

Documentation

OTRS 3.1 - Admin Manual

Build Date:

2012-02-14

OTRS 3.1 - Admin Manual

Авторские права © 2003-2012 OTRS AG

René Bakker, Stefan Bedorf, Michiel Beijen, Shawn Beasley, Hauke Böttcher, Jens Bothe, Udo Bretz, Martin Edenhofer, Carlos Javier García, Martin Gruner, Manuel Hecht, Christopher Kuhn, André Mindermann, Marc Nilius, Elva María Novoa, Henning Oschwald, Martha Elia Pascual, Thomas Raith, Carlos Fernando Rodríguez, Stefan Rother, Burchard Steinbild, Daniel Zamorano.

Эта работа защищена авторским правом OTRS AG.

Это руководство может быть скопировано целиком или частично, при условии, что это уведомление об авторских правах будет включено в каждую копию.

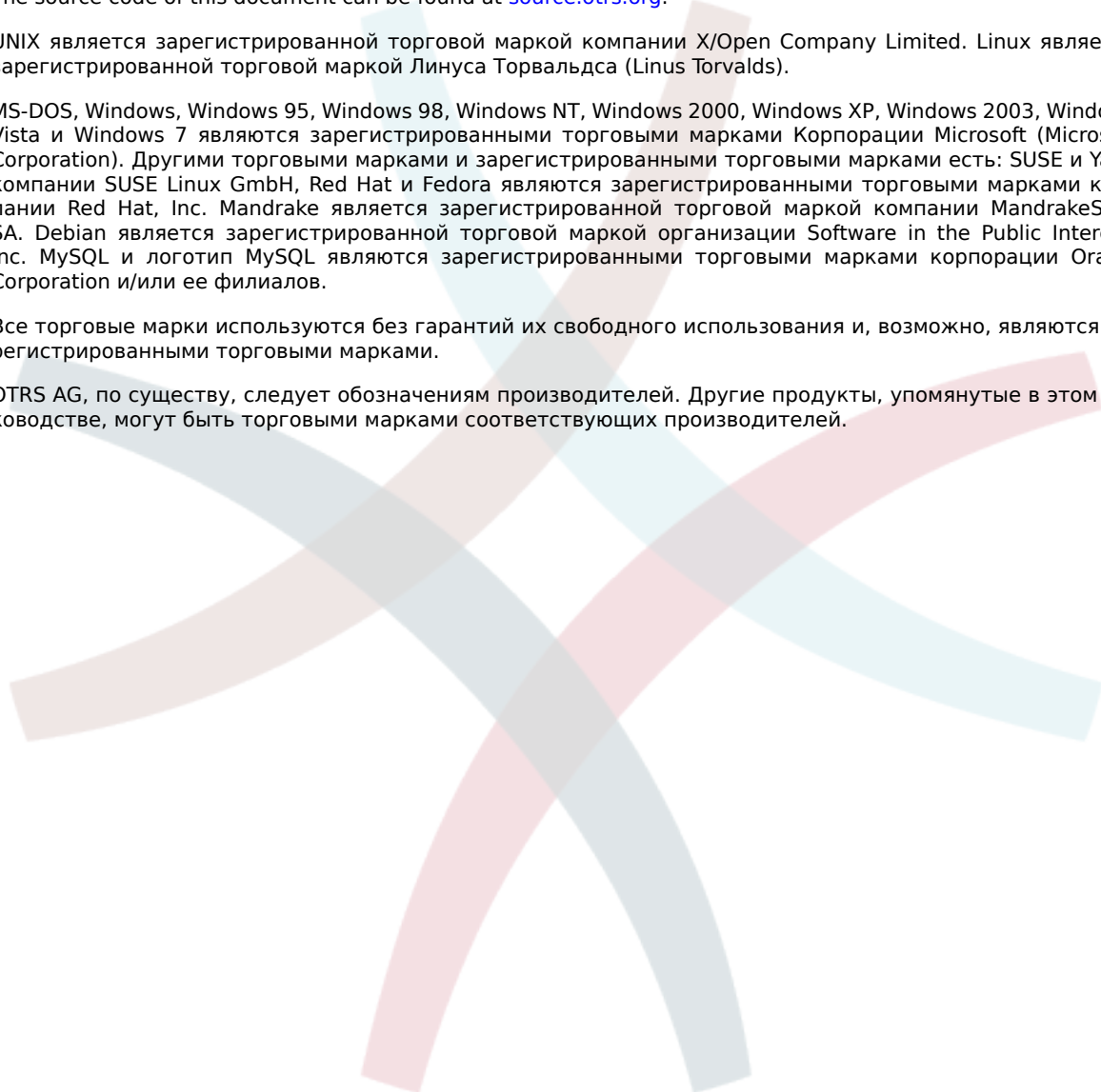
The source code of this document can be found at source.otrs.org.

UNIX является зарегистрированной торговой маркой компании X/Open Company Limited. Linux является зарегистрированной торговой маркой Линуса Торвальдса (Linus Torvalds).

MS-DOS, Windows, Windows 95, Windows 98, Windows NT, Windows 2000, Windows XP, Windows 2003, Windows Vista и Windows 7 являются зарегистрированными торговыми марками Корпорации Microsoft (Microsoft Corporation). Другими торговыми марками и зарегистрированными торговыми марками есть: SUSE и YaST компании SUSE Linux GmbH, Red Hat и Fedora являются зарегистрированными торговыми марками компании Red Hat, Inc. Mandrake является зарегистрированной торговой маркой компании MandrakeSoft, SA. Debian является зарегистрированной торговой маркой организации Software in the Public Interest, Inc. MySQL и логотип MySQL являются зарегистрированными торговыми марками корпорации Oracle Corporation и/или ее филиалов.

Все торговые марки используются без гарантий их свободного использования и, возможно, являются зарегистрированными торговыми марками.

OTRS AG, по существу, следует обозначениям производителей. Другие продукты, упомянутые в этом руководстве, могут быть торговыми марками соответствующих производителей.



Содержание

Предисловие	xii
1. Системы Обработки Заявок - Основы	1
1. Что такое система обработки заявок и зачем она нужна?	1
2. Что такое заявка?	2
2. OTRS Help Desk	3
1. Основы	3
2. Особенности и функциональные возможности	3
2.1. New features of OTRS 3.1	5
2.2. Новые возможности OTRS 3.0	9
2.3. Новые возможности OTRS 2.4	10
3. Требования к аппаратному и программному обеспечению	13
3.1. Perl support	13
3.2. Web server support	13
3.3. Database support	14
3.4. Web browser support	14
4. Комюнити	14
5. Коммерческая Поддержка и Услуги для OTRS	14
3. Установка	16
1. Самый простой способ - установка из готовых бинарных пакетов	16
1.1. Установка из rpm-пакетов на сервер под управлением Suse Linux	16
1.2. Установка OTRS в операционной системе CentOS	17
1.3. Установка OTRS в операционной системе Debian	17
1.4. Установка OTRS на системе Ubuntu	17
1.5. Установка OTRS на операционной системе Microsoft Windows	18
2. Установка из исходных кодов (Linux, Unix)	18
2.1. Подготовка к установке из исходных кодов	18
2.2. Установка Perl-модулей	19
2.3. Настройка веб-сервера Apache	21
2.4. Настройка базы данных	22
2.5. Настрой планировщика задач (cron jobs) для OTRS	28
3. Обновление OTRS-фреймворка	32
4. Обновление с помощью Windows Installer	36
5. Upgrading Microsoft SQL Server Data Types	37
4. Первые шаги	38
1. Веб-интерфейс для агентов	38
2. Пользовательский веб-интерфейс	38
3. Общедоступный веб-интерфейс	39
4. Первое применение	39
5. Веб-интерфейс - обзор	40
6. Что такое очередь?	42
7. Пользовательские настройки	43
5. Панель управления OTRS	46
1. Основы	46
2. Агенты, Группы и Роли	46
2.1. Агенты	46
2.2. Группы	47
2.3. Роли	50
3. Клиенты и Группы Клиентов	54
3.1. Клиенты	54
3.2. Группы Клиентов	56
4. Очереди	57
5. Приветствия, подписи, вложения и ответы	59
5.1. Приветствия	59
5.2. Подписи	60

5.3. Вложения	61
5.4. Ответы	63
6. Авто ответы	65
7. Адреса электронной почты	68
8. Уведомления	69
9. SMIME	71
10. PGP	71
11. Состояния	72
12. SysConfig	73
13. Использование почтовый ящиков	74
14. Filtering incoming email messages	74
15. Выполнение автоматизированных заданий с помощью GenericAgent	77
16. Электронная почта администратора	78
17. Управление сеансами	78
18. Системный журнал	79
19. SQL запросы	80
20. Менеджер пакетов	80
21. Web Services	81
22. Dynamic Fields	82
6. Настройка Системы	83
1. Конфигурационные файлы OTRS	83
2. Настройка системы через графический веб-интерфейс	83
7. Отправка / получение электронной почты	85
1. Отправка почты	85
1.1. через Sendmail (по умолчанию)	85
1.2. Через SMTP-сервер или smarthost	85
2. Получение сообщений электронной почты	85
2.1. Учетные записи электронной почты настраиваемые через графический пользовательский интерфейс OTRS	85
2.2. Через командную строку программы и например, procmail (otrs.PostMaster.pl)	86
2.3. Получение электронной почты по протоколу POP3 или IMAP и обработка для otrs.PostMaster.pl	87
2.4. Фильтрация/рассылка модулями OTRS/PostMaster (для более сложной диспетчеризации)	87
8. Функции связанные с временем	89
1. Настройка периода рабочего времени, праздников и часовых поясов	89
1.1. Бизнес Время	89
1.2. Праздники с фиксированными датами	89
1.3. TimeVacationDaysOneTime	89
2. Автоматические Разблокировки	89
9. Ответственность за Заявку & Наблюдение за Заявкой	91
1. Ответственность за Заявку	91
2. Просмотр Заявок	92
10. Customizing the PDF output	94
11. Использование внешних хранилищ данных	95
1. Данные клиентов	95
2. Пользовательский бэк-энд	95
2.1. База Данных (По умолчанию)	95
2.2. LDAP	98
2.3. Использование больше чем одного хранилища информации с OTRS	100
3. Хранилища (бэк-энды) для аутентификации Агентов и Клиентов	102
3.1. Хранилища данных (бэк-энды) для аутентификации Агентов	102
3.2. Хранилища информации для аутентификации (проверки подлинности) клиентов	104
4. Customizing the customer self-registration	106
4.1. Настройка веб-интерфейса	106

4.2. Отображения клиентов	106
4.3. Customizing the customer_user table in the OTRS DB	107
12. Состояния	109
1. Предопределенные состояния	109
1.1. Новая	109
1.2. Открытая	109
1.3. Ожидание с напоминанием	109
1.4. Ожидание авто-закрытие -	109
1.5. Ожидание авто-закрытие+	109
1.6. Объединенные	109
1.7. Закрыта Успешно	109
1.8. Закрыта Неудачно	109
2. Настраиваемы состояния	109
13. Изменение приоритетов заявки	113
14. Создание своих собственных тем (шаблонов)	114
15. Локализация интерфейса OTRS	115
16. PGP	116
17. S/MIME	119
18. Списки Контроля Доступа (ACLs)	123
1. Introduction	123
2. Examples	123
3. Reference	125
19. Stats-модуль	129
1. Обработка модуля агентом	129
1.1. Обзор	129
1.2. Создание и просмотр отчетов	130
1.3. Редактировать / Создать новый	133
1.4. Импорт	138
2. Администрирование модуля статистики OTRS администратором.	139
2.1. Настройки полномочий, Групп и Ролей	139
2.2. SysConfig	139
3. Администрирование модуля статистики системным администрато- ром	139
3.1. Таблица базы данных	139
3.2. Список файлов	139
3.3. Кэширование	140
3.4. otrs.GenerateStats.pl	140
3.5. Автоматическое создание статистики с помощью планировщика заданий - Cronjob	140
3.6. Статическая статистика	141
3.7. Использование устаревших данных статической статистики	141
3.8. Статистика по умолчанию	141
20. Generic Interface	142
1. Generic Interface Layers	142
1.1. Network Transport	142
1.2. Data Mapping	142
1.3. Controller	143
1.4. Operation (OTRS as a provider)	143
1.5. Invoker (OTRS as a requester)	143
2. Generic Interface Communication Flow	143
2.1. OTRS as Provider	143
2.2. OTRS as Requester	144
3. Web Services	146
4. Web Service Graphical Interface	146
4.1. Web Service Overview	146
4.2. Web Service Add	147
4.3. Web Service Change	147
5. Web Service Command Line Interface	154

5.1. Web Service Configuration	154
5.2. Web Service Debugger	155
6. Web Service Configuration	156
6.1. Configuration Details	157
7. Connectors	162
7.1. Bundled Connectors	162
7.2. Examples:	170
21. OTRS Scheduler	174
1. Scheduler Graphical Interface	174
1.1. Scheduler Not Running Notification	174
1.2. Start Scheduler	174
2. Scheduler Command Line Interface	175
2.1. Unix / Linux	175
2.2. Windows	177
22. Dynamic Fields	179
1. Introduction	179
2. Configuration	179
2.1. Adding a Dynamic Field	180
2.2. Text Dynamic Field Configuration	181
2.3. Textarea Dynamic Field Configuration	182
2.4. Checkbox Dynamic Field Configuration	182
2.5. Dropdown Dynamic Field Configuration	182
2.6. Multiselect Dynamic Field Configuration	183
2.7. Date Dynamic Field Configuration	184
2.8. Date / Time Dynamic Field Configuration	184
2.9. Editing a Dynamic Field	185
2.10. Showing a Dynamic Field on a Screen	185
2.11. Setting a Default Value by a Ticket Event Module	188
2.12. Set a Default Value by User Preferences	189
2.13. Updating from OTRS 3.0	190
23. Дополнительные приложения	191
1. FAQ (Часто Задаваемые Вопросы)	191
24. Настройка Производительности	192
1. OTRS	192
1.1. TicketIndexModule	192
1.2. TicketStorageModule	192
1.3. Архивирование Заявок	193
2. База данных	194
2.1. MySQL	194
2.2. PostgreSQL	194
3. Веб-сервер	194
3.1. Предустановленное соединение с базой данных	194
3.2. Предварительно загруженные модули - startup.pl	194
3.3. Перезагрузка Perl-модулей во время обновления с диска	194
3.4. Выбор Правильной Стратегии	195
3.5. mod_gzip/mod_deflate	195
25. Резервное копирование системы	196
1. Резервное копирование	196
2. Восстановление	196
A. Дополнительные ресурсы	198
1. Домашняя страница OTRS.org	198
2. Списки рассылки	198
3. Трекинг ошибок	199
4. Коммерческая поддержка	200
B. Configuration Options Reference	201
1. Framework	201
1.1. Core	201
1.2. Core::LinkObject	210

1.3. Core::Log	211
1.4. Core::MIME-Viewer	213
1.5. Core::MirrorDB	214
1.6. Core::PDF	215
1.7. Core::Package	218
1.8. Core::PerformanceLog	220
1.9. Core::ReferenceData	221
1.10. Core::SOAP	221
1.11. Core::Sendmail	222
1.12. Core::Session	224
1.13. Core::SpellChecker	228
1.14. Core::Stats	229
1.15. Core::Stats::Graph	230
1.16. Core::Time	235
1.17. Core::Time::Calendar1	239
1.18. Core::Time::Calendar2	241
1.19. Core::Time::Calendar3	244
1.20. Core::Time::Calendar4	246
1.21. Core::Time::Calendar5	249
1.22. Core::Time::Calendar6	252
1.23. Core::Time::Calendar7	254
1.24. Core::Time::Calendar8	257
1.25. Core::Time::Calendar9	259
1.26. Core::Web	262
1.27. Core::WebUserAgent	274
1.28. Crypt::PGP	275
1.29. Crypt::SMIME	276
1.30. Frontend::Admin::AdminCustomerUser	277
1.31. Frontend::Admin::ModuleRegistration	278
1.32. Frontend::Agent	291
1.33. Frontend::Agent::Dashboard	297
1.34. Frontend::Agent::LinkObject	300
1.35. Frontend::Agent::ModuleMetaHead	300
1.36. Frontend::Agent::ModuleNotify	300
1.37. Frontend::Agent::ModuleRegistration	302
1.38. Frontend::Agent::NavBarModule	307
1.39. Frontend::Agent::Preferences	307
1.40. Frontend::Agent::SearchRouter	312
1.41. Frontend::Agent::Stats	312
1.42. Frontend::Customer	315
1.43. Frontend::Customer::Auth	322
1.44. Frontend::Customer::ModuleMetaHead	329
1.45. Frontend::Customer::ModuleNotify	329
1.46. Frontend::Customer::ModuleRegistration	330
1.47. Frontend::Customer::Preferences	331
1.48. Frontend::Public	334
1.49. Frontend::Public::ModuleRegistration	335
2. Ticket	335
2.1. Core	335
2.2. Core::FulltextSearch	336
2.3. Core::LinkObject	337
2.4. Core::PostMaster	337
2.5. Core::Stats	346
2.6. Core::Ticket	347
2.7. Core::TicketACL	364
2.8. Core::TicketBulkAction	364
2.9. Core::TicketDynamicFieldDefault	365
2.10. Core::TicketWatcher	371

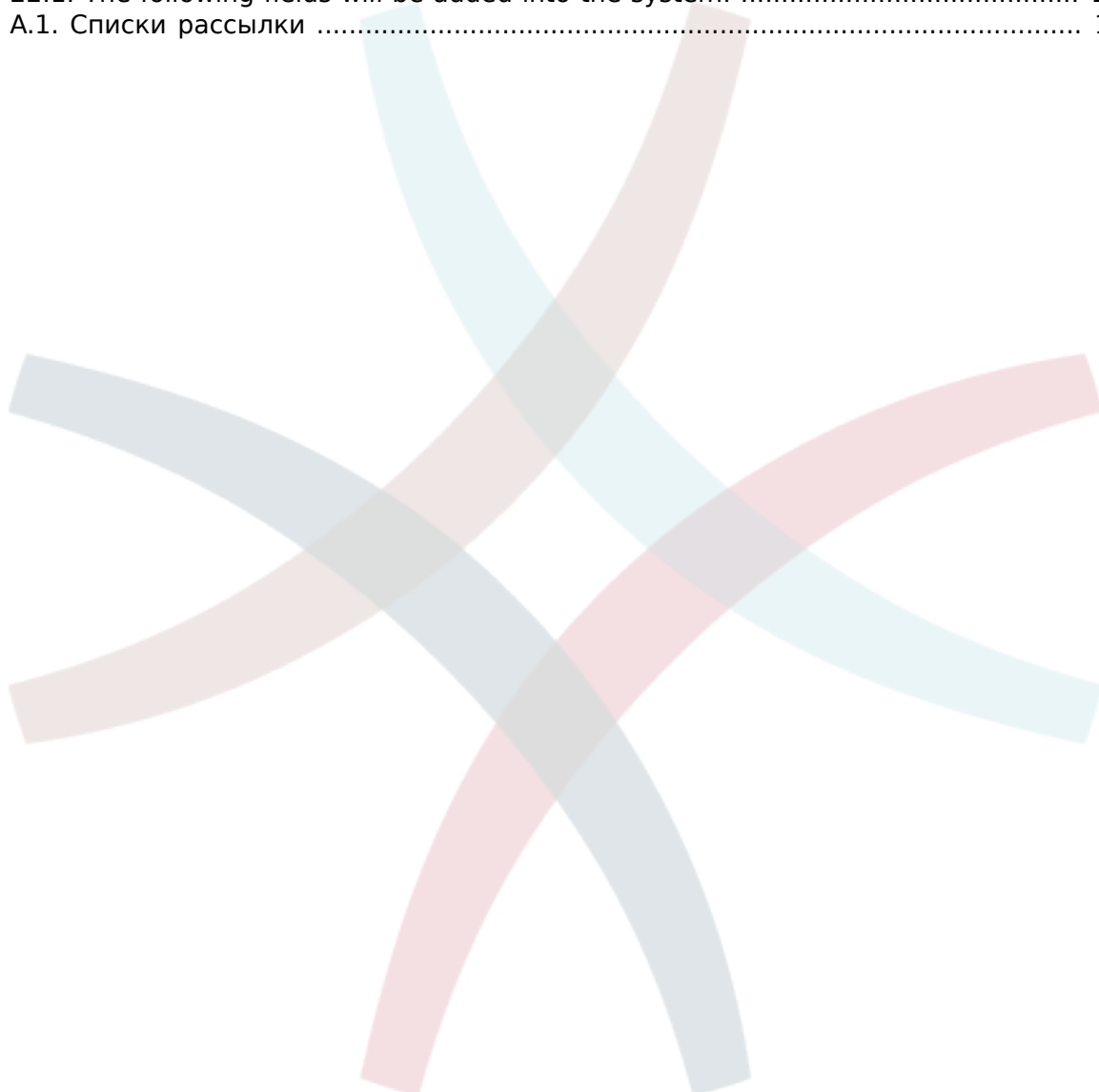
2.11. Frontend::Admin::ModuleRegistration	371
2.12. Frontend::Agent	379
2.13. Frontend::Agent::CustomerSearch	386
2.14. Frontend::Agent::Dashboard	388
2.15. Frontend::Agent::ModuleMetaHead	391
2.16. Frontend::Agent::ModuleNotify	391
2.17. Frontend::Agent::ModuleRegistration	391
2.18. Frontend::Agent::Preferences	404
2.19. Frontend::Agent::SearchRouter	409
2.20. Frontend::Agent::Ticket::ArticleAttachmentModule	409
2.21. Frontend::Agent::Ticket::ArticleComposeModule	410
2.22. Frontend::Agent::Ticket::ArticleViewModule	410
2.23. Frontend::Agent::Ticket::ArticleViewModulePre	411
2.24. Frontend::Agent::Ticket::MenuModule	411
2.25. Frontend::Agent::Ticket::MenuModulePre	418
2.26. Frontend::Agent::Ticket::ViewBounce	421
2.27. Frontend::Agent::Ticket::ViewBulk	422
2.28. Frontend::Agent::Ticket::ViewClose	425
2.29. Frontend::Agent::Ticket::ViewCompose	431
2.30. Frontend::Agent::Ticket::ViewCustomer	434
2.31. Frontend::Agent::Ticket::ViewEmailNew	435
2.32. Frontend::Agent::Ticket::ViewEscalation	438
2.33. Frontend::Agent::Ticket::ViewForward	439
2.34. Frontend::Agent::Ticket::ViewFreeText	441
2.35. Frontend::Agent::Ticket::ViewHistory	447
2.36. Frontend::Agent::Ticket::ViewMailbox	447
2.37. Frontend::Agent::Ticket::ViewMerge	449
2.38. Frontend::Agent::Ticket::ViewMove	450
2.39. Frontend::Agent::Ticket::ViewNote	452
2.40. Frontend::Agent::Ticket::ViewOwner	458
2.41. Frontend::Agent::Ticket::ViewPending	464
2.42. Frontend::Agent::Ticket::ViewPhoneInbound	470
2.43. Frontend::Agent::Ticket::ViewPhoneNew	473
2.44. Frontend::Agent::Ticket::ViewPhoneOutbound	476
2.45. Frontend::Agent::Ticket::ViewPrint	479
2.46. Frontend::Agent::Ticket::ViewPriority	480
2.47. Frontend::Agent::Ticket::ViewQueue	486
2.48. Frontend::Agent::Ticket::ViewResponsible	488
2.49. Frontend::Agent::Ticket::ViewSearch	494
2.50. Frontend::Agent::Ticket::ViewStatus	504
2.51. Frontend::Agent::Ticket::ViewZoom	504
2.52. Frontend::Agent::TicketOverview	507
2.53. Frontend::Agent::ToolBarModule	510
2.54. Frontend::Customer	513
2.55. Frontend::Customer::ModuleMetaHead	517
2.56. Frontend::Customer::ModuleRegistration	517
2.57. Frontend::Customer::Preferences	520
2.58. Frontend::Customer::Ticket::ViewNew	521
2.59. Frontend::Customer::Ticket::ViewPrint	525
2.60. Frontend::Customer::Ticket::ViewSearch	525
2.61. Frontend::Customer::Ticket::ViewZoom	528
2.62. Frontend::Queue::Preferences	531
2.63. Frontend::SLA::Preferences	531
2.64. Frontend::Service::Preferences	532
C. Благодарности	533
D. GNU Free Documentation License	535
0. PREAMBLE	535
1. APPLICABILITY AND DEFINITIONS	535

2. VERBATIM COPYING	536
3. COPYING IN QUANTITY	536
4. MODIFICATIONS	537
5. COMBINING DOCUMENTS	538
6. COLLECTIONS OF DOCUMENTS	538
7. AGGREGATION WITH INDEPENDENT WORKS	538
8. TRANSLATION	539
9. TERMINATION	539
10. FUTURE REVISIONS OF THIS LICENSE	539
How to use this License for your documents	539



Список таблиц

3.1. Perl-модули, необходимые для работы OTRS	19
3.2. Описание некоторых сценариев планировщика задач cron job.	29
5.1. Предопределенные группы, доступны для новой (только что) установленной системы OTRS	47
5.2. Права, связанные с OTRS-группами	50
5.3. События для Автоматических ответов	66
5.4. Функции различных X-OTRS-заголовков	75
21.1. List of Init Scripts And Supported Operating Systems	175
22.1. The following fields will be added into the system:	181
A.1. Списки рассылки	198



Список примеров

5.1. Сортирование спама в определенную очередь	76
7.1. .fetchmailrc	87
7.2. Пример задания для модуля фильтрации Kernel::System::PostMaster::Filter::Match	88
7.3. Пример задания для модуля фильтрации Kernel::System::PostMaster::Filter::CMD	88
11.1. Настройка клиентского хранилища базы данных (DB)	95
11.2. Хранение Заявок Компании в базе данных DB	98
11.3. Настройка LDAP в качестве клиентского бэк-енда	98
11.4. Использование заявок Компании с LDAP-бэкэндом	100
11.5. Использование больше чем одного пользовательского хранилища дан- ных с OTRS	100
11.6. Проверка подлинности агентов путем использования Базы Данных (DB) в качестве хранилища информации.	102
11.7. Проверка подлинности агентов при использовании LDAP в качестве хра- нилища данных	103
11.8. Аутентификация Агентов с помощью HTTPBasic	104
11.9. Аутентификация (проверка подлинности) агентов с использованием Radius-сервера в качестве хранилища информации	104
11.10. Аутентификация Клиента в Базе Данных	104
11.11. Аутентификация пользователей с помощью LDAP-бэкэнда	105
11.12. Аутентификация клиентов с помощью HTTPBasic	105
11.13. Аутентификация клиентов с использованием Radius	106
18.1. Списки прав доступа (ACL) позволяют перемещать в очереди только за- явки с приоритетом 5.	123
18.2. ACL allowing movement into a queue of only those tickets with ticket priority 5 stored in the database.	123
18.3. Списки прав доступа (ACL) делают невозможным закрытие заявок в оче- реди raw и скрывают кнопку "закрыть".	124
18.4. ACL removing always state closed successful.	125
18.5. ACL only showing Hardware services for tickets that are created in queues that start with "HW".	125
18.6. Reference showing all possible important ACL settings.	125
19.1. Определение значения серии - один из элементов	137
19.2. Определение значения серии - один из элементов	137
21.1. Example To Start The OTRS Scheduler Form An Init.d Script	175
21.2. Example To Start The OTRS Scheduler	176
21.3. Example To Force Stop The OTRS Scheduler	176
21.4. Example To Register The OTRS Scheduler Into the Widows SCM	177
21.5. Example To Start The OTRS Scheduler	177
21.6. Example To Force Stop The OTRS Scheduler	178
22.1. Activate Field1 in New Phone Ticket Screen.	186
22.2. Activate Field1 in New Phone Ticket Screen as mandatory.	186
22.3. Activate several fields in New Phone Ticket Screen.	187
22.4. Deactivate some fields in New Phone Ticket Screen.	187
22.5. Activate Field1 in Ticket Zoom Screen.	187
22.6. Activate Field1 in Ticket Overview Small Screens.	188
22.7. Activate Field1 in TicketCreate event.	188
22.8. Activate Field1 in the User preferences.	189

Предисловие

Эта книга предназначена для использования OTRS-администраторами. Он также послужит хорошим справочником для новичков OTRS.

В следующих главах описывается установка, настройка и администрирование программы OTRS. В первой трети текста описываются ключевые функциональные возможности программного обеспечения, в то время как остальной текст является ссылкой на полный набор настраиваемых параметров.

Эта книга и дальше находится в стадии разработки и совершенствуется с выходом новых версий продукта. Чтобы сделать это справочное руководство более качественным, полным и точным нам нужна обратная связь с вами. Пожалуйста, напишите нам, если заметите отсутствие некоторых разделов в этой книге, если некоторые вещи объясняются недостаточно хорошо или, если просто видите орфографические, грамматические ошибки или опечатки. Мы будем признательны за любой вид обратной связи, которую можно осуществить посредством нашей системы отслеживания ошибок, которая находится по адресу <http://bugs.otrs.org>. Заранее благодарим вас за помощь!



Глава 1. Системы Обработки Заявок - Основы

В этой главе дается краткий обзор идеи заявок в целом, и систем обработки заявок, в частности. Краткий пример иллюстрирует преимущества использования системы обработки заявок.

1. Что такое система обработки заявок и зачем она нужна?

Следующий пример показывает, что собой являет система обработки заявок и как можно сэкономить много времени и денег, используя подобную систему в своей компании.

Давайте представим, что Максим занимается производством видеомэгнитофонов. Максим получает много писем от клиентов, которые нуждаются в помощи по устройству. Спустя несколько дней, у него нету свободного времени чтобы оперативно ответить на все письма клиентов или хотя бы просмотреть их. Некоторые клиенты не хотят долго ждать и пишут второе письмо с тем же вопросом. Все письма, содержащие просьбы о поддержке хранятся в одном почтовом ящике. Запросы не сортируются и Максим отвечает на письма пользуясь обычной программой для работы с электронной почтой.

Поскольку Максим не успевает быстро дать ответ на все письма, его инженеры: Андрей и Александр берутся помочь Максиму. Для работы с почтой Андрей и Александр используют один и тот же почтовый ящик, который хранит все письма клиентов. Они ничего не подозревают о том, что Максим получил два одинаковых письма с вопросом о помощи от отчаявшегося клиента. Временами они оба отвечают на один и тот же запрос, и клиент получает два различных ответа. Более того, Максим не имеет представления о содержимом этих ответов. Также он не знает подробностей большинства проблем его клиентов и о том как они решаются, какие проблемы возникают чаще всего, а также, сколько времени и денег он должен потратить на поддержку клиентов.

На собрании коллега рассказывает Максиму о системах обработки заявок и как они могут решить его проблемы с поддержкой клиентов. После поиска информации в Интернете, Максим принимает решение установить Open Ticket Request System (OTRS) - Открытую Систему Обработки Заявок (далее просто OTRS) на компьютере, который будет доступен из веб как для клиентов компании так и для ее сотрудников. С этого момента, запросы клиентов больше не приходят на частный почтовый ящик Максима, теперь они направляются на почтовый ящик, который используется для OTRS. Система обработки заявок (OTRS) подключена к этому почтовому ящику и автоматически сохраняет все запросы в своей базе данных. Система генерирует авто-ответ для каждого нового запроса и отправляет его клиенту, так что теперь клиент может быть уверен в том, что его запрос поступил в обработку и ответ будет дан в ближайшее время. Для каждого запроса OTRS создает прямую ссылку - номер заявки. Теперь клиенты довольны, они получают быстрый ответ на свою заявку и им уже нет никакого смысла писать повторно письмо с тем же вопросом. Максим, Андрей и Александр могут спокойно отвечать на поступившие запросы используя обычный веб-браузер для работы с OTRS. Также, следует отметить тот факт, что ни одно сообщение клиента не будет отредактировано дважды, поскольку система автоматически блокирует заявку, для которой создается ответ.

Давайте представим что г-н Гришко делает запрос в компанию Максима и его сообщение обрабатывается системой OTRS. Александр дает краткий ответ на его вопрос. Но у г-на Гришко есть дополнительный вопрос и на следующий день он отвечает на письмо Александра. Поскольку у Александра и своих дел хватает, на письмо г-на Гришко отвечает Максим. Способность OTRS хранить историю позволяет Максиму просмотреть всю цепочку переписки по этому запросу и написать более подробный ответ г-ну Гришко. Г-н Гришко, в свою очередь, даже не догадывается, что в процесс общения были вовлечены два разных человека, он доволен советами

относительно решения своей проблемы, которые пришли в последнем сообщении от Максима.

Конечно, это всего лишь краткий обзор функциональных возможностей и особенностей систем обработки заявок. Но в случае, если в вашу компанию приходит множество запросов от клиентов через письма электронной почты и телефонные звонки и они должны обрабатываться в разное время, - система обработки заявок поможет существенно сэкономить время и деньги, позволит структурировать всю систему поддержки клиентов, сделать более прозрачным процесс обмена информацией между клиентами и компаниями, что неотменно приведет к повышению эффективности как для клиентов так и для обслуживающего персонала и позитивно скажется на финансовом положении компании.

2. Что такое заявка?

Заявка очень похожа на медицинскую карточку пациента больницы. Медицинская карточка создается, когда пациент посещает больницу впервые, все важные данные о пациенте, такие как: личные данные, информация о состоянии его здоровья, медицинском осмотре записываются в медицинскую карточку. С каждым новым визитом пациента в больницу, каждый из лечащих врачей добавляет в медицинскую карточку информацию о состоянии пациента, истории болезни и, используемые для лечения лекарственные препараты. Таким образом другие лечащие врачи и медсестры могут видеть, подробную картину состояния пациента. Когда пациент выздоровел и выписался из больницы, медицинская карточка закрывается и вся информация передается в архив.

Системы обработки заявок, такие как OTRS, обрабатывают заявки как обыкновенные емейлы (email). Когда клиент посылает запрос, система создает новую заявку, что сравнимо с открытием новой медицинской карточки пациента больницы. Ответ в этой новой заявке можно сравнить с записью лечащего врача в медицинской карточке, свидетельствующей о изменении состояния здоровья пациента. Заявка считается закрытой, когда клиенту отправлен ответ или если заявка закрыта системой автоматически. Если клиент отправляет ответ на уже закрытую заявку, она будет заново открыта системой и в нее будет добавлена новая информация. Чтобы сохранить целостность всех данных, хранящихся в системе, каждая заявка хранится и архивируется вместе со всей информацией. Поскольку заявки обрабатываются как обыкновенные сообщения электронной почты - вместе с ними также будут сохранены все вложения и контекстные аннотации. Конечно же, такая полезная информация как примечания, даты, данные о нанятых работниках, рабочее время необходимое для работы с заявкой и т.д также сохраняется системой. Все заявки могут быть отсортированы в соответствующем порядке, также можно с легкостью производить поиск всей необходимой информации.

Глава 2. OTRS Help Desk

This chapter describes the features of OTRS Help Desk (OTRS). You will find information about the hardware and software requirements for OTRS. Additionally, this chapter tells you how to get commercial support for OTRS, should you require it, and how to contact the community.

1. Основы

OTRS Help Desk (OTRS) is a web application which is installed on a web server and can be used with a web browser.

OTRS is separated into several components. The basic component is the OTRS framework that contains all central functions for the application and the ticket system. Via the web interface of the OTRS framework, it is possible to install additional applications such as ITSM modules, integrations with Network Monitoring solutions, a knowledge base (FAQ), et cetera.

2. Особенности и функциональные возможности

OTRS обладает широким набором функций. В следующем списке приводится обзор функциональных возможностей основного компонента - OTRS-фреймворка.

Особенности и функциональные возможности OTRS

- Веб-интерфейс:
 - Easy and initial handling with any modern web browser, even with mobile phones or other mobile computers.
 - Существует возможность администрирования системы через веб-интерфейс.
 - Существует встроенный веб-интерфейс для обработки клиентских запросов сотрудниками/агентами системы.
 - Клиентам доступен веб-интерфейс для создания новых заявок, проверки состояния существующих, написания ответов на старые заявки а также поиска их собственных заявок.
 - В веб-интерфейсе предусмотрена поддержка различных шаблонов (тем); также существует возможность интеграции своих собственных шаблонов.
 - Поддержка многих языков.
 - Настраиваемые шаблоны вывода (DTL).
 - Входящие и исходящие сообщения электронной почты могут содержать многочисленные вложения.
- Почтовый интерфейс:
 - Поддержка почтовых вложений (поддержка MIME).
 - Автоматическое преобразование HTML в обычные текстовые сообщения (это сделано для улучшения безопасности в случае получения сообщений с потенциально опасным содержанием, а также для поддержки более быстрого поиска).
 - Почтовые сообщения могут быть отфильтрованы несколькими способами: с помощью системных X-OTRS-заголовков или используя различные почтовые ящики, например для спам-сообщений.

- Поддержка PGP, создание и импорт собственных ключей, подписка и шифрование исходящей почты, возможность просмотра подписанных и зашифрованных сообщений.
- Поддерживается просмотр и шифрование S/MIME-сообщений, обработка S/MIME-сертификатов.
- Автоматические ответы (автоответы) на сообщения клиентов, настраиваемые для каждой очереди.
- Уведомления агентов по электронной почте о новых, последующих или разблокированных заявках.
- Follow-ups by references or In-Reply-To header entries.
- Заявки:
 - Удобный шаблон для просмотра информации в очереди, беглый обзор новых заявок в очереди.
 - Заявки могут быть заблокированы.
 - Создание собственных шаблонов автоответов.
 - Создание собственных автоответчиков, настраиваемых для каждой очереди.
 - История заявки, просмотр всех событий определенной заявки (изменения состояний заявки, ответов, заметок и т.д.).
 - Шаблон "Версия для печати" для заявок.
 - Добавление собственных (внутренних или внешних) примечаний к заявке (текста и вложений).
 - Просмотр подробной информации о заявке.
 - Для заявок можно определять списки контроля доступа (ACL-Access Control List).
 - Пересылка заявок на другие почтовые адреса.
 - Перемещение заявок между очередями.
 - Установка/Изменение приоритета заявки.
 - Подсчет рабочего времени, необходимого для работы с каждой заявкой.
 - Для заявки можно определить предстоящие задачи.
 - Одно и то же действие можно совершать одновременно над несколькими заявками.
 - "GenericAgent" позволяет совершать автоматические и синхронные действия над заявками.
 - Полнотекстовый поиск по всем заявкам и экспорт результатов в PDF.
- Система:
 - OTRS работает во многих операционных системах (Linux, Solaris, AIX, FreeBSD, OpenBSD, Mac OS 10.x, Microsoft Windows).

- Поддержка ASP (поддержка активных служб).
- Предусмотрена возможность связывания нескольких объектов, например заявок и записей в FAQ-системе.
- Для хранения данных о клиентах предусмотрена возможность интеграции внешних источников данных, используя AD, eDirectory или OpenLDAP.
- Установка собственных идентификаторов заявки, например Звонок№, Заявка№ или Запрос№.
- Интеграция своего собственного счетчика заявок.
- Support of several database systems for the central OTRS back-end, e.g. MySQL, PostgreSQL, Oracle, MSSQL).
- Фреймворк для создания статистики.
- Полная поддержка utf-8 (на уровне всей системы).
- Аутентификация пользователей (агентов и клиентов) может быть реализована с помощью базы данных, LDAP, HTTP Auth или Radius.
- Поддержка учетных записей пользователей, групп пользователей и ролей.
- Поддержка различных уровней доступа к отдельным компонентам системы или очередям.
- Интеграция стандартных текстов ответа.
- Поддержка под-очереди.
- Для каждой очереди можно определить собственные фразы приветствия и подписи.
- Уведомления администраторов по электронной почте.
- Получение информации об обновлениях по электронной почте или через веб-интерфейс.
- Эскалация заявок.
- Поддержка различных часовых поясов.
- Простая интеграция собственных дополнений и приложений с помощью OTRS API.
- Простое создание собственных пользовательских интерфейсов, как это например делается в графической оконной с-ме X11 (X-Window) операционной системы UNIX, консоли и т.п.

2.1. New features of OTRS 3.1

2.1.1. GENERIC INTERFACE - A Web Service Framework

- GI is a flexible framework to allow web service interconnections of OTRS with third party applications.
- OTRS can act in both ways - as a provider (server, requested from remote) or requester (client, requesting remotely).

- Simple web service connections can be created without programming by configuring the Generic Interface.
- Complex scenarios can be realized by plugging in custom OTRS extensions that add perl code to the GI infrastructure on different architectural layers.
- *Connectors* expose parts of OTRS to Generic Interface web services. For example, a ticket connector exposes the ticket create/update function, so that they can be used in a web service regardless which network transport is used.
- A scheduler daemon process supports asynchronous event handling. This is useful to asynchronously start web service requests from OTRS to another system, after the agents request has been answered (e.g. when a ticket has been created). Otherwise, it might block the response, resulting in increased response times for the agent.

With the Generic Interface new web services can be configured easily by using existing OTRS modules, without additional code. They can be combined to create a new web service. When configuring a new web service connection, the administrator has to add:

- A new web service in the admin GUI
- The basic meta data (Transport type (SOAP), URL etc.) and
- Existing operations (part of a connector) and specify for each operation how the data must be mapped (inbound and outbound)

A Generic Interface Debugger will help the OTRS administrator to check how requests are coming in and how they are handled through the different layers.

2.1.1.1. Current Features

- Network transports: SOAP/HTTP. Others like REST and JSON are scheduled to be added in the future depending on customers demand.
- Configurable data mapping Graphical User Interface for key/value transformations with respect to incoming and outgoing data.
- Graphical debugger to check the configuration and flow of information of configured web services.
- A ticket connector allowing the use of OTRS as a web service for ticket handling.

2.1.1.2. Future Features

- Additional network transports (REST, JSON).
- The GI will replace the iPhoneHandle as the backend for mobile apps.
- Additional connectors will be added to provide more parts of OTRS for use with web services (e.g. to allow the creation, update or deletion of agents, users, services or CIs).

2.1.2. DYNAMIC FIELDS

The DynamicFields Feature replaces the existing ticket and article FreeText and FreeTime fields with a dynamic structure that will also allow to create custom forms in OTRS.

- An unlimited amount of fields can be configured using an own graphical user interface for administration.
- The fields can have different types that can be used for both, tickets and articles. Available by default are:
 - Text

- Multiline text
- Checkbox
- Dropdown
- Multi-select
- Date
- Date and time
- New custom field types (e.g. custom field type dropdown with an external data source) can be added with small effort as the fields are created in a modular, pluggable way.
- A future scenario is, that DynamicFields can be used for objects other than tickets or in custom modules. For example, a custom module adding objects to handle "orders" in OTRS could use the DynamicFields to attach properties/data to these orders.
- A database update script will transform historic FreeText fields and related configuration settings into the new structure.

2.1.3. TICKET MANAGEMENT IMPROVEMENTS

2.1.3.1. Ticket creation improved

- Multiple email addresses can now be specified as 'To:', 'CC:' or 'BCC:' when creating a new phone or email ticket.

2.1.3.2. Inbound phone call support

- Inbound phone calls can now be registered within an existing tickets (until now, only outbound calls were registered).

2.1.3.3. Ticket overview preview improved

- It is now possible to exclude articles of certain sender types (e.g. articles from internal agents) in the SysConfig from being displayed in the overview preview mode.
- A certain article type can be configured which will display articles of that type as expanded by default when the view is accessed.

2.1.3.4. Ticket move improved

- The screen shown after moving a ticket is now configurable. Options are the ticket zoom view (LastScreenView) or the ticket list (LastScreenOverview).

2.1.3.5. Bulk action improved

- With the new bulk action, outbound emails can now be sent from multiple tickets at the same time. As tickets can have different queues, and these queues each can have different templates, salutations and signatures, these are not used in the Bulk Action email.
- An additional bulk action allows configuring the ticket type for selected tickets.

2.1.3.6. Configurable Reject Sender Email Address

- The feature allows configuring an email address instead of the administrator address to reject the creation of new tickets by email. This feature can be used in all cases where customers are not allowed to create new tickets by email.

2.1.4. PROCESS AUTOMATION

2.1.4.1. Escalation events added

- OTRS will now create events for each of the available escalation types (response, update and resolution). This allows performing actions (such as notifications) before the escalation occurs, in the moment it occurs and in the moment that the escalation ends.

2.1.4.2. Notification mechanism improved

- A new generic agent notification module allows the OTRS administrator to define messages that will be shown in the agent web front-end when agents log into the system.

2.1.4.3. Time calculation improved

- All kind of times are from now on calculated by and based on the application server only solving the issues that were caused by variances between the clock times of application and data base servers.

2.1.4.4. GenericAgent improved

- The GenericAgent can now filter for tickets change time.
- In addition, the GenericAgent can set the ticket responsible for matched tickets.

2.1.5. USER INTERFACE, RICH TEXT EDITOR, CHARSET

2.1.5.1. User interface performance improved

- The speed for rendering and article display was improved, thanks to Stelios Gikas <stelios.gikas@noris.net>!

2.1.5.2. Rich Text Editor Update

- IOS5 support added.
- Block quotes can be left with the enter key.
- Update from CKEditor 3.4 to CKEditor 3.6, so improvements refer to the releases of [CKEditor 3.5](#) and [CKEditor 3.6](#).
- IE9 support improved.
- Resizable dialogs.

2.1.5.3. Unicode Support - Non-UTF-8 Internal Encodings Dropped

- UTF-8 is now the only allowed internal charset of OTRS.
- All language files are now formatted in UTF-8, which simplifies their handling and future improvements of the translation mechanism.

2.1.6. DATABASE DRIVER SUPPORT

2.1.6.1. PostgreSQL DRIVER compatibility improved

- PostgreSQL 9.1 support added.
- A new legacy driver is now available for PostgreSQL 8.1 or earlier versions.

2.1.6.2. MS SQL DRIVER compatibility improved

- The MS SQL driver now stores binary data in VARBINARY rather than deprecated type TEXT as well as NVARCHAR to store text strings rather than VARCHAR (for improved Unicode support).

2.1.7. MAIL INTEGRATION

2.1.7.1. Mail handling improved

- When connecting to IMAP mail accounts, it is now possible to handle emails from a specific email folder, other than the INBOX folder.
- OTRS can now also connect to IMAP servers using Transport Layer Security (TLS), useful for modern restricted environments.

2.2. Новые возможности OTRS 3.0

Контекст

- Существенно переработан графический интерфейс пользователя. Теперь, благодаря переходу к более мощным и современным технологиям, таким как Ajax, XHTML, CSS приложение стало более динамическим и ориентированным на конечного пользователя.

Новый индикатор оповещения о новой Заявке/Статье.

- Эта новая функция была реализована для заявок и статей. Это позволяет агенту сразу же увидеть, есть ли изменения в определенной заявке или статье, проверить новые и непрочитанные статьи. Вы получаете выгоду от повышения прозрачности и снижения времени отклика.

Оптимизированный Полнотекстовый Поиск

- Новая функция поиска позволяет более гибко производить поиск по имеющейся информации. Опции нового поиска позволяют задать диапазон как в виде простой строки так и в виде сложных многострочных логических операций, включая различные операторы. Таким образом вы получаете полностью настраиваемый поиск, способный удовлетворить потребности самого требовательного пользователя.

Новый шаблон для просмотра подробной информации о заявке.

- Редизайн, базирующийся на технологии Ajax предоставляет агентам возможность в режиме реального времени просматривать сложные, связанные между собой структуры данных, при этом сохраняя без изменений текущую рабочую среду. Выиграш от этого нововведения заключается в повышении удобства пользования системой в целом и увеличении эффективности всего рабочего процесса.

Обзор заявок

- Еще в OTRS 2.4 для повышения интерактивности приложения был оптимизирован способ отображения заявок. В зависимости от вариантов использования и предпочтений ваших агентов, они могут с легкостью изменить способ отображения заявок в соответствии с их собственными потребностями. При отображении информации возможны варианты с различным уровнем детализации: минимальный, средний, максимальный.

Accessibility

- В процессе редизайна системы, во внимание были приняты признанные во всем мире стандарты WCAG и WAI-ARIA, которые позволяют пользователям с ограниченными возможностями, лучше взаимодействовать со службой технической поддержки OTRS. Требования раздела 508, Закона США о Реабилитации (датованный 1973 г) также поддерживаются системой.

Новый Пользовательский Интерфейс

- Пользовательский веб-интерфейс можно интегрировать в локальную сеть (интранет) вашей организации и полностью интегрировать с реструктурированной службой поддержки клиентов.

Функция Архивирования

- OTRS 3.0 предлагает новую функцию архивации, которая существенно сокращает время поисковых запросов и ускоряет вывод результатов поиска.

2.3. Новые возможности OTRS 2.4

Переход на лицензионное соглашение AGPL третьей версии (AGPL v.3)

- Почему AGPL вместо GPL? - AGPL и GPL идентичны, за одним единственным исключением: для SaaS-решений (SaaS - решения, которые используют сервис-ориентированную архитектуру) "Авторское Лево" (Авторское Право) реализовано более эффективно именно в AGPL - а не в GPL. Принимая во внимание растущий мир SaaS-решений, ((otrs)) стремится к тому, чтобы будущие разработки и в дальнейшем возвращались к сообществу OTRS. Это является главной причиной перехода на AGPL.

Почему третья версия (v.3) вместо второй? - вторая версия GPL "морально устаревает" и, особенно в США, имеет различные правовые неопределенности. По мнению ((otrs)) GPL v.3 сохранила в себе "дух" GPL v.2 и в то же время была создана с учетом новых потребностей. ((otrs)) рассматривает GPL v.3, а особенно AGPL v.3, как наиболее сбалансированную Открытую "Копилефт" Лицензию среди всех существующих на сегодняшний день, предлагающую Защиту авторских прав владельцев и пользователей и обеспечивающую лучшую правовую безопасность без нарушений законодательства.

Новая Панель Управления

- Интегрированная Панель Управления была создана в связи с необходимостью персонализированного представления полезной информации на системном уровне и в режиме реального времени. Также существует возможность создавать подключаемые модули для отображения контента из собственных расширений наряду с отображением стандартного контента. Стандартные плагины:
- Заявки (новые и открытые) за последние 24, 48 и 72 часа.
- Календарь, включающий краткий обзор предстоящих событий (эскалации, автоматические разблокировки, и т.п.)
- Общесистемный обзор распределения заявок внутри очередей
- Время первого ответа (Время/Решение)
- Интеграция RSS

Новый Стандарт Отчетов

- OTRS 2.4 предоставляет новые отчеты:
- Созданные Заявки
- Закрытые Заявки
- SLA Анализ
- Время, которое необходимо выделить на одного пользователя/очередь
- Анализ времени решения требуемого на одного пользователя/очередь
- Анализ времени отклика требуемого на одного клиента/очередь

Новая функция "Ведущая/Ведомая Заявка"

- Функция "Ведущая/Ведомая Заявка" позволяет связать воедино несколько похожих заявок, и обрабатывать их вместе. Как только проблема будет решена, закрывается только "Ведущая" заявка, все остальные (ведомые) будут закрыты системой автоматически, а текст, с решением проблемы для "Ведущей" заявки будет разослан всем владельцам "Ведомых" заявок.

Доступен новый тип ссылок "Ведомая". Все заявки этого типа унаследуют действия от своей "Ведущей" заявки:

- Изменение состояния
- Email-ответы
- Изменения в полях FreeText
- Примечания
- Задержка изменения
- Изменения приоритета
- Изменение владельца
- Изменение обязательств

Электронная почта поддерживает "обогащенный текст"/HTML (WYSIWYG)

- Теперь, благодаря этой функции, появилась возможность писать письма, заметки и уведомления в расширенном (обогащенном) текстовом формате (HTML-формате). Используя WYSIWYG-редактор (Что видишь, То и Получаешь), можно с легкостью писать тексты, форматировать их и даже встраивать изображения.

Новая функция "Вне офиса"

- Благодаря новой функции "Вне офиса" все пользователи системы имеют возможность поставить в известность своих коллег (в т.ч. саму систему OTRS) о времени их отсутствия в офисе. Функция "Вне офиса" активна в период времени, установленный пользователем. Активация этой функции имеет следующий эффект:

В списках выбора, которые используются для назначения владельца или ответственного участника, время отсутствия и его продолжительность отображаются за именем пользователя.

Если агенту, в период его отсутствия, приходит новый отклик от клиента, - то заявка разблокируется автоматически, а всем агентам из этой очереди отсылается уведомление. Это позволяет немедленно реагировать на вопросы клиента другим сотрудникам службы технической поддержки.

Новые шаблоны отображения заявок с возможностью выполнения групповых действий

- Гибкость и удобство представления (подачи) информации является обязательным условием. С помощью трех шаблонов отображения заявок: Упрощенный/Обычный/Расширенный, каждый агент имеет возможность простым щелчком мыши (по соответствующей иконке), фактически "на лету", изменить способ отображения информации (отображение очередей, статусов т.д.). Такая гибкость предоставляет высокий уровень индивидуализации и адаптации пользователей к любой оперативной ситуации.

Функция групповых операций интегрирована во все шаблоны отображения заявок (напомним, что групповые действия позволяют обрабатывать несколько заявок одновременно).

Почтовый Фильтр автоматически распознает поступающие сообщения в случае внутренних пересылок

- В настоящее время, email-ответы на перенаправленные статьи поступают в OTRS как внешние емейлы. Проблема заключается в том, что клиенты могут просматривать ответы на эти статьи используя графический веб-интерфейс пользователя. Несмотря на то, что все же есть возможность классифицировать емейлы приходящие из всего домена, и обрабатывать их как внутренние - это только откладывает проблему в "долгий ящик", не решая ее. Кроме этого, такой шаг не позволяет должным образом обслуживать клиентов (определенного домена), так как клиент больше не имеет возможности отслеживать заявки через свой пользовательский интерфейс. Используя эту новую функцию можно отследить email-ответы, а также, базируясь на типе оригинальной статьи, автоматически определить внутренние/внешние емейлы.

Настраиваемые уведомления, базирующиеся на событиях

- Для достижения большей эффективности и гибкости системы уведомлений, механизм рассылки был тщательно изучен и заново переработан. Новая система позволяет обмениваться сообщениями между клиентами и агентами, или удаленными почтовыми адресами на основе происходящих событий.

Таким образом, клиенту можно просто сообщить о закрытии заявки. Или, например, когда VIP-клиент создает заявку, сообщение может быть отправлено на определенный адрес. События (т.е. Создание Заявки, Изменение Состояния Заявки, Изменение Приоритета Заявки, Создание Статьи) и все известные переменные сообщений (такие как, например, <OTRS_TICKET_TicketNumber> <OTRS_TICKET_Priority>), можно легко выбирать для создания сообщений через веб-интерфейс.

Права "Только Для Чтения" и Предупреждения для просматриваемых Заявок

- В текущем релизе OTRS-пользователям предоставлена новая возможность создавать свои собственные Списки Просматриваемых Заявок. Эта функция распространяется на заявки, помеченные пользователем как "подписанные". Выигрыш от такого нововведения заключается в том, что пользователи больше не будут те-

рять свои списки заявок, а смогут всегда "держать их под рукой", просматривая отдельным списком. До этого времени функция "Только Для Чтения" позволяла агентам просматривать заявки, на которые они подписались, только в том случае, если они находились в очереди, для которой агент имеет права "Только для Чтения". С этого момента, функция "Только для чтения" позволяет агентам всегда иметь права "только для чтения" для заявок, на которые они подписались, даже в том случае, если заявка будет перемещена в другую очередь, к которой у них нету доступа. Используя меню "Персональные настройки", каждый агент может настроить функцию "Оповещения", для определения того, следует ли получать уведомления о заявках, которые получает владелец или ответственный участник. Это позволяет активно отслеживать наблюдаемые заявки.

Безопасный SMTP

- OTRS может получать и отправлять письма несколькими способами. В OTRS 2.3 были реализованы все доступные на сегодня способы получения писем (POP3, POP3S, IMAP, IMAPS). До сих пор было только два варианта для отправки электронной: используя локальный Почтовый Агент (Mail Transfer Agent, например Sendmail, Postfix и т. д.) или используя SMTP. SMTPS (Безопасный SMTP) был внедрен в OTRS 2.4.x с целью поддержки растущих стандартов безопасности.

3. Требования к аппаратному и программному обеспечению

OTRS can be installed on many different operating systems. OTRS can run on linux and on other unix derivatives (e.g. OpenBSD or FreeBSD). You can also deploy it on Microsoft Windows. OTRS does not have excessive hardware requirements. We recommend using a machine with at least a 2 GHz Xeon or comparable CPU, 2 GB RAM and a 160 GB hard drive for a small setup.

To run OTRS you'll also need to use a web server and a database server. Apart from that, on the OTRS machine, you should install perl and/or install some additional perl modules. The web server and Perl have to be installed on the same machine as OTRS. The database back-end can be installed locally or on another host.

For the web server we recommend using the Apache HTTP Server, because its module mod_perl improves greatly the performance of OTRS. Apart from that, OTRS should run on any web server that can execute Perl scripts.

You can deploy OTRS on different databases. You can choose between MySQL, PostgreSQL, Oracle, or Microsoft SQL Server. If you use MySQL you have the advantage that the database and some system settings can be configured during the installation, through a web front-end.

Для Perl, мы рекомендуем использовать по крайней мере версию 5.8.8. Также вам будут нужны дополнительные модули, которые могут быть установлены как при установке самого Perl и CPAN так и с помощью менеджера пакетов вашей операционной системы (rpm, yast, apt-get).

Требования к программному обеспечению

3.1. Perl support

- Perl 5.8.8 или выше

3.2. Web server support

- Apache2 + mod_perl2 или выше (рекомендуется, mod_perl он действительно быстрее!)
- Веб-сервер с поддержкой CGI-интерфейса (CGI-интерфейс не рекомендуется)

- Microsoft Internet Information Server (IIS) 6 or higher

3.3. Database support

- MySQL 4.1 или выше
- PostgreSQL 7.0 or higher (8.2 or higher recommended)
- Oracle 10g или выше
- Microsoft SQL Server 2005 or higher

The section in the manual about installation of Perl modules describes in more detail how you can set up those which are needed for OTRS.

Если для установки OTRS используются бинарные пакеты, собранные специально под вашу операционную систему, то все необходимые модули языка Perl уже включены. Если нет, - используйте менеджер пакетов вашей операционной системы (например rpm, Windows-Инсталлятор) для установки всех необходимых модулей.

3.4. Web browser support

For the Agent interface of OTRS, you'll be OK if you use a modern browser with JavaScript support enabled. We support the following browsers:

- Internet Explorer 8.0 or higher
- Mozilla Firefox 3.6 or higher
- Google Chrome
- Opera 10 or higher
- Safari 4 or higher

We recommend to always use the latest version of your browser, because it has the best JavaScript and rendering performance. Dramatical performance varieties between the used browsers can occur with big data or big systems. We are happy to consult you on that matter.

For the OTRS Customer Interface, in addition to the browsers listed above, you can also use Internet Explorer versions 6 or 7, and we do not require JavaScript either.

4. Комюнити

OTRS has a large user community. Users and developers discuss about OTRS and interchange information on related issues through the mailing-lists. You can use the mailing lists to discuss installation, configuration, usage, localization and development of OTRS. You can report software bugs in our bug tracking system.

The homepage of the OTRS community is: <http://www.otrs.com/open-source/>.

5. Коммерческая Поддержка и Услуги для OTRS

Commercial support for OTRS is also available. You can find the available options on the website of OTRS Group, the company behind OTRS: <http://www.otrs.com/>.

OTRS Group provides subscription support services, customization, consulting and training for *OTRS Help Desk* and *OTRS ITSM*. It also provides *Best Practice Product Editions*. With these Editions, OTRS Group helps organizations to design, deploy and optimize OTRS for each unique environment. Additionally, OTRS Group provides hosted versions including *OTRS OnDemand* and *Managed OTRS*.

You can find more detailed information about OTRS Group on <http://www.otrs.com> and you can contact us via email on sales@otrs.com.



Глава 3. Установка

В этой главе описывается установка и базовая конфигурация главного OTRS-фреймворка. Здесь вы найдете информацию об установке OTRS из исходных кодов или из бинарных пакетов, например RPM, или с помощью Windows-инсталлятора.

В этой главе рассматриваются такие вопросы как: конфигурация веб-сервера и сервера базы данных, интерфейса между OTRS и базой данных, установка дополнительных Perl-модулей, установка соответствующих прав доступа для OTRS, настройка планировщика задач cron jobs для OTRS и основных параметров в конфигурационных файлах.

Следуйте подробным шагам этой главы чтобы установить OTRS на своем сервере. Потом можно использовать его веб-интерфейс чтобы войти в систему и производить администрирование.

1. Самый простой способ - установка из готовых бинарных пакетов

Самый простой и удобный способ установки OTRS - использовать готовые (встроенные) пакеты, которые можно найти и загрузить перейдя по ссылке <http://www.otrs.org>. В следующих разделах описывается установка OTRS из готовых (встроенных) или бинарных пакетов, специально собранных для операционных систем: SUSE, Debian и Microsoft Windows. В случае если нету возможности использовать готовые (встроенные) пакеты - установку придется производить вручную.

1.1. Установка из rpm-пакетов на сервер под управлением Suse Linux

This section demonstrates the installation of a pre-built RPM package on a SUSE Linux distro. We have tested against all recent SLES and openSUSE versions. Before you start the installation, please have a look at <http://www.otrs.com/try/> and check if a newer OTRS RPM package is available. Always use the latest RPM package.

Для установки OTRS используйте конфигурационную утилиту yast (yast2), командную строку **rpm**. Так как для работы OTRS необходимы Perl-модули, которые по умолчанию не устанавливаются в системе SUSE, мы всегда рекомендуем использовать менеджер пакетов yast, поскольку он позволяет автоматически разрешать конфликты и зависимости между пакетами.

If you decide to install OTRS via the command line and **rpm**, first you have to manually install the needed Perl modules. Assuming you saved the file `otrs.rpm` in the directory /tmp, you can execute the command specified in the following script to install OTRS.

```
linux:~ # rpm -i /tmp/otrs-xxx.rpm
otrs
Check OTRS user (/etc/passwd)... otrs exists.

Next steps:

[SuSEconfig]
Execute 'SuSEconfig' to configure the web server.

[start Apache and MySQL]
Execute 'rcapache2 start' and 'rcmysql start' in case they don't run.

[install the OTRS database]
Use a web browser and open this link:
http://localhost/otrs/installer.pl

[OTRS services]
Start OTRS 'rcotrs start-force' (rcotrs {start|stop|status|restart|start-force|stop-force}).
```



```
Have fun!  
  
Your OTRS Team  
http://otrs.org/  
  
linux:~ #
```

Script: Command to install OTRS.

After the installation of the OTRS RPM package, you have to run SuSEconfig, as shown in the following script.

```
linux:~ # SuSEconfig  
Starting SuSEconfig, the SuSE Configuration Tool...  
Running in full featured mode.  
Reading /etc/sysconfig and updating the system...  
Executing /sbin/conf.d/SuSEconfig.aaa_at_first...  
Executing /sbin/conf.d/SuSEconfig.apache...  
Including /opt/otrs/scripts/apache-httpd.include.conf  
Executing /sbin/conf.d/SuSEconfig.bootsplash...  
Executing /sbin/conf.d/SuSEconfig.doublecheck...  
Executing /sbin/conf.d/SuSEconfig.guile...  
Executing /sbin/conf.d/SuSEconfig.hostname...  
Executing /sbin/conf.d/SuSEconfig.ispell...  
Executing /sbin/conf.d/SuSEconfig.perl...  
Executing /sbin/conf.d/SuSEconfig.permissions...  
Executing /sbin/conf.d/SuSEconfig.postfix...  
Setting up postfix local as MDA...  
Setting SPAM protection to "off"...  
Executing /sbin/conf.d/SuSEconfig.profiles...  
Finished.  
linux:~ #
```

Script: Running the SuSEconfig command.

The OTRS installation is done. Restart your web server to load the OTRS specific changes in its configuration, as shown in the script below.

```
linux:~ # rcapache2 restart  
Shutting down httpd done  
Starting httpd [ PERL ] done  
linux:~ #
```

Script: Restarting the web server.

Следующим шагом является установка базы данных OTRS, как это сделать, читайте здесь раздел 3.2.4.

1.2. Установка OTRS в операционной системе CentOS

On the OTRS Wiki you can find detailed instructions for setting up OTRS on a CentOS system. Please note that these instructions will also apply to Red Hat Linux systems since they use the same source: http://wiki.otrs.org/index.php?title=Installation_of_OTRS_3.0b1_on_CentOS_5.5 .

1.3. Установка OTRS в операционной системе Debian

Подробную информацию по установке OTRS в операционной системе Debian можно найти на сайте OTRS Wiki по адресу http://wiki.otrs.org/index.php?title=Installation_on_Debian_5.04_lenny .

1.4. Установка OTRS на системе Ubuntu

Подробную информацию по установке OTRS в операционной системе Ubuntu можно найти на сайте OTRS Wiki по адресу [http://wiki.otrs.org/index.php?title=Installation_on_Ubuntu_Lucid_Lynx_\(10.4\)](http://wiki.otrs.org/index.php?title=Installation_on_Ubuntu_Lucid_Lynx_(10.4)) ..

1.5. Установка OTRS на операционной системе Microsoft Windows

Устанавливать OTRS в операционной системе Microsoft Windows очень легко и просто. Перейдя по ссылке <http://www.otrs.com/try/> загрузите последнюю версию установщика для Win32-платформы и сохраните файл на жестком диске компьютера. Затем просто запустите установочный файл и выполните все шаги, предложенные мастером установки. После этого можно войти в OTRS с правами администратора и сконфигурировать систему под свои требования. Используйте логин `root@localhost` и пароль `root`, чтобы войти в систему с правами администратора. (Примечание: логин `root@localhost` и пароль `root` прописаны в системе по умолчанию).

Предупреждение

Постарайтесь как можно быстрее изменить пароль для аккаунта `'root@localhost'`.

Важно

Win32-инсталлятор содержит все компоненты, необходимые для работы OTRS: веб-сервер Apache, сервер баз данных MySQL, Perl (со всеми необходимыми модулями), а также планировщик задач для WindowsCron. Именно по этой причине, OTRS необходимо устанавливать на Windows-системах, которые еще не содержат установленного веб-сервера Apache или другого, а также сервера баз данных MySQL.

2. Установка из исходных кодов (Linux, Unix)

2.1. Подготовка к установке из исходных кодов

Если вы решили устанавливать OTRS из исходных кодов, перейдите по ссылке <http://www.otrs.com/try/> и загрузите архив с исходными кодами в любом удобном для вас формате: `.tar.gz`, `.tar.bz2`, или `.zip`

Unpack the archive (for example, using `tar`) into the directory `/opt`, and rename the directory from `otrs-3.1.x` to `otrs` (see Script below).

```
linux:/opt# tar xf /tmp/otrs-3.1.tar.gz
linux:/opt# mv otrs-3.1 otrs
linux:/opt# ls
otrs
linux:/opt#
```

Script: First steps to install OTRS.

OTRS should NOT be run with root rights. You should add a new user for OTRS as the next step. The home directory of this new user should be `/opt/otrs`. If your web server is not running with the same user rights as the new `'otrs'` user, which is the case on most systems, you have to add the new `'otrs'` user to the group of the web server user (see Script below).

```
linux:/opt# useradd -r -d /opt/otrs/ -c 'OTRS user' otrs
linux:/opt# usermod -G nogroup otrs
linux:/opt#
```

Script: Adding a new user 'otrs', and adding it to a group.

Next, you have to copy some sample configuration files. The system will later use the copied files. The files are located in `/opt/otrs/Kernel` and `/opt/otrs/Kernel/Config` and have the suffix `.dist` (see Script below).

```
linux:/opt# cd otrs/Kernel/
linux:/opt/otrs/Kernel# cp Config.pm.dist Config.pm
linux:/opt/otrs/Kernel# cd Config
linux:/opt/otrs/Kernel/Config# cp GenericAgent.pm.dist GenericAgent.pm
```

Script: Copying some sample files.

На завершающем этапе установки OTRS необходимо установить соответствующие права доступа к файлам. Для этого используйте сценарий **otrs.SetPermissions.pl**, находящийся в директории bin домашнего каталога пользователя 'otrs'. Скрипт можно вызвать со следующими параметрами:

```
otrs.SetPermissions.pl {Home directory of the OTRS user} { --otrs-user= OTRS user} { --web-user= Web server user} [ --otrs-group= Group of the OTRS user] [ --web-group= Group of the web server user]
```

Если ваш веб-сервер работает с теми же правами что и пользователь 'otrs', то команда установки надлежащих прав доступа будет выглядеть так: **otrs.SetPermissions.pl /opt/otrs --otrs-user=otrs --web-user=otrs**. На SUSE-системах веб-сервер работает с правами пользователя 'wwwrun'. На Debian-системах - 'www-data'. Для установки надлежащих прав доступа используйте команду **otrs.SetPermissions.pl /opt/otrs --otrs-user=otrs --web-user=wwwrun --otrs-group=nogroup --web-group=www**.

2.2. Установка Perl-модулей

Исходя из Таблицы 3-1. для работы OTRS необходимо установить некоторые дополнительные модули Perl-. При установке OTRS из исходных кодов, эти модули придется установить вручную. Конечно же это намного проще сделать используя менеджер пакетов, который прилагается к вашему Linux-дистрибутиву (yast, apt-get) или, как описано в этом разделе, использовать оболочку Perl shell и CPAN. Если вы используете ActiveState Perl, например, на Windows, то можно использовать PPM, встроенный менеджер пакетов Perl (Perl Package Manager). Мы рекомендуем использовать менеджер пакетов, если это возможно.

Таблица 3.1. Perl-модули, необходимые для работы OTRS

Название	Описание
DBI	Устанавливает подключение к базе данных приложения.
DBD::mysql	Содержит специальные функции для подключения к серверу базы данных MySQL (только в случае, использования сервера базы данных MySQL)
DBD::pg	Содержит специальные функции для подключения к серверу базы данных PostgreSQL (требуется только в случае использования PostgreSQL в качестве сервера базы данных).
Digest::MD5	Позволяет использовать алгоритм MD5.
CSS::Minifier	Уменьшение размера CSS-файла и запись выходного потока напрямую в другой файл.
Crypt::PasswdMD5	Обеспечение криптографических возможностей на основе алгоритма хеширования MD5

Название	Описание
MIME::Base64	Кодирование / декодирование Base64-кодированных строк, например для вложений электронной почты.
JavaScript:Minifier	Уменьшение размера JavaScript-файла и запись выходного потока напрямую в другой файл.
Net::DNS	Perl-интерфейс для DNS (Domain Name System - системы доменных имен).
LWP::UserAgent	Обработка HTTP-запросов.
Net::LDAP	Perl-интерфейс к LDAP-каталогу (только в случае использования LDAP в качестве хранилища данных).
GD	Интерфейс к графической библиотеке GD (требуется только в том случае, если установлен модуль статистики OTRS).
GD::Text, GD::Graph, GD::Graph::lines, GD::Text::Align	Еще более широкий набор текстовых и графических инструментов для графической библиотеки GD (требуется только в том случае, если установлен модуль статистики OTRS).
PDF::API2, Compress::Zlib	Эти модули необходимы для генерации отчетов, результатов поиска, информации о заявке в формате PDF.

You can verify which modules you need to install with **otrs.CheckModules.pl**. This script is located in the bin directory, in the home directory of the 'otrs' user (see Script below).

Пожалуйста, обратите внимание на то, что некоторые модули не являются обязательными.

```
linux:~# cd /opt/otrs/bin/
linux:/opt/otrs/bin# ./otrs.CheckModules.pl
o CGI.....ok (v3.49)
o Crypt::PasswdMD5.....ok (v1.3)
o CSS::Minifier.....ok (v0.01)
o Date::Format.....ok (v2.24)
o Date::Pcalc.....ok (v1.2)
o DBI.....ok (v1.609)
o DBD::mysql.....ok (v4.013)
o Digest::MD5.....ok (v2.36_01)
o Encode::HanExtra.....ok (v0.23)
o GD.....ok (v2.44)
  o GD::Text.....ok (v0.86)
  o GD::Graph.....ok (v1.44)
  o GD::Graph::lines.....ok (v1.15)
  o GD::Text::Align.....ok (v1.18)
o IO::Scalar.....ok (v2.110)
o IO::Wrap.....ok (v2.110)
o JavaScript::Minifier.....ok (v1.05)
o JSON.....ok (v2.21)
  o JSON::PP.....ok (v2.27003)
  o JSON::XS.....Not installed! (Optional - Install it for faster AJAX/
JavaScript handling.)
  o LWP::UserAgent.....ok (v5.829)
  o Mail::Internet.....ok (v2.06)
  o Mail::POP3Client.....ok (v2.18 )
    o IO::Socket::SSL.....ok (v1.31)
  o MIME::Base64.....ok (v3.07_01)
  o MIME::Tools.....ok (v5.428)
```

```

o Net::DNS.....ok (v0.65)
o Net::POP3.....ok (v2.29)
o Net::IMAP::Simple.....ok (v1.1916)
  o Net::IMAP::Simple::SSL.....ok (v1.3)
o Net::SMTP.....ok (v2.31)
  o Authen::SASL.....ok (v2.15)
  o Net::SMTP::SSL.....ok (v1.01)
o Net::LDAP.....ok (v0.4001)
o PDF::API2.....ok (v0.73)
  o Compress::Zlib.....ok (v2.008)
o SOAP::Lite.....ok (v0.712)
o Text::CSV.....ok (v1.18)
  o Text::CSV_PP.....ok (v1.26)
  o Text::CSV_XS.....Not installed! (Optional - Optional, install it for
faster CSV handling.)
  o XML::Parser.....ok (v2.36)
linux:/opt/otrs/bin#

```

Script: Checking needed modules.

Для установки недостающих модулей старайтесь использовать менеджер управления пакетами, входящий в дистрибутив Linux. Таким образом, в случае выхода обновлений или новых решений по обеспечению большей безопасности, пакеты будут обновляться автоматически. Чтобы узнать как установить дополнительные пакеты обратитесь к документации, которая поставляется вместе с дистрибутивом вашей операционной системы. Если модуль (соответствующая версия) не доступен из репозитория пакетов, его всегда можно установить используя CPAN, Comprehensive Perl Archive Network (всеобъемлющую сеть архивов Perl).

Для установки любого из выше описанных модулей через CPAN, нужно выполнить команду **perl -e shell -MCPAN**. Оболочка Perl запустится в интерактивном режиме и модуль CPAN будет загружен. Если CPAN уже сконфигурирован, то с помощью команды **install** "имя модуля" (`install module_name`) можно начинать установку необходимых модулей. CPAN автоматически отслеживает зависимости между Perl-модулями и тут же оповещает, какие еще модули нужно установить.

Execute also the commands **perl -cw bin/cgi-bin/index.pl** **perl -cw bin/cgi-bin/customer.pl** and **perl -cw bin/otrs.PostMaster.pl** after changing into the directory /opt/otrs. If the output of both commands is "syntax OK", your Perl is properly set up (see Script below).

```

linux:~# cd /opt/otrs
linux:/opt/otrs# perl -cw bin/cgi-bin/index.pl
cgi-bin/installer.pl syntax OK
linux:/opt/otrs# perl -cw bin/cgi-bin/customer.pl
cgi-bin/customer.pl syntax OK
linux:/opt/otrs# perl -cw bin/otrs.PostMaster.pl
bin/otrs.PostMaster.pl syntax OK
linux:/opt/otrs#

```

Script: Syntax check.

2.3. Настройка веб-сервера Apache

В этом разделе описывается базовая конфигурация веб-сервера Apache с модулем `mod_cgi`, необходимым для работы OTRS. Веб-сервер должен поддерживать выполнение CGI-сценариев. OTRS не будет работать если нету возможности выполнять Perl-сценарии. Поэтому проверьте настройки в конфигурационных файлах веб-сервера и убедитесь в том, что строка, отвечающая за загрузку CGI-модуля не закомментирована. Если вы видите что-то вроде следующего, значит CGI-модуль уже загружен и используется.

```
LoadModule cgi_module /usr/lib/apache2/modules/mod_cgi.so
```

Для простого и удобного доступа к веб-интерфейсу OTRS через короткий адрес, нужно использовать `Alias` и `ScriptAlias`. Большинство установок Apache имеют директорию `conf.d`. Очень часто в Linux-системах эта директория находится в `/etc/apache` или `/etc/apache2`. Войдите в систему с правами администратора (под `root`-ом), затем перейдите в директорию `conf.d` и скопируйте соответствующий шаблон конфигурационного файла `/opt/otrs/scripts/apache2-httpd.include.conf` в файл `otrs.conf`, который находится в каталоге настроек Apache.

Restart your web server to load the new configuration settings. On most systems you can start/restart your web server with the command **`/etc/init.d/apache2 restart`** (see Script below).

```
linux:/etc/apache2/conf.d# /etc/init.d/apache2 restart
Forcing reload of web server: Apache2.
linux:/etc/apache2/conf.d#
```

Script: Restarting the web server.

Теперь веб-сервер полностью настроен для работы OTRS.

Для увеличения производительности можно установить `mod_perl`, отключить и не использовать модуль `mod_cgi`, а затем, сконфигурировать веб-сервер Apache на использование модуля `mod_perl` следующим образом:

Чтобы воспользоваться этой функцией убедитесь в том, что модуль `mod_perl` установлен и загружен. В связи с структурой сценария запуска, сервер не удастся запустить если модуль `mod_perl` загружен/скомпилирован неправильно или если он дальше продолжает работать. С технической точки зрения, вы все же можете оставить модуль `mod_cgi` работать, но делать это не нужно.

Search your `/etc/apache*` directory for `mod_perl.so` (see Script below) to see if the module is already loaded.

```
#!/ grep -Rn mod_perl.so /etc/apache*
```

Script: Searching for mod_perl.

Когда вы используете соответствующий сценарий запуска, приведенный выше и модуль загружен, то сценарий `/opt/otrs/scripts/apache2-perl-startup.pl` может использоваться для загрузки perl-модулей в память только один раз, что существенно экономит время загрузки и повышает производительность системы в целом.

2.4. Настройка базы данных

2.4.1. Самый простой способ - использование веб-инсталлятора (работает только с MySQL)

Если в качестве базы данных используется MySQL, то можно воспользоваться веб-инсталлятором OTRS: <http://localhost/otrs/installer.pl> .

Когда запустится веб-инсталлятор, выполните, пожалуйста, следующие шаги для установки системы:

1. Check out the information about the OTRS offices and click on next to continue (see Figure below).

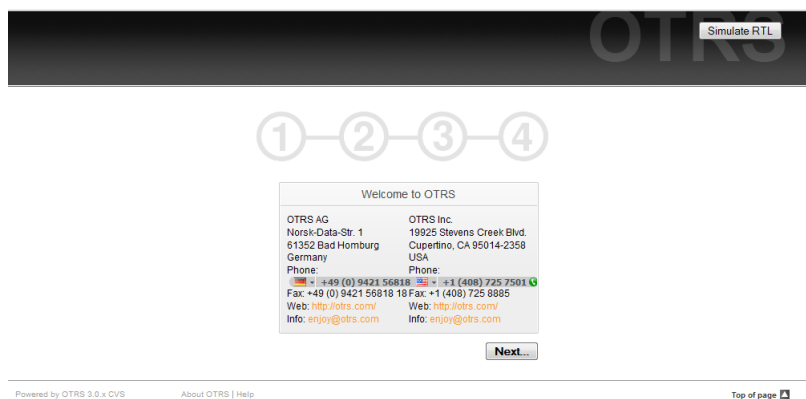


Figure: Welcome screen.

2. Read the GNU Affero General Public License (see Figure below) and accept it, by clicking the corresponding button at the bottom of the page.



Figure: GNU Affero General Public License.

3. Provide the username and password of the administrator, the DNS name of the computer which hosts OTRS and the type of database system to be used. After that, check the settings (see Figure below).

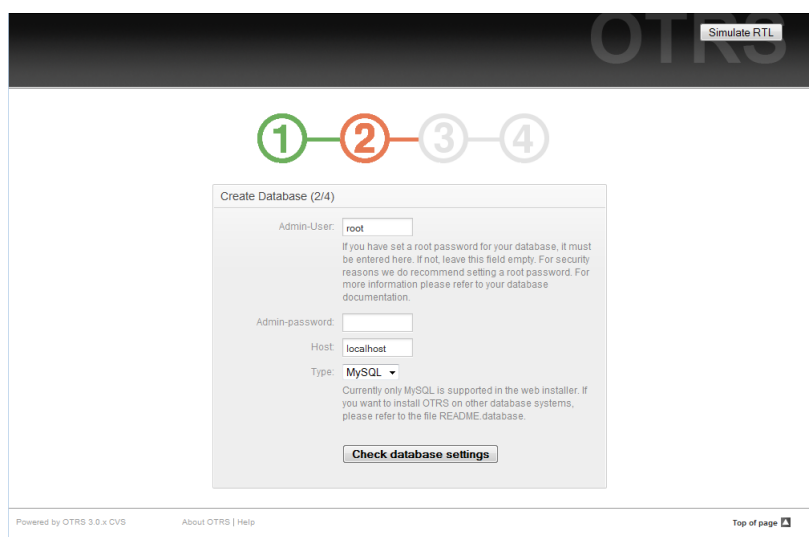


Figure: Database initial settings.

You will be notified if the check was successful. Press OK to continue (see Figure below).

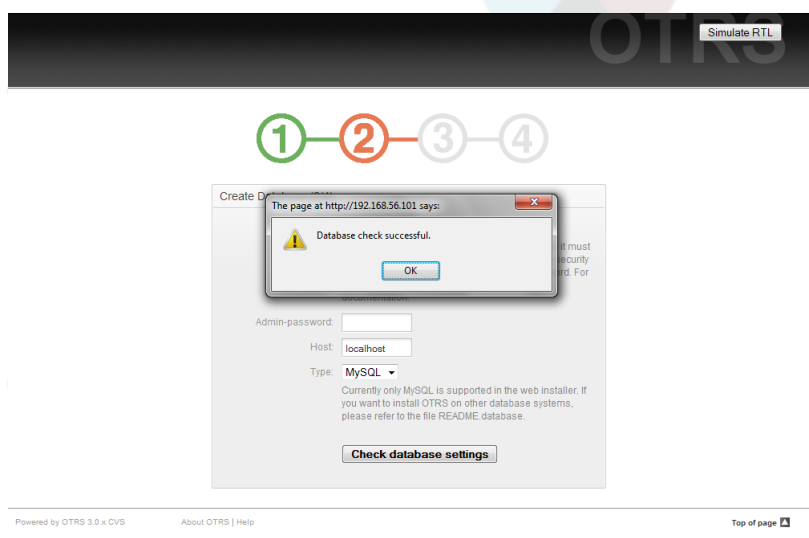


Figure: Notification for successful check.

4. Create a new database user, choose a name for the database and click on 'Next' (see Figure below).

Предупреждение

Использовать пароли по умолчанию, - не очень хорошая идея. Измените пожалуйста пароль по умолчанию для базы данных OTRS!

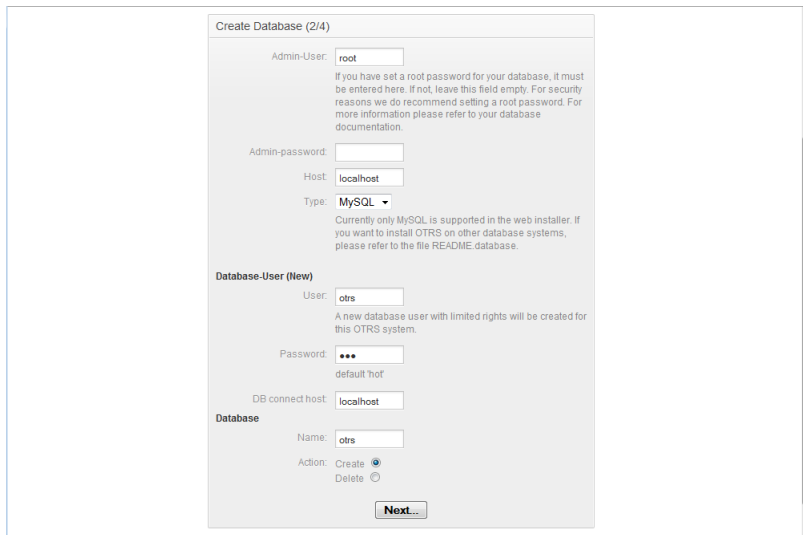


Figure: Database settings.

If the database and its user were successfully created, you will get a setup notification, as shown in Figure. Click 'Next' to go to the next screen.

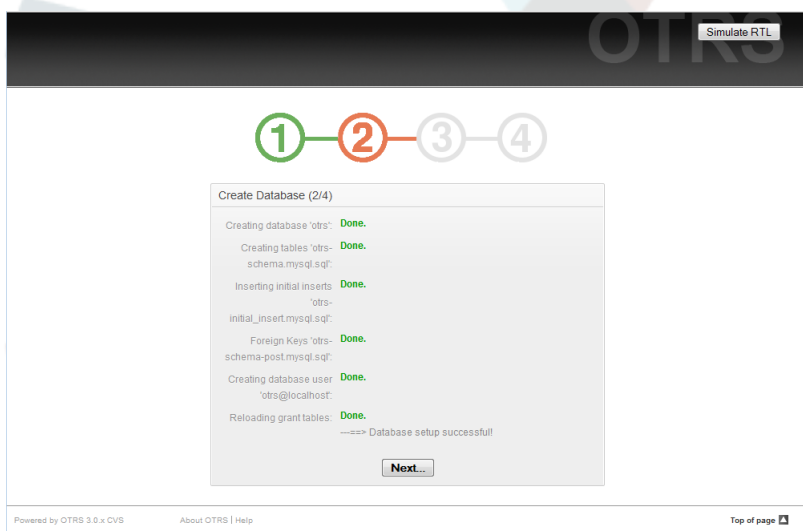


Figure: Notification indicating successful database setup.

5. Provide all the required system settings and click on 'Next' (see Figure below).

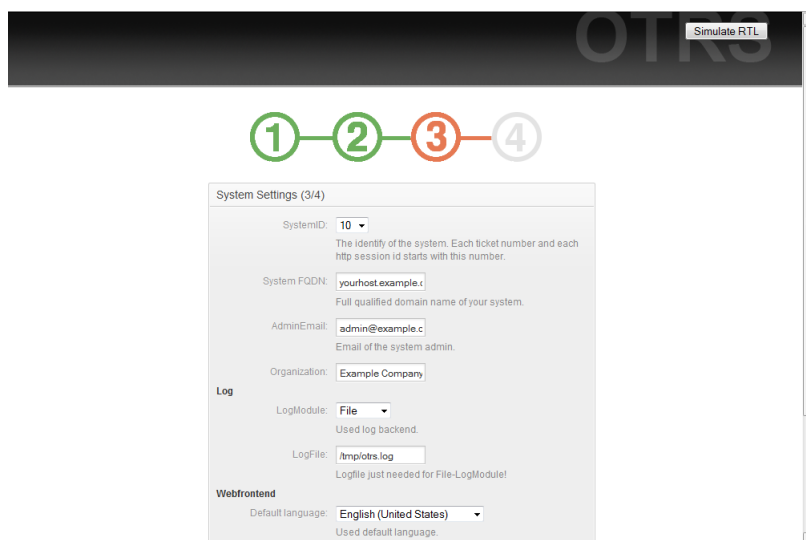


Figure: System settings.

6. If you want, you can provide the needed data to configure your inbound and outbound mail, or skip this step by pressing the right button at the bottom of the screen (see Figure below).

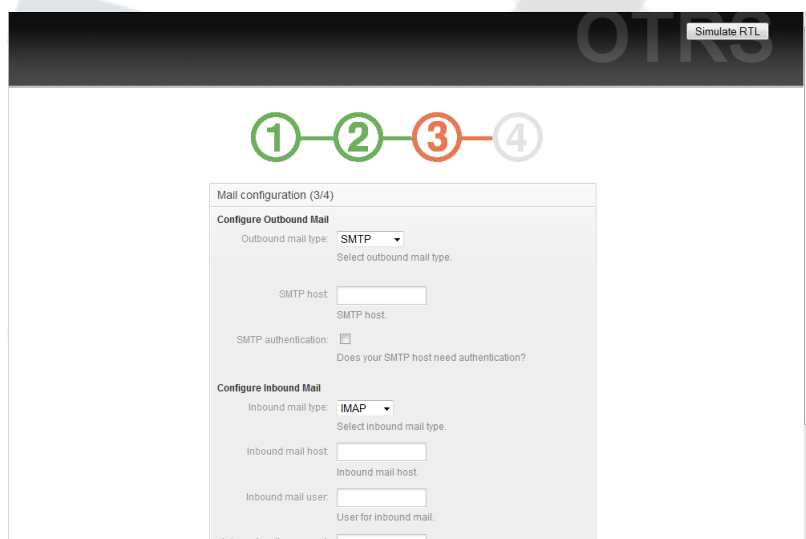


Figure: Mail configuration.

7. Restart the OTRS service now to use the new configuration settings as shown in the script below.

```
linux:~ # rcotrs restart-force
Shutting down OTRS
Disable /opt/otrs/bin/otrs.PostMaster.pl ... done.
no crontab for otrs
Shutting down cronjobs ... failed!
Shutting down OTRS (completely)
Shutting down Apache ... done.
Shutting down MySQL ... done.
done

Starting OTRS (completely)
Starting Apache ... done.
Starting MySQL ... done.
Starting OTRS
Checking Apache ... done.
```

```

Checking MySQL ... done.
Checking database connect... (It looks Ok!).
Enable /opt/otrs/bin/otrs.PostMaster.pl ... done.
Checