner',  
  'UserFirstname',  
  'UserLastname',  
  'CustomerID',  
  'CustomerName',  
  'From',  
  'Subject',  
  'AccountedTime',  
  'ArticleTree',  
  'SolutionInMin',  
  'SolutionDiffInMin',  
  'FirstResponseInMin',  
  'FirstResponseDiffInMin'  
];
```

### **Ticket::Frontend::AgentTicketSearch###ArticleCreateTime**

Includes article create times in the ticket search of the agent interface.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'ArticleCreateTime'} = '0';
```

### **Ticket::Frontend::AgentTicketSearch###Defaults###Fulltext**

Defines the default shown ticket search attribute for ticket search screen.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'Fulltext'} = '';
```

### **Ticket::Frontend::AgentTicketSearch###Defaults###TicketNumber**

Defines the default shown ticket search attribute for ticket search screen.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'TicketNumber'} = '';
```

### **Ticket::Frontend::AgentTicketSearch###Defaults###Title**

Defines the default shown ticket search attribute for ticket search screen.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'Title'} = '';
```

### **Ticket::Frontend::AgentTicketSearch###Defaults###From**

Defines the default shown ticket search attribute for ticket search screen.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'From'} = '';
```

### **Ticket::Frontend::AgentTicketSearch###Defaults###To**

Defines the default shown ticket search attribute for ticket search screen.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'To'} = '';
```

### **Ticket::Frontend::AgentTicketSearch###Defaults###Cc**

Defines the default shown ticket search attribute for ticket search screen.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'Cc'} = '';
```

### **Ticket::Frontend::AgentTicketSearch###Defaults###Subject**

Defines the default shown ticket search attribute for ticket search screen.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'Subject'} = '';
```

### **Ticket::Frontend::AgentTicketSearch###Defaults###Body**

Defines the default shown ticket search attribute for ticket search screen.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'Body'} = '';
```

### **Ticket::Frontend::AgentTicketSearch###Defaults###CustomerID**

Defines the default shown ticket search attribute for ticket search screen.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'CustomerID'} = '';
```

### **Ticket::Frontend::AgentTicketSearch###Defaults###CustomerUserLogin**

Defines the default shown ticket search attribute for ticket search screen.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'CustomerUserLogin'} = '';
```

### **Ticket::Frontend::AgentTicketSearch###Defaults###StateIDs**

Defines the default shown ticket search attribute for ticket search screen.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'StateIDs'} = [];
```

### **Ticket::Frontend::AgentTicketSearch###Defaults###QueueIDs**

Defines the default shown ticket search attribute for ticket search screen.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'QueueIDs'} = [];
```

### **Ticket::Frontend::AgentTicketSearch###Defaults###TicketCreateTimePoint**

Default data to use on attribute for ticket search screen. Example: "TicketCreateTimePointFormat=year;TicketCreateTimePointStart=Last;TicketCreateTimePoint=2;"

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'TicketCreateTimePoint'} = '';
```

### **Ticket::Frontend::AgentTicketSearch###Defaults###TicketCreateTimeSlot**

Default data to use on attribute for ticket search screen. Example: "TicketCreateTimeStartYear=2010;TicketCreateTimeStartMonth=10;TicketCreateTimeStartDay=4;T

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'TicketCreateTimeSlot'} = '';
```

### **Ticket::Frontend::AgentTicketSearch###Defaults###TicketChangeTimePoint**

Defines the default shown ticket search attribute for ticket search screen.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'TicketChangeTimePoint'} = '';
```

### **Ticket::Frontend::AgentTicketSearch###Defaults###TicketChangeTimeSlot**

Defines the default shown ticket search attribute for ticket search screen.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'TicketChangeTimeSlot'} = '';
```

### **Ticket::Frontend::AgentTicketSearch###Defaults###TicketCloseTimePoint**

Defines the default shown ticket search attribute for ticket search screen.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'TicketCloseTimePoint'} = '';
```

### **Ticket::Frontend::AgentTicketSearch###Defaults###TicketCloseTimeSlot**

Defines the default shown ticket search attribute for ticket search screen.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'TicketCloseTimeSlot'} = '';
```

### **Ticket::Frontend::AgentTicketSearch###Defaults###TicketEscalationTimePoint**

Defines the default shown ticket search attribute for ticket search screen.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'TicketEscalationTimePoint'} = '';
```

### **Ticket::Frontend::AgentTicketSearch###Defaults###TicketEscalationTimeSlot**

Defines the default shown ticket search attribute for ticket search screen.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'TicketEscalationTimeSlot'} = '';
```

### **Ticket::Frontend::AgentTicketSearch###Defaults###ArticleCreateTimePoint**

Defines the default shown ticket search attribute for ticket search screen.

这个设置默认没有激活。

默认值:



```
$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'ArticleCreateTimePoint'} = '';
```

### **Ticket::Frontend::AgentTicketSearch###Defaults###ArticleCreateTimeSlot**

Defines the default shown ticket search attribute for ticket search screen.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'ArticleCreateTimeSlot'} = '';
```

### **Ticket::Frontend::AgentTicketSearch###Defaults###SearchInArchive**

Defines the default shown ticket search attribute for ticket search screen (AllTickets/ArchivedTickets/NotArchivedTickets).

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'SearchInArchive'} = '';
```

### **Ticket::Frontend::AgentTicketSearch###DynamicField**

Dynamic fields shown in the ticket search screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and shown by default.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'DynamicField'} = {};
```

### **Ticket::Frontend::AgentTicketSearch###Defaults###DynamicField**

Defines the default shown ticket search attribute for ticket search screen. Example: "Key" must have the name of the Dynamic Field in this case 'X', "Content" must have the value of the Dynamic Field depending on the Dynamic Field type, Text: 'a text', Dropdown: '1', Date/Time: 'Search\_DynamicField\_XTimeSlotStartYear=1974; Search\_DynamicField\_XTimeSlotStartMonth=01; Search\_DynamicField\_XTimeSlotStartDay=26; Search\_DynamicField\_XTimeSlotStartHour=00; Search\_DynamicField\_XTimeSlotStartMinute=00; Search\_DynamicField\_XTimeSlotStartSecond=00; Search\_DynamicField\_XTimeSlotStopYear=2013; Search\_DynamicField\_XTimeSlotStopMonth=01; Search\_DynamicField\_XTimeSlotStopDay=26; Search\_DynamicField\_XTimeSlotStopHour=23; Search\_DynamicField\_XTimeSlotStopMinute=59; Search\_DynamicField\_XTimeSlotStopSecond=59;' and or 'Search\_DynamicField\_XTimePointFormat=week; Search\_DynamicField\_XTimePointStart=Before; Search\_DynamicField\_XTimePointValue=7';

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'DynamicField'} = {};
```

### **Ticket::Frontend::AgentTicketSearch###SearchCSVDynamicField**

Dynamic Fields used to export the search result in CSV format.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'SearchCSVDynamicField'} = {};
```

### **Ticket::Frontend::AgentTicketSearch###DefaultColumns**

Columns that can be filtered in the ticket search result view of the agent interface. Possible settings: 0 = Disabled, 1 = Available, 2 = Enabled by default. Note: Only Ticket attributes, Dynamic Fields (DynamicField\_NameX) and Customer attributes (e.g. CustomerUserPhone, CustomerCompanyName, ...) are allowed.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'DefaultColumns'} = {
  'Age' => '2',
  'Changed' => '1',
  'Created' => '1',
  'CustomerCompanyName' => '1',
  'CustomerID' => '2',
  'CustomerName' => '1',
  'CustomerUserID' => '1',
  'EscalationResponseTime' => '1',
  'EscalationSolutionTime' => '1',
  'EscalationTime' => '1',
  'EscalationUpdateTime' => '1',
  'Lock' => '2',
  'Owner' => '2',
  'PendingTime' => '1',
  'Priority' => '1',
  'Queue' => '2',
  'Responsible' => '1',
  'SLA' => '1',
  'Service' => '1',
  'State' => '2',
  'TicketNumber' => '2',
  'Title' => '2',
  'Type' => '1'
};
```

## **Ticket → Frontend::Agent::Ticket::ViewService**

### **Ticket::Frontend::AgentTicketService###StripEmptyLines**

Strips empty lines on the ticket preview in the service view.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketService'}->{'StripEmptyLines'} = '0';
```

### **Ticket::Frontend::AgentTicketService###ViewAllPossibleTickets**

Shows all both ro and rw tickets in the service view.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketService'}->{'ViewAllPossibleTickets'} = '0';
```

### **Ticket::Frontend::AgentTicketService###ServiceSort**

Sorts the tickets (ascendingly or descendingly) when a single queue is selected in the service view and after the tickets are sorted by priority. Values: 0 = ascending (oldest on top, default), 1 = descending (youngest on top). Use the ServiceID for the key and 0 or 1 for value.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketService'}->{'ServiceSort'} = {  
  '3' => '0',  
  '7' => '1'  
};
```

### **Ticket::Frontend::AgentTicketService###SortBy::Default**

Defines the default sort criteria for all services displayed in the service view.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketService'}->{'SortBy::Default'} = 'Age';
```

### **Ticket::Frontend::AgentTicketService###PreSort::ByPriority**

Defines if a pre-sorting by priority should be done in the service view.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketService'}->{'PreSort::ByPriority'} = '1';
```

### **Ticket::Frontend::AgentTicketService###Order::Default**

Defines the default sort order for all services in the service view, after priority sort.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketService'}->{'Order::Default'} = 'Up';
```

### **Ticket::Frontend::AgentTicketService###DefaultColumns**

Columns that can be filtered in the service view of the agent interface. Possible settings: 0 = Disabled, 1 = Available, 2 = Enabled by default. Note: Only Ticket attributes, Dynamic Fields (DynamicField\_NameX) and Customer attributes (e.g. CustomerUserPhone, CustomerCompanyName, ...) are allowed.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketService'}->{'DefaultColumns'} = {  
  'Age' => '2',  
  'Changed' => '1',  
  'Created' => '1',  
  'CustomerCompanyName' => '1',  
  'CustomerID' => '2',  
  'CustomerName' => '1',  
  'CustomerUserID' => '1',  
  'EscalationResponseTime' => '1',  
  'EscalationSolutionTime' => '1',  
  'EscalationTime' => '1',  
  'EscalationUpdateTime' => '1',  
  'Lock' => '2',  
  'Owner' => '2',  
  'PendingTime' => '1',  
  'Priority' => '1',  
  'Queue' => '2',  
  'Responsible' => '1',  
  'SLA' => '1',  
  'Service' => '2',
```

```
'State' => '2',
'TicketNumber' => '2',
'Title' => '2',
'Type' => '1'
};
```

## Ticket → Frontend::Agent::Ticket::ViewStatus

### Ticket::Frontend::AgentTicketStatusView###ViewableTicketsPage

Shows all open tickets (even if they are locked) in the status view of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketStatusView'}->{'ViewableTicketsPage'} = '50';
```

### Ticket::Frontend::AgentTicketStatusView###SortBy::Default

Defines the default ticket attribute for ticket sorting in the status view of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketStatusView'}->{'SortBy::Default'} = 'Age';
```

### Ticket::Frontend::AgentTicketStatusView###Order::Default

Defines the default ticket order (after priority sort) in the status view of the agent interface. Up: oldest on top. Down: latest on top.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketStatusView'}->{'Order::Default'} = 'Down';
```

### Ticket::Frontend::AgentTicketStatusView###DefaultColumns

Columns that can be filtered in the status view of the agent interface. Possible settings: 0 = Disabled, 1 = Available, 2 = Enabled by default. Note: Only Ticket attributes, Dynamic Fields (DynamicField\_NameX) and Customer attributes (e.g. CustomerUserPhone, CustomerCompanyName, ...) are allowed.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketStatusView'}->{'DefaultColumns'} = {
  'Age' => '2',
  'Changed' => '1',
  'Created' => '1',
  'CustomerCompanyName' => '1',
  'CustomerID' => '2',
  'CustomerName' => '1',
  'CustomerUserID' => '1',
  'EscalationResponseTime' => '1',
  'EscalationSolutionTime' => '1',
  'EscalationTime' => '1',
  'EscalationUpdateTime' => '1',
  'Lock' => '2',
  'Owner' => '2',
  'PendingTime' => '1',
  'Priority' => '1',
```

```
'Queue' => '2',
'Responsible' => '1',
'SLA' => '1',
'Service' => '1',
'State' => '2',
'TicketNumber' => '2',
'Title' => '2',
'Type' => '1'
};
```

## Ticket → Frontend::Agent::Ticket::ViewWatch

### Ticket::Frontend::AgentTicketWatchView###SortBy::Default

Defines the default ticket attribute for ticket sorting in the watch view of the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketWatchView'}->{'SortBy::Default'} = 'Age';
```

### Ticket::Frontend::AgentTicketWatchView###Order::Default

Defines the default ticket order in the watch view of the agent interface. Up: oldest on top. Down: latest on top.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketWatchView'}->{'Order::Default'} = 'Up';
```

### Ticket::Frontend::AgentTicketWatchView###DefaultColumns

Columns that can be filtered in the watch view of the agent interface. Possible settings: 0 = Disabled, 1 = Available, 2 = Enabled by default. Note: Only Ticket attributes, Dynamic Fields (DynamicField\_NameX) and Customer attributes (e.g. CustomerUserPhone, CustomerCompanyName, ...) are allowed.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketWatchView'}->{'DefaultColumns'} = {
  'Age' => '2',
  'Changed' => '1',
  'Created' => '1',
  'CustomerCompanyName' => '1',
  'CustomerID' => '2',
  'CustomerName' => '1',
  'CustomerUserID' => '1',
  'EscalationResponseTime' => '1',
  'EscalationSolutionTime' => '1',
  'EscalationTime' => '1',
  'EscalationUpdateTime' => '1',
  'Lock' => '2',
  'Owner' => '2',
  'PendingTime' => '1',
  'Priority' => '1',
  'Queue' => '2',
  'Responsible' => '1',
  'SLA' => '1',
  'Service' => '1',
  'State' => '2',
  'TicketNumber' => '2',
  'Title' => '2',
  'Type' => '1'
};
```

## Ticket → Frontend::Agent::Ticket::ViewZoom

### Ticket::Frontend::PlainView

Shows a link to see a zoomed email ticket in plain text.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::PlainView'} = '0';
```

### Ticket::Frontend::ZoomExpand

Shows all the articles of the ticket (expanded) in the zoom view.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::ZoomExpand'} = '0';
```

### Ticket::Frontend::ZoomExpandSort

Shows the articles sorted normally or in reverse, under ticket zoom in the agent interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::ZoomExpandSort'} = 'reverse';
```

### Ticket::ZoomAttachmentDisplayCount

Shows a count of icons in the ticket zoom, if the article has attachments.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::ZoomAttachmentDisplayCount'} = '20';
```

### Ticket::ZoomTimeDisplay

Displays the accounted time for an article in the ticket zoom view.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::ZoomTimeDisplay'} = '0';
```

### Ticket::UseArticleColors

Shows colors for different article types in the article table.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::UseArticleColors'} = '1';
```

### Ticket::Frontend::TicketArticleFilter

Activates the article filter in the zoom view to specify which articles should be shown.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::TicketArticleFilter'} = '0';
```

### **Ticket::Frontend::HTMLArticleHeightDefault**

Set the default height (in pixels) of inline HTML articles in AgentTicketZoom.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::HTMLArticleHeightDefault'} = '100';
```

### **Ticket::Frontend::HTMLArticleHeightMax**

Set the maximum height (in pixels) of inline HTML articles in AgentTicketZoom.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::HTMLArticleHeightMax'} = '2500';
```

### **Ticket::Frontend::MaxArticlesZoomExpand**

The maximal number of articles expanded on a single page in AgentTicketZoom.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::MaxArticlesZoomExpand'} = '400';
```

### **Ticket::Frontend::MaxArticlesPerPage**

The maximal number of articles shown on a single page in AgentTicketZoom.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::MaxArticlesPerPage'} = '1000';
```

### **Ticket::Frontend::ZoomRichTextForce**

Show article as rich text even if rich text writing is disabled.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::ZoomRichTextForce'} = '1';
```

### **Ticket::Frontend::AgentTicketZoom###DynamicField**

Dynamic fields shown in the sidebar of the ticket zoom screen of the agent interface.  
Possible settings: 0 = Disabled, 1 = Enabled.

默认值:

```
$Self->{'Ticket::Frontend::AgentTicketZoom'}->{'DynamicField'} = {};
```

## Ticket::Frontend::ZoomCollectMeta

Whether or not to collect meta information from articles using filters configured in Ticket::Frontend::ZoomCollectMetaFilters.

默认值:

```
$Self->{'Ticket::Frontend::ZoomCollectMeta'} = '0';
```

## Ticket::Frontend::ZoomCollectMetaFilters###CVE-Mitre

Defines a filter to collect CVE numbers from article texts in AgentTicketZoom. The results will be displayed in a meta box next to the article. Fill in URLPreview if you would like to see a preview when moving your mouse cursor above the link element. This could be the same URL as in URL, but also an alternate one. Please note that some websites deny being displayed within an iframe (e.g. Google) and thus won't work with the preview mode.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::ZoomCollectMetaFilters'}->{'CVE-Mitre'} = {
  'Meta' => {
    'Name' => 'CVE Mitre',
    'Target' => '_blank',
    'URL' => 'http://cve.mitre.org/cgi-bin/cvename.cgi?name=<MATCH1>-<MATCH2>-<MATCH3>',
    'URLPreview' => 'http://cve.mitre.org/cgi-bin/cvename.cgi?name=<MATCH1>-<MATCH2>-<MATCH3>'
  },
  'RegExp' => [
    '(CVE|CAN)\-\(\d{3,4})\-\(\d{2,})'
  ]
};
```

## Ticket::Frontend::ZoomCollectMetaFilters###CVE-Google

Defines a filter to collect CVE numbers from article texts in AgentTicketZoom. The results will be displayed in a meta box next to the article. Fill in URLPreview if you would like to see a preview when moving your mouse cursor above the link element. This could be the same URL as in URL, but also an alternate one. Please note that some websites deny being displayed within an iframe (e.g. Google) and thus won't work with the preview mode.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::ZoomCollectMetaFilters'}->{'CVE-Google'} = {
  'Meta' => {
    'Name' => 'CVE Google Search',
    'Target' => '_blank',
    'URL' => 'http://google.com/search?q=<MATCH1>-<MATCH2>-<MATCH3>',
    'URLPreview' => ''
  },
  'RegExp' => [
    '(CVE|CAN)\-\(\d{3,4})\-\(\d{2,})'
  ]
};
```

## Ticket → Frontend::Agent::TicketOverview

### Ticket::Frontend::Overview###Small

Allows having a small format ticket overview (CustomerInfo => 1 - shows also the customer information).



默认值:

```
$Self->{'Ticket::Frontend::Overview'}->{'Small'} = {
  'CustomerInfo' => '1',
  'Module' => 'Kernel::Output::HTML::TicketOverview::Small',
  'ModulePriority' => '100',
  'Name' => 'Small',
  'NameShort' => 'S'
};
```

### **Ticket::Frontend::OverviewSmall###ColumnHeader**

Shows either the last customer article's subject or the ticket title in the small format overview.

默认值:

```
$Self->{'Ticket::Frontend::OverviewSmall'}->{'ColumnHeader'} = 'LastCustomerSubject';
```

### **Ticket::Frontend::Overview###Medium**

Allows having a medium format ticket overview (CustomerInfo => 1 - shows also the customer information).

默认值:

```
$Self->{'Ticket::Frontend::Overview'}->{'Medium'} = {
  'CustomerInfo' => '0',
  'Module' => 'Kernel::Output::HTML::TicketOverview::Medium',
  'ModulePriority' => '200',
  'Name' => 'Medium',
  'NameShort' => 'M',
  'OverviewMenuModules' => '1',
  'TicketActionsPerTicket' => '1'
};
```

### **Ticket::Frontend::Overview###Preview**

Shows a preview of the ticket overview (CustomerInfo => 1 - shows also Customer-Info, CustomerInfoMaxSize max. size in characters of Customer-Info).

默认值:

```
$Self->{'Ticket::Frontend::Overview'}->{'Preview'} = {
  'CustomerInfo' => '0',
  'CustomerInfoMaxSize' => '18',
  'DefaultPreViewLines' => '25',
  'DefaultViewNewLine' => '90',
  'Module' => 'Kernel::Output::HTML::TicketOverview::Preview',
  'ModulePriority' => '300',
  'Name' => 'Large',
  'NameShort' => 'L',
  'OverviewMenuModules' => '1',
  'StripEmptyLines' => '0',
  'TicketActionsPerTicket' => '1'
};
```

### **Ticket::Frontend::Overview::PreviewArticleSenderTypes**

Defines which article sender types should be shown in the preview of a ticket.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::Overview::PreviewArticleSenderTypes'} = {
  'agent' => '1',
  'customer' => '1',
  'system' => '1'
};
```

```
};
```

### **Ticket::Frontend::Overview::PreviewArticleLimit**

Sets the count of articles visible in preview mode of ticket overviews.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::Overview::PreviewArticleLimit'} = '5';
```

### **Ticket::Frontend::Overview::PreviewArticleTypeExpanded**

Defines which article type should be expanded when entering the overview. If nothing defined, latest article will be expanded.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::Overview::PreviewArticleTypeExpanded'} = '';
```

### **Ticket::Frontend::OverviewSmall###DynamicField**

Dynamic fields shown in the ticket small format overview screen of the agent interface. Possible settings: 0 = Disabled, 1 = Available, 2 = Enabled by default.

默认值:

```
$Self->{'Ticket::Frontend::OverviewSmall'}->{'DynamicField'} = {};
```

### **Ticket::Frontend::OverviewMedium###DynamicField**

Dynamic fields shown in the ticket medium format overview screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled.

默认值:

```
$Self->{'Ticket::Frontend::OverviewMedium'}->{'DynamicField'} = {};
```

### **Ticket::Frontend::OverviewPreview###DynamicField**

Dynamic fields shown in the ticket preview format overview screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled.

默认值:

```
$Self->{'Ticket::Frontend::OverviewPreview'}->{'DynamicField'} = {};
```

## **Ticket → Frontend::Agent::ToolBarModule**

### **Frontend::ToolBarModule###1-Ticket::AgentTicketQueue**

Toolbar Item for a shortcut. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move\_into:group2".

这个设置默认没有激活。

默认值:

```
$Self->{'Frontend::ToolBarModule'}->{'1-Ticket::AgentTicketQueue'} = {
  'AccessKey' => 'q',
  'Action' => 'AgentTicketQueue',
  'CssClass' => 'QueueView',
  'Icon' => 'fa fa-folder',
  'Link' => 'Action=AgentTicketQueue',
```

```
'Module' => 'Kernel::Output::HTML::ToolBar::Link',
'Name' => 'Queue view',
'Priority' => '1010010'
};
```

### Frontend::ToolBarModule###2-Ticket::AgentTicketStatus

Toolbar Item for a shortcut. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move\_into:group2".

这个设置默认没有激活。

默认值:

```
$Self->{'Frontend::ToolBarModule'}->{'2-Ticket::AgentTicketStatus'} = {
'AccessKey' => 'S',
'Action' => 'AgentTicketStatusView',
'CssClass' => 'StatusView',
'Icon' => 'fa fa-list-ol',
'Link' => 'Action=AgentTicketStatusView',
'Module' => 'Kernel::Output::HTML::ToolBar::Link',
'Name' => 'Status view',
'Priority' => '1010020'
};
```

### Frontend::ToolBarModule###3-Ticket::AgentTicketEscalation

Toolbar Item for a shortcut. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move\_into:group2".

这个设置默认没有激活。

默认值:

```
$Self->{'Frontend::ToolBarModule'}->{'3-Ticket::AgentTicketEscalation'} = {
'AccessKey' => 'w',
'Action' => 'AgentTicketEscalationView',
'CssClass' => 'EscalationView',
'Icon' => 'fa fa-exclamation',
'Link' => 'Action=AgentTicketEscalationView',
'Module' => 'Kernel::Output::HTML::ToolBar::Link',
'Name' => 'Escalation view',
'Priority' => '1010030'
};
```

### Frontend::ToolBarModule###4-Ticket::AgentTicketPhone

Toolbar Item for a shortcut. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move\_into:group2".

这个设置默认没有激活。

默认值:

```
$Self->{'Frontend::ToolBarModule'}->{'4-Ticket::AgentTicketPhone'} = {
'AccessKey' => '',
'Action' => 'AgentTicketPhone',
'CssClass' => 'PhoneTicket',
'Icon' => 'fa fa-phone',
'Link' => 'Action=AgentTicketPhone',
'Module' => 'Kernel::Output::HTML::ToolBar::Link',
'Name' => 'New phone ticket',
'Priority' => '1020010'
};
```

### Frontend::ToolBarModule###5-Ticket::AgentTicketEmail

Toolbar Item for a shortcut. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move\_into:group2".

这个设置默认没有激活。

默认值:

```
$Self->{'Frontend::ToolBarModule'}->{'5-Ticket::AgentTicketEmail'} = {
  'AccessKey' => '',
  'Action' => 'AgentTicketEmail',
  'CssClass' => 'EmailTicket',
  'Icon' => 'fa fa-envelope',
  'Link' => 'Action=AgentTicketEmail',
  'Module' => 'Kernel::Output::HTML::ToolBar::Link',
  'Name' => 'New email ticket',
  'Priority' => '1020020'
};
```

### Frontend::ToolBarModule###6-Ticket::AgentTicketProcess

Toolbar Item for a shortcut. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move\_into:group2".

这个设置默认没有激活。

默认值:

```
$Self->{'Frontend::ToolBarModule'}->{'6-Ticket::AgentTicketProcess'} = {
  'AccessKey' => '',
  'Action' => 'AgentTicketProcess',
  'CssClass' => 'ProcessTicket',
  'Icon' => 'fa fa-th-large',
  'Link' => 'Action=AgentTicketProcess',
  'Module' => 'Kernel::Output::HTML::ToolBar::Link',
  'Name' => 'New process ticket',
  'Priority' => '1020030'
};
```

### Frontend::ToolBarModule###7-Ticket::TicketResponsible

Agent interface notification module to see the number of tickets an agent is responsible for. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move\_into:group2".

默认值:

```
$Self->{'Frontend::ToolBarModule'}->{'7-Ticket::TicketResponsible'} = {
  'AccessKey' => 'r',
  'AccessKeyNew' => '',
  'AccessKeyReached' => '',
  'CssClass' => 'Responsible',
  'CssClassNew' => 'Responsible New',
  'CssClassReached' => 'Responsible Reached',
  'Icon' => 'fa fa-user',
  'IconNew' => 'fa fa-user',
  'IconReached' => 'fa fa-user',
  'Module' => 'Kernel::Output::HTML::ToolBar::TicketResponsible',
  'Priority' => '1030010'
};
```

### Frontend::ToolBarModule###8-Ticket::TicketWatcher

Agent interface notification module to see the number of watched tickets. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move\_into:group2".

默认值:

```
$Self->{'Frontend::ToolBarModule'}->{'8-Ticket::TicketWatcher'} = {
  'AccessKey' => '',
  'AccessKeyNew' => '',
```

```
'AccessKeyReached' => '',
'CssClass' => 'Watcher',
'CssClassNew' => 'Watcher New',
'CssClassReached' => 'Watcher Reached',
'Icon' => 'fa fa-eye',
'IconNew' => 'fa fa-eye',
'IconReached' => 'fa fa-eye',
'Module' => 'Kernel::Output::HTML::ToolBar::TicketWatcher',
'Priority' => '1030020'
};
```

### Frontend::ToolBarModule###9-Ticket::TicketLocked

Agent interface notification module to see the number of locked tickets. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move\_into:group2".

默认值:

```
$Self->{'Frontend::ToolBarModule'}->{'9-Ticket::TicketLocked'} = {
'AccessKey' => 'k',
'AccessKeyNew' => '',
'AccessKeyReached' => '',
'CssClass' => 'Locked',
'CssClassNew' => 'Locked New',
'CssClassReached' => 'Locked Reached',
'Icon' => 'fa fa-lock',
'IconNew' => 'fa fa-lock',
'IconReached' => 'fa fa-lock',
'Module' => 'Kernel::Output::HTML::ToolBar::TicketLocked',
'Priority' => '1030030'
};
```

### Frontend::ToolBarModule###10-Ticket::AgentTicketService

Agent interface notification module to see the number of tickets in My Services. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move\_into:group2".

这个设置默认没有激活。

默认值:

```
$Self->{'Frontend::ToolBarModule'}->{'10-Ticket::AgentTicketService'} = {
'CssClass' => 'ServiceView',
'Icon' => 'fa fa-wrench',
'Module' => 'Kernel::Output::HTML::ToolBar::TicketService',
'Priority' => '1030035'
};
```

### Frontend::ToolBarModule###11-Ticket::TicketSearchProfile

Agent interface module to access search profiles via nav bar. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move\_into:group2".

这个设置默认没有激活。

默认值:

```
$Self->{'Frontend::ToolBarModule'}->{'11-Ticket::TicketSearchProfile'} = {
'Block' => 'ToolBarSearchProfile',
'Description' => 'Search template',
'MaxWidth' => '40',
'Module' => 'Kernel::Output::HTML::ToolBar::TicketSearchProfile',
'Name' => 'Search template',
'Priority' => '1990010'
};
```

### Frontend::ToolBarModule###12-Ticket::TicketSearchFulltext

Agent interface module to access fulltext search via nav bar. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move\_into:group2".

这个设置默认没有激活。

默认值:

```
$Self->{'Frontend::ToolBarModule'}->{'12-Ticket::TicketSearchFulltext'} = {
  'Block' => 'ToolBarSearchFulltext',
  'CSS' => 'Core.Agent.Toolbar.FulltextSearch.css',
  'Description' => 'Fulltext search',
  'Module' => 'Kernel::Output::HTML::ToolBar::Generic',
  'Name' => 'Fulltext search',
  'Priority' => '1990020',
  'Size' => '10'
};
```

### Frontend::ToolBarModule###13-CICSearchCustomerID

Agent interface module to access CIC search via nav bar. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move\_into:group2".

这个设置默认没有激活。

默认值:

```
$Self->{'Frontend::ToolBarModule'}->{'13-CICSearchCustomerID'} = {
  'Block' => 'ToolBarCICSearchCustomerID',
  'CSS' => 'Core.Agent.Toolbar.CICSearch.css',
  'Description' => 'CustomerID search',
  'Module' => 'Kernel::Output::HTML::ToolBar::Generic',
  'Name' => 'CustomerID search',
  'Priority' => '1990030',
  'Size' => '10'
};
```

### Frontend::ToolBarModule###14-CICSearchCustomerUser

Agent interface module to access CIC search via nav bar. Additional access control to show or not show this link can be done by using Key "Group" and Content like "rw:group1;move\_into:group2".

这个设置默认没有激活。

默认值:

```
$Self->{'Frontend::ToolBarModule'}->{'14-CICSearchCustomerUser'} = {
  'Block' => 'ToolBarCICSearchCustomerUser',
  'CSS' => 'Core.Agent.Toolbar.CICSearch.css',
  'Description' => 'Customer user search',
  'Module' => 'Kernel::Output::HTML::ToolBar::Generic',
  'Name' => 'Customer user search',
  'Priority' => '1990040',
  'Size' => '10'
};
```

## Ticket → Frontend::Customer

### Ticket::Frontend::CustomerDisableCompanyTicketAccess

This option will deny the access to customer company tickets, which are not created by the customer user.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::CustomerDisableCompanyTicketAccess'} = '0';
```

### **Ticket::Frontend::CustomerTicketOverviewCustomEmptyText**

Custom text for the page shown to customers that have no tickets yet (if you need those text translated add them to a custom translation module).

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketOverviewCustomEmptyText'} = {
  'Button' => 'Create your first ticket',
  'Text' => 'Please click the button below to create your first ticket.',
  'Title' => 'Welcome!'
};
```

### **Frontend::CustomerUser::Item###15-OpenTickets**

Customer item (icon) which shows the open tickets of this customer as info block. Setting CustomerUserLogin to 1 searches for tickets based on login name rather than CustomerID.

默认值:

```
$Self->{'Frontend::CustomerUser::Item'}->{'15-OpenTickets'} = {
  'Action' => 'AgentTicketSearch',
  'Attributes' => 'StateType=Open;',
  'CSS' => 'Core.Agent.CustomerUser.OpenTicket.css',
  'CSSClassNoOpenTicket' => 'NoOpenTicket',
  'CSSClassOpenTicket' => 'OpenTicket',
  'CustomerUserLogin' => '0',
  'IconNameNoOpenTicket' => 'fa-check-circle',
  'IconNameOpenTicket' => 'fa-exclamation-circle',
  'Module' => 'Kernel::Output::HTML::CustomerUser::GenericTicket',
  'Subaction' => 'Search',
  'Target' => '_blank',
  'Text' => 'Open tickets (customer)'
};
```

### **Frontend::CustomerUser::Item###16-OpenTicketsForCustomerUserLogin**

Customer item (icon) which shows the open tickets of this customer as info block. Setting CustomerUserLogin to 1 searches for tickets based on login name rather than CustomerID.

这个设置默认没有激活。

默认值:

```
$Self->{'Frontend::CustomerUser::Item'}->{'16-OpenTicketsForCustomerUserLogin'} = {
  'Action' => 'AgentTicketSearch',
  'Attributes' => 'StateType=Open;',
  'CSS' => 'Core.Agent.CustomerUser.OpenTicket.css',
  'CSSClassNoOpenTicket' => 'NoOpenTicket',
  'CSSClassOpenTicket' => 'OpenTicket',
  'CustomerUserLogin' => '1',
  'IconNameNoOpenTicket' => 'fa-check-circle',
  'IconNameOpenTicket' => 'fa-exclamation-circle',
  'Module' => 'Kernel::Output::HTML::CustomerUser::GenericTicket',
  'Subaction' => 'Search',
  'Target' => '_blank',
  'Text' => 'Open tickets (customer user)'
};
```

### Frontend::CustomerUser::Item###17-ClosedTickets

Customer item (icon) which shows the closed tickets of this customer as info block. Setting CustomerUserLogin to 1 searches for tickets based on login name rather than CustomerID.

这个设置默认没有激活。

默认值:

```
$Self->{'Frontend::CustomerUser::Item'}->{'17-ClosedTickets'} = {
  'Action' => 'AgentTicketSearch',
  'Attributes' => 'StateType=Closed;',
  'CSS' => 'Core.Agent.CustomerUser.OpenTicket.css',
  'CSSClassNoOpenTicket' => 'NoOpenTicket',
  'CSSClassOpenTicket' => 'OpenTicket',
  'CustomerUserLogin' => '0',
  'IconNameNoOpenTicket' => 'fa-power-off',
  'IconNameOpenTicket' => 'fa-power-off',
  'Module' => 'Kernel::Output::HTML::CustomerUser::GenericTicket',
  'Subaction' => 'Search',
  'Target' => '_blank',
  'Text' => 'Closed tickets (customer)';
};
```

### Frontend::CustomerUser::Item###18-ClosedTicketsForCustomerUserLogin

Customer item (icon) which shows the closed tickets of this customer as info block. Setting CustomerUserLogin to 1 searches for tickets based on login name rather than CustomerID.

这个设置默认没有激活。

默认值:

```
$Self->{'Frontend::CustomerUser::Item'}->{'18-ClosedTicketsForCustomerUserLogin'} = {
  'Action' => 'AgentTicketSearch',
  'Attributes' => 'StateType=Closed;',
  'CSS' => 'Core.Agent.CustomerUser.OpenTicket.css',
  'CSSClassNoOpenTicket' => 'NoOpenTicket',
  'CSSClassOpenTicket' => 'OpenTicket',
  'CustomerUserLogin' => '1',
  'IconNameNoOpenTicket' => 'fa-power-off',
  'IconNameOpenTicket' => 'fa-power-off',
  'Module' => 'Kernel::Output::HTML::CustomerUser::GenericTicket',
  'Subaction' => 'Search',
  'Target' => '_blank',
  'Text' => 'Closed tickets (customer user)';
};
```

### CustomerFrontend::CommonParam###Action

Defines the default used Frontend-Module if no Action parameter given in the url on the customer interface.

不能使这个设置失效。

默认值:

```
$Self->{'CustomerFrontend::CommonParam'}->{'Action'} = 'CustomerTicketOverview';
```

### CustomerFrontend::CommonParam###TicketID

Default ticket ID used by the system in the customer interface.

不能使这个设置失效。

默认值:



```
$Self->{'CustomerFrontend::CommonParam'}->{'TicketID'} = '';
```

## Ticket → Frontend::Customer::ModuleMetaHead

### CustomerFrontend::HeaderMetaModule###2-TicketSearch

Module to generate html OpenSearch profile for short ticket search in the customer interface.

默认值:

```
$Self->{'CustomerFrontend::HeaderMetaModule'}->{'2-TicketSearch'} = {
  'Action' => 'CustomerTicketSearch',
  'Module' => 'Kernel::Output::HTML::HeaderMeta::CustomerTicketSearch'
};
```

## Ticket → Frontend::Customer::ModuleRegistration

### CustomerFrontend::Module###CustomerTicketOverview

Frontend module registration for the customer interface.

默认值:

```
$Self->{'CustomerFrontend::Module'}->{'CustomerTicketOverview'} = {
  'Description' => 'Overview of customer tickets.',
  'NavBar' => [
    {
      'AccessKey' => 'm',
      'Block' => '',
      'Description' => 'Tickets.',
      'Link' => 'Action=CustomerTicketOverview;Subaction=MyTickets',
      'LinkOption' => '',
      'Name' => 'Tickets',
      'NavBar' => 'Ticket',
      'Prio' => '100',
      'Type' => 'Menu'
    },
    {
      'AccessKey' => '',
      'Block' => '',
      'Description' => 'My Tickets.',
      'Link' => 'Action=CustomerTicketOverview;Subaction=MyTickets',
      'LinkOption' => '',
      'Name' => 'My Tickets',
      'NavBar' => 'Ticket',
      'Prio' => '110',
      'Type' => 'Submenu'
    },
    {
      'AccessKey' => 'M',
      'Block' => '',
      'Description' => 'Company Tickets.',
      'Link' => 'Action=CustomerTicketOverview;Subaction=CompanyTickets',
      'LinkOption' => '',
      'Name' => 'Company Tickets',
      'NavBar' => 'Ticket',
      'Prio' => '120',
      'Type' => 'Submenu'
    }
  ],
  'NavBarName' => 'Ticket',
  'Title' => 'Overview'
};
```

### CustomerFrontend::Module###CustomerTicketMessage

Frontend module registration for the customer interface.

默认值:

```
$Self->{'CustomerFrontend::Module'}->{'CustomerTicketMessage'} = {
  'Description' => 'Create tickets.',
  'NavBar' => [
    {
      'AccessKey' => 'n',
      'Block' => '',
      'Description' => 'Create new Ticket.',
      'Link' => 'Action=CustomerTicketMessage',
      'LinkOption' => '',
      'Name' => 'New Ticket',
      'NavBar' => 'Ticket',
      'Prio' => '100',
      'Type' => 'Submenu'
    }
  ],
  'NavBarName' => 'Ticket',
  'Title' => 'New Ticket'
};
```

### CustomerFrontend::Module###CustomerTicketZoom

Frontend module registration for the customer interface.

默认值:

```
$Self->{'CustomerFrontend::Module'}->{'CustomerTicketZoom'} = {
  'Description' => 'Ticket zoom view.',
  'Loader' => {
    'JavaScript' => [
      'Core.Customer.TicketZoom.js',
      'Core.UI.Popup.js'
    ]
  },
  'NavBarName' => 'Ticket',
  'Title' => 'Zoom'
};
```

### CustomerFrontend::Module###CustomerTicketPrint

Frontend module registration for the customer interface.

默认值:

```
$Self->{'CustomerFrontend::Module'}->{'CustomerTicketPrint'} = {
  'Description' => 'Customer Ticket Print Module.',
  'NavBarName' => '',
  'Title' => 'Print'
};
```

### CustomerFrontend::Module###CustomerTicketAttachment

Frontend module registration for the customer interface.

默认值:

```
$Self->{'CustomerFrontend::Module'}->{'CustomerTicketAttachment'} = {
  'Description' => 'To download attachments.',
  'NavBarName' => '',
  'Title' => ''
};
```

### CustomerFrontend::Module###CustomerTicketSearch

Frontend module registration for the customer interface.

默认值:

```
$Self->{'CustomerFrontend::Module'}->{'CustomerTicketSearch'} = {
```

```
'Description' => 'Customer ticket search.',
'NavBar' => [
  {
    'AccessKey' => 's',
    'Block' => '',
    'Description' => 'Search.',
    'Link' => 'Action=CustomerTicketSearch',
    'LinkOption' => '',
    'Name' => 'Search',
    'NavBar' => 'Ticket',
    'Prio' => '300',
    'Type' => 'Submenu'
  }
],
'NavBarName' => 'Ticket',
'Title' => 'Search'
};
```

## Ticket → Frontend::Customer::Preferences

### CustomerPreferencesGroups###ShownTickets

Defines all the parameters for the ShownTickets object in the customer preferences of the customer interface.

默认值:

```
$Self->{'CustomerPreferencesGroups'}->{'ShownTickets'} = {
  'Active' => '1',
  'Column' => 'User Profile',
  'Data' => {
    '15' => '15',
    '20' => '20',
    '25' => '25',
    '30' => '30'
  },
  'DataSelected' => '25',
  'Desc' => 'Select how many tickets should be shown in overviews by default.',
  'Key' => 'Tickets per page',
  'Label' => 'Number of displayed tickets',
  'Module' => 'Kernel::Output::HTML::Preferences::Generic',
  'PrefKey' => 'UserShowTickets',
  'Prio' => '4000'
};
```

### CustomerPreferencesGroups###RefreshTime

Defines all the parameters for the RefreshTime object in the customer preferences of the customer interface.

默认值:

```
$Self->{'CustomerPreferencesGroups'}->{'RefreshTime'} = {
  'Active' => '1',
  'Column' => 'User Profile',
  'Data' => {
    '0' => 'off',
    '10' => '10 minutes',
    '15' => '15 minutes',
    '2' => ' 2 minutes',
    '5' => ' 5 minutes',
    '7' => ' 7 minutes'
  },
  'DataSelected' => '0',
  'Desc' => 'Select after which period ticket overviews should refresh automatically.',
  'Key' => 'Refresh interval',
  'Label' => 'Ticket overview',
  'Module' => 'Kernel::Output::HTML::Preferences::Generic',
  'PrefKey' => 'UserRefreshTime',
  'Prio' => '4000'
};
```

```
};
```

## **Ticket → Frontend::Customer::Ticket::ViewNew**

### **Ticket::Frontend::CustomerTicketMessage###NextScreenAfterNewTicket**

Determines the next screen after new customer ticket in the customer interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'NextScreenAfterNewTicket'} =  
'CustomerTicketOverview';
```

### **Ticket::Frontend::CustomerTicketMessage###Priority**

Allows customers to set the ticket priority in the customer interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'Priority'} = '1';
```

### **Ticket::Frontend::CustomerTicketMessage###PriorityDefault**

Defines the default priority of new customer tickets in the customer interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'PriorityDefault'} = '3 normal';
```

### **Ticket::Frontend::CustomerTicketMessage###Queue**

Allows customers to set the ticket queue in the customer interface. If this is set to 'No', QueueDefault should be configured.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'Queue'} = '1';
```

### **Ticket::Frontend::CustomerTicketMessage###QueueDefault**

Defines the default queue for new customer tickets in the customer interface.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'QueueDefault'} = 'Postmaster';
```

### **Ticket::Frontend::CustomerTicketMessage###TicketType**

Allows customers to set the ticket type in the customer interface. If this is set to 'No', TicketTypeDefault should be configured.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'TicketType'} = '1';
```

### **Ticket::Frontend::CustomerTicketMessage###TicketTypeDefault**

Defines the default ticket type for new customer tickets in the customer interface.

这个设置默认没有激活。

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'TicketTypeDefault'} = 'Unclassified';
```

### **Ticket::Frontend::CustomerTicketMessage###Service**

Allows customers to set the ticket service in the customer interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'Service'} = '1';
```

### **Ticket::Frontend::CustomerTicketMessage###SLA**

Allows customers to set the ticket SLA in the customer interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'SLA'} = '1';
```

### **Ticket::Frontend::CustomerTicketMessage###ServiceMandatory**

Sets if service must be selected by the customer.

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'ServiceMandatory'} = '0';
```

### **Ticket::Frontend::CustomerTicketMessage###SLAMandatory**

Sets if SLA must be selected by the customer.

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'SLAMandatory'} = '0';
```

### **Ticket::Frontend::CustomerTicketMessage###StateDefault**

Defines the default state of new customer tickets in the customer interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'StateDefault'} = 'new';
```

### **Ticket::Frontend::CustomerTicketMessage###ArticleType**

Defines the default type for article in the customer interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'ArticleType'} = 'webrequest';
```

### **Ticket::Frontend::CustomerTicketMessage###SenderType**

Sender type for new tickets from the customer interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'SenderType'} = 'customer';
```

### **Ticket::Frontend::CustomerTicketMessage###HistoryType**

Defines the default history type in the customer interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'HistoryType'} =  
'WebRequestCustomer';
```

### **Ticket::Frontend::CustomerTicketMessage###HistoryComment**

Comment for new history entries in the customer interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'HistoryComment'} = '';
```

### **CustomerPanelSelectionType**

Defines the recipient target of the tickets ("Queue" shows all queues, "SystemAddress" shows only the queues which are assigned to system addresses) in the customer interface.

不能使这个设置失效。

默认值:

```
$Self->{'CustomerPanelSelectionType'} = 'Queue';
```

### **CustomerPanelSelectionString**

Determines the strings that will be shown as recipient (To:) of the ticket in the customer interface. For Queue as CustomerPanelSelectionType, "<Queue>" shows the names of the queues, and for SystemAddress, "<Realname> <<Email>>" shows the name and email of the recipient.

默认值:

```
$Self->{'CustomerPanelSelectionString'} = '<Queue>';
```

### **CustomerPanelOwnSelection**

Determines which queues will be valid for ticket's recipients in the customer interface.

这个设置默认没有激活。

默认值:

```
$Self->{'CustomerPanelOwnSelection'} = {  
'Junk' => 'First Queue',
```

```
'Misc' => 'Second Queue'
};
```

### CustomerPanel::NewTicketQueueSelectionModule

Module for To-selection in new ticket screen in the customer interface.

不能使这个设置失效。

默认值:

```
$Self->{'CustomerPanel::NewTicketQueueSelectionModule'} =
'Kernel::Output::HTML::CustomerNewTicket::QueueSelectionGeneric';
```

### Ticket::Frontend::CustomerTicketMessage###DynamicField

Dynamic fields options shown in the ticket message screen of the customer interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required. NOTE. If you want to display these fields also in the ticket zoom of the customer interface, you have to enable them in CustomerTicketZoom###DynamicField.

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'DynamicField'} = {};
```

### Ticket → Frontend::Customer::Ticket::ViewPrint

#### Ticket::Frontend::CustomerTicketPrint###DynamicField

Dynamic fields shown in the ticket print screen of the customer interface. Possible settings: 0 = Disabled, 1 = Enabled.

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketPrint'}->{'DynamicField'} = {};
```

### Ticket → Frontend::Customer::Ticket::ViewSearch

#### Ticket::CustomerTicketSearch::SearchLimit

Maximum number of tickets to be displayed in the result of a search in the customer interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::CustomerTicketSearch::SearchLimit'} = '5000';
```

#### Ticket::CustomerTicketSearch::SearchPageShown

Number of tickets to be displayed in each page of a search result in the customer interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::CustomerTicketSearch::SearchPageShown'} = '40';
```

#### Ticket::CustomerTicketSearch::SortBy::Default

Defines the default ticket attribute for ticket sorting in a ticket search of the customer interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::CustomerTicketSearch::SortBy::Default'} = 'Age';
```

### **Ticket::CustomerTicketSearch::Order::Default**

Defines the default ticket order of a search result in the customer interface. Up: oldest on top. Down: latest on top.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::CustomerTicketSearch::Order::Default'} = 'Down';
```

### **Ticket::Frontend::CustomerTicketSearch###ExtendedSearchCondition**

Allows extended search conditions in ticket search of the customer interface. With this feature you can search e. g. ticket title with this kind of conditions like "(\*key1\*&&\*key2\*)" or "(\*key1\*||\*key2\*)".

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketSearch'}->{'ExtendedSearchCondition'} = '1';
```

### **Customer::TicketSearch::AllServices**

If enabled, the customer can search for tickets in all services (regardless what services are assigned to the customer).

不能使这个设置失效。

默认值:

```
$Self->{'Customer::TicketSearch::AllServices'} = '0';
```

### **Ticket::Frontend::CustomerTicketSearch###SearchArticleCSVTree**

Exports the whole article tree in search result (it can affect the system performance).

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketSearch'}->{'SearchArticleCSVTree'} = '0';
```

### **Ticket::Frontend::CustomerTicketSearch###SearchCSVData**

Data used to export the search result in CSV format.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketSearch'}->{'SearchCSVData'} = [  
  'TicketNumber',  
  'Age',  
  'Created',  
  'Closed',  
  'State',  
  'Priority',
```



```
'Lock',
'CustomerID',
'CustomerName',
'From',
'Subject'
];
```

### **Ticket::Frontend::CustomerTicketSearch###DynamicField**

Dynamic fields shown in the ticket search screen of the customer interface. Possible settings: 0 = Disabled, 1 = Enabled.

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketSearch'}->{'DynamicField'} = {};
```

### **Ticket::Frontend::CustomerTicketSearch###SearchOverviewDynamicField**

Dynamic fields shown in the ticket search overview results screen of the customer interface. Possible settings: 0 = Disabled, 1 = Enabled.

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketSearch'}->{'SearchOverviewDynamicField'} = {};
```

### **Ticket::Frontend::CustomerTicketSearch###SearchCSVDynamicField**

Dynamic Fields used to export the search result in CSV format.

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketSearch'}->{'SearchCSVDynamicField'} = {};
```

## **Ticket → Frontend::Customer::Ticket::ViewZoom**

### **Ticket::Frontend::CustomerTicketZoom###NextScreenAfterFollowUp**

Determines the next screen after the follow-up screen of a zoomed ticket in the customer interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'NextScreenAfterFollowUp'} = 'CustomerTicketOverview';
```

### **Ticket::Frontend::CustomerTicketZoom###ArticleType**

Defines the default type of the note in the ticket zoom screen of the customer interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'ArticleType'} = 'webrequest';
```

### **Ticket::Frontend::CustomerTicketZoom###SenderType**

Defines the default sender type for tickets in the ticket zoom screen of the customer interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'SenderType'} = 'customer';
```

### **Ticket::Frontend::CustomerTicketZoom###HistoryType**

Defines the history type for the ticket zoom action, which gets used for ticket history in the customer interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'HistoryType'} = 'FollowUp';
```

### **Ticket::Frontend::CustomerTicketZoom###HistoryComment**

Defines the history comment for the ticket zoom action, which gets used for ticket history in the customer interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'HistoryComment'} = '';
```

### **Ticket::Frontend::CustomerTicketZoom###Priority**

Allows customers to change the ticket priority in the customer interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'Priority'} = '1';
```

### **Ticket::Frontend::CustomerTicketZoom###PriorityDefault**

Defines the default priority of follow-up customer tickets in the ticket zoom screen in the customer interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'PriorityDefault'} = '3 normal';
```

### **Ticket::Frontend::CustomerTicketZoom###State**

Allows choosing the next compose state for customer tickets in the customer interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'State'} = '1';
```

### **Ticket::Frontend::CustomerTicketZoom###StateDefault**

Defines the default next state for a ticket after customer follow-up in the customer interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'StateDefault'} = 'open';
```

### **Ticket::Frontend::CustomerTicketZoom###StateType**

Defines the next possible states for customer tickets in the customer interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'StateType'} = [
  'open',
  'closed'
];
```

### **Ticket::Frontend::CustomerTicketZoom###AttributesView**

Shows the activated ticket attributes in the customer interface (0 = Disabled and 1 = Enabled).

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'AttributesView'} = {
  'Owner' => '0',
  'Priority' => '1',
  'Queue' => '1',
  'Responsible' => '0',
  'SLA' => '0',
  'Service' => '0',
  'State' => '1',
  'Type' => '0'
};
```

### **Ticket::Frontend::CustomerTicketZoom###DisplayNoteFrom**

Defines the displayed style of the From field in notes that are visible for customers. A default agent name can be defined in Ticket::Frontend::CustomerTicketZoom###DefaultAgentName setting.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'DisplayNoteFrom'} = 'FromField';
```

### **Ticket::Frontend::CustomerTicketZoom###DefaultAgentName**

Defines the default agent name in the ticket zoom view of the customer interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'DefaultAgentName'} = 'Support Agent';
```

### **Ticket::Frontend::CustomerTicketZoom###DynamicField**

Dynamic fields shown in the ticket zoom screen of the customer interface. Possible settings: 0 = Disabled, 1 = Enabled.

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'DynamicField'} = {};
```

## **Ticket::Frontend::CustomerTicketZoom###FollowUpDynamicField**

Dynamic fields options shown in the ticket reply section in the ticket zoom screen of the customer interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'FollowUpDynamicField'} = {};
```

## **Ticket → Frontend::Customer::TicketOverview**

### **Ticket::Frontend::CustomerTicketOverviewSortable**

Controls if customers have the ability to sort their tickets.

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketOverviewSortable'} = '';
```

### **Ticket::Frontend::CustomerTicketOverview###ColumnHeader**

Shows either the last customer article's subject or the ticket title in the small format overview.

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketOverview'}->{'ColumnHeader'} = 'TicketTitle';
```

### **Ticket::Frontend::CustomerTicketOverview###Owner**

Show the current owner in the customer interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketOverview'}->{'Owner'} = '0';
```

### **Ticket::Frontend::CustomerTicketOverview###Queue**

Show the current queue in the customer interface.

不能使这个设置失效。

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketOverview'}->{'Queue'} = '0';
```

### **Ticket::Frontend::CustomerTicketOverview###DynamicField**

Dynamic fields shown in the ticket overview screen of the customer interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.

默认值:

```
$Self->{'Ticket::Frontend::CustomerTicketOverview'}->{'DynamicField'} = {};
```

## **Ticket → Frontend::Queue::Preferences**

### **QueuePreferences###Comment2**

Parameters of the example queue attribute Comment2.

这个设置默认没有激活。

默认值:

```
$Self->{'QueuePreferences'}->{'Comment2'} = {  
  'Block' => 'TextArea',  
  'Cols' => '50',  
  'Desc' => 'Define the queue comment 2.',  
  'Label' => 'Comment2',  
  'Module' => 'Kernel::Output::HTML::QueuePreferences::Generic',  
  'PrefKey' => 'Comment2',  
  'Rows' => '5'  
};
```

## Ticket → Frontend::SLA::Preferences

### SLAPreferences###Comment2

Parameters of the example SLA attribute Comment2.

这个设置默认没有激活。

默认值:

```
$Self->{'SLAPreferences'}->{'Comment2'} = {  
  'Block' => 'TextArea',  
  'Cols' => '50',  
  'Desc' => 'Define the sla comment 2.',  
  'Label' => 'Comment2',  
  'Module' => 'Kernel::Output::HTML::SLAPreferences::Generic',  
  'PrefKey' => 'Comment2',  
  'Rows' => '5'  
};
```

## Ticket → Frontend::Service::Preferences

### ServicePreferences###Comment2

Parameters of the example service attribute Comment2.

这个设置默认没有激活。

默认值:

```
$Self->{'ServicePreferences'}->{'Comment2'} = {  
  'Block' => 'TextArea',  
  'Cols' => '50',  
  'Desc' => 'Define the service comment 2.',  
  'Label' => 'Comment2',  
  'Module' => 'Kernel::Output::HTML::ServicePreferences::Generic',  
  'PrefKey' => 'Comment2',  
  'Rows' => '5'  
};
```

# 附录 C. GNU 自由文档许可证

版本1.1, 2000年3月

Copyright (C) 2000 Free Software Foundation, Inc. 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

## 0. 导言

本协议着重于保证手册、教程或其它功能、用途的文档的自由：以确保任何人不管在商业领域还是非商业领域都可以复制和(修改或没有修改并)重新发布这些文档的自由。其次，本协议保护文档作者和发布者由他们的工作获得的信誉不会因他人对其文档的修改而受损。

本协议是一种“copyleft”，其含义为：文档的派生产物必须与原文档同样是自由的。本文档是 GNU 通用公共协议(为自由软件设计的一种copyleft协议)的补充。

我们设计本协议供自由软件的手册使用，因为自由软件需要自由文档：一个自由的程序附带的手册当然要享有与该软件同等的自由。不过本协议并不仅限于对程序的手册使用，所有的文档作品都可以使用本协议。我们鼓励那些指导性性质或参考书性质的文档作品使用本协议。

## 1. 适用范围与约定

本协议适用于任何媒体上的任何手册或其它文档，文档版权所有人只需在文档中声明使用本协议的条款发布。该声明允许了在后文中的规定下，在世界范围、免版税、无时限地使用该作品。以下的“文档”都是指此类(使用GFDL发布的)手册或作品。公众的任何一个人都是本协议的许可对象，这里将用“你”来称呼。如果你复制、修改或发布了这些(某种意义上也可以说是在版权法保护下的)文档，就表明你已经接受了本协议。

文档的“Modified Version(修正版)”是指任何包含全部或部分文档的作品，不论是照搬照抄还是经过加工修改或是翻译成其它语言。

“Secondary Section(附属章节)”是指指定的附录或文档的前序部分，专门描述有关文档主题的文档出版者或作者的相关信息(或对一些相关情况进行说明)并且包含文档主题不会直接提及的内容。(如果文档是数学教材的一部分，附属章节或许连一点数学都不会提到。) 这些相关信息可以是与主题有关的历史关联、相关事物或者相关的法律、商业、哲理、道德或政治立场。

“Invariant Sections(固定章节)”是某些指定了标题的附属章节，固定章节做为文档的一部分也象声明提到的那样，在本协议的保护下发布。

“Cover Texts(封皮文字)”是那些简短的小段文字列，就象封面或封底的文字。封皮文字做为文档的一部分也象声明提到的那样，在本协议的保护下发布。封面文字一般最多5个词语，封底文字一般最多25个词语。

文档的“Transparent透明”副本是指可以用计算机处理的副本，表现为其格式符合一般通用规范，这样就适于直接用通用文本编辑器或通用绘画程序(对由像素构成的图片)或那些被广泛应用的图像编辑器(用来绘图)修改文档，并且其格式适于输入文本格式器或可转换为适于输入文本格式器的透明格式。如果某副本(不论有没有标记)为反对或防止读者的后续修正而采用其它格式，那么就是不兼容的。如果图片格式的使用有任何实质性专利条文的限制，那么该图片格式就是不透明的。不“透明”的文档副本我们称为“Opaque不兼容”。

文档副本适用的透明格式包括没有标记的纯ASCII码、Texinfo格式、LaTeX格式、SGML格式或XML格式这些公认有效的文件(格式)类型定义，还有与标准兼容的精简HTML、PostScript或PDF这些便于编辑的格式。图片适用的透明格式包括PNG、XCF和JPG。不透明格式包括用专有文档处理器才能阅读编辑的专有格式、使用非公认的文件格式定义和/或使用非公认的处理工具的SGML格式或XML格式，还有那些机器生成的HTML及某些文档处理器产生的PostScript或PDF等只用于输出的格式。

“Title Page(扉页)”是指刊印成册的书的扉页本身及附加下列有必要明显地保留的页面：本协议要求在扉页出现的材料。像那些版式中没有扉页的作品，“Title Page”是指在作品正文之前最显著的标题附近的文字。

## 2. 原样复制

在提供本协议和版权声明，同时在许可声明中指明文档及其所有副本均在本协议保护下，并且不加入任何其它条件到文档的协议的前提下，你可以在任何媒体复制和发布该文档，不论是用于商业或非商业。你不可以对你制作的副本进行技术处理以妨碍或控制阅读或再复制。不过，你可以因交换副本而接受他人给你的一些补偿。如果你发布的副本数量大到一定程度，你还必须看看下面的第3节。

在与上述相同的情形下，你还可以出借文档副本和公开展示文档副本。

## 3. 大量发布

如果你发布了印制的文档副本数量超过100份，你必须把副本用清晰明了地印有文档的许可声明所要求的封皮文字的封皮封装好，这些封皮文字包括封面文字和封底文字。封面和封底还都必须写明你是该副本的发布者。封面必须印上完整的书名，书名的每个字都必须同样地显著。你可以在封面添一些其它素材做为补充。在满足了遵循对封皮改动的限制、维持文档标题等这些条件下，副本在某方面上可以被视同原样复制的副本。

如果封面或封底必要的文字太多而导致看起来不清晰明了，那么你可以把优先的条目(尽量适合地)列在封面或封底，剩下的那些接在邻近的页面里。

如果你发布了印制的文档副本数量超过100份，你必须把副本用清晰明了地印有文档的许可声明所要求的封皮文字的封皮封装好，这些封皮文字包括封面文字和封底文字。封面和封底还都必须写明你是该副本的发布者。封面必须印上完整的书名，书名的每个字都必须同样地显著。你可以在封面添一些其它素材做为补充。在满足了遵循对封皮改动的限制、维持文档标题等这些条件下，副本在某方面上可以被视同原样复制的副本。

在你对文档副本进行大量再发布前取得与文档作者良好的联系，这虽然不是必须的，但是是有必要的，这样你就有机会得到他们提供的文档的最新版本。

## 4. 修正

你可以在上文2、3两节的条件下复制和发布文档的修正版，只要你严格地在本协议下发布承担了原来文档的任务的修正版，这样就允许任何持有该修正版一份副本的人可对修正版进行发行和修改。而且，你必须对修正版做如下一些事情：

- A. 在扉页(如果有封皮，则封皮和扉页都要)使用的书名要与原来文档的书名明显地区别开来，以及那些先前版本(哪些是需要区别的呢？如果原来文档有历史小节，那么历史小节中列出的都是)。如果最初的发行人允许的话，你也可以使用相同的书名。
- B. 扉页上必须列出作者、对修正版文档的修改负责的个人或团体或机构、文档的至少五个主要作者(如果主要作者少于五个，就全部写上)，除非他们允许你不这么做。
- C. 扉页上必须列出作者、对修正版文档的修改负责的个人或团体或机构、文档的至少五个主要作者(如果主要作者少于五个，就全部写上)，除非他们允许你不这么做。
- D. 保留原有文档的所有版权声明不变。
- E. 在其它版权声明附近为你所做的修正添加适当的版权声明。
- F. 在版权声明后面立即接上允许公众在本协议下使用修正版文档的许可声明，其形式如下面附录所示。
- G. 在文档的许可声明里保留所有固定章节和封皮文字所要求的许可声明。

H. 放上这些许可证的一份原件。

I. 保留题为“历史”的章节，不要改动其标题，并且给它添加一条包括修改版的书名、日期、作者、发布者的条目，就像扉页里给出的那样。如果原文档没有题为“历史”的章节，你就建立该章节，创建一条包括原文档的书名、日期、作者、发布者的条目，如原文档的扉页里给出的那样，然后再象上面规定那样，添一条描述修改版的条目。

J. 如果文档中给出了访问兼容文档的链接，你应该保护网址的有效，并且该网址还应该提供该文档的前承版本。这些链接可以在“历史”章节给出。如果前承文档发布已超过四年或者原作者在文档中提到不必指明前承版本，那么你可以在作品中略去其链接。

K. 原文档中的任何题为“鸣谢”或“贡献”的章节都必须维持其标题和章节中对每位贡献者的谢词的原旨和感情。

L. 维持原文档的固定章节的标题和内容不变。当然，章节标号不属于章节标题的一部分(可以改动)。

M. 删除原文档中任何题为“注记”的章节。这些章节可能是之针对原文档的。

N. 不要把任何已有章节标题改为“注记”，或改为与固定章节有冲突的标题。

如果文档修正版包含了新的没有从原文档摘取素材的有附属章节资格的前序章节或附录，你可以在自己的选择下指定这些章节的全部或一些为固定章节。你需要为修正版的许可声明中的固定章节序列添加标题。这些标题必须与其它章节的标题明显地区别开来。

你可以添加题为“注记”的章节，只要包含各方面对你的修正版的评议，例如：综合评论或被某组织认可的某标准的权威定义的原文。

你可以在修正版封皮文字序列末尾添加5个词左右的书名和25个词左右的封底文字。一个实体(进行出版的个人或组织)可能只会添加(或通过整理形成)一段封面文字和封底文字。如果文档已经包含同样用于封皮的封皮文字(先前由你添加或由你代理的实体整理)，你就不要再另外添加了；不过在添加旧封皮的原发布者的明确许可下你可以把旧的替换掉。

本协议不允许文档作者和发布者用他们的名字公开声称或暗示对任何修正版的认可。

## 5. 合并文档

你可以在本协议下依据第4节里对修正版定义的条款把文档与其它文档合并起来发布，只要你在合并文档里原封不动地包含所有原始文档的所有固定章节，并在你的合并文档的许可声明中将其全部列为固定章节，而且保留其所有免责声明。

合并后文档只需包含一份本协议，重复的多处固定章节应该用单独的章节替代。如果重复的固定章节只是标题相同，但内容是不同的，就在每一小段标题末端的括号里添加已知的原作者或原发布者(的名字)，或给每小段一个唯一的标号。在合并文档的许可声明里的固定章节中的章节标题也要做相应的调整。

在合并文档里，你必须把不同原文档的“历史”章节合并成一个“历史”章节；同样需要合并的章节有“鸣谢”、“贡献”。你还必需把所有“注记”章节删除。

## 6. 文档合集

如果你对每份文档不论在哪方面都遵循了本协议的原样复制条款，那么你可以在本协议下发布你以该文档和其他文档制作的合集，并以一个单独的副本来替代合集中各文档里分别使用的本协议副本。

你可以从该合集中提取一个单独的文档，并在本协议下单独发布，只要你在该提取出的文档中加入一份本协议的副本，并在文档的其他所有方面都遵循本协议的原样复制的条款。

## 7. 独立作品汇集

文档和其它独立的受本协议约束的派生文档或作品编纂在一个大文库中或大媒体上，如果汇编作品的版权对该汇编作品用户的权利限制没有超出那些独立作品的许可范围，该汇编作品称



为“aggregate(汇集)”。当文档被包含在汇集中的时候，本协议不对汇集中该文档的非直承(不是以本协议发布的)作品起作用。

如果第3节中对封皮文字的要求适用于文档的该副本，那么如果文档在汇集中所占的比例小于全文的一半，文档的封皮文字可以被置于汇集内该文档的内封页上，或是电子文档中的等效部分(如果文档以电子版发布)。否则，必须将其现于封装整个汇集的印好的封皮上。

## 8. 翻译

我们认为翻译是一种修改，所以你可以按照第4节的条款发布文档的翻译版本。如果要将文档的固定章节用译文取代，需要得到版权持有者的许可，但你可以将部分或全部固定章节的译文附加在原始版本的后面。你可以包含一份本协议、所有许可声明及免责声明的译文，只要你同时包含其原始英文版本即可。当译文和原文有歧义的时候，以原文为准。

## 9. 协议终止

除非你明确遵循本协议，否则你不可以对文档进行复制、修改、分授许可及发布。对文档的任何其它(违反本协议的)意图的复制、修改、分授许可及发布是无效的，并且这将自动终止你在本协议下享有的权利。不过，从你这里获得文档副本或权利的群体的协议不会被终止，只要他们完全遵循本协议。

## 10. 本协议的未来修订版

自由软件基金会有时会发布新的 GNU自由文档协议修订版。新版本将会和当前版本体现类似的精神，但在处理某些新的问题和利害关系的细节上会有所不同。参阅 <http://www.gnu.org/copyleft/>。

本协议的每个版本都有一个唯一的版本号。如果文档指定遵循一个特定的本协议版本“或任何后续版本”，你可以选择遵循指定版本或自由软件基金会发布的任何后续(非草案)版本的条款和条件。如果文档没有指定本许可证的版本，那么你可以选择遵循任何自由软件基金会已发布的(非草案)版本。

## · 你的文档怎样使用GNU自由文档许可证

要使用本许可证发布你写的文档，请在文档中包含本协议的一个副本，并在紧接扉页之后加入如下版权声明与许可声明：

```
Copyright (c) YEAR YOUR NAME. Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.1 or any later version published by the Free Software Foundation; with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST. A copy of the license is included in the section entitled "GNU Free Documentation License".
```

如果你没有固定章节，请用“with no Invariant Sections”替换“with the Invariant Sections being LIST THEIR TITLES”，如果你没有封面文字，请用“no Front-Cover Texts”替换“Front-Cover Texts being LIST”；封底文字也同样按此处理。

如果你的文件中包含有并非微不足道的程序代码范例，我们建议以你选择的自由软件授权比如GNU General Public License 来并行地发布这些范例，从而允许它们作为自由软件使用。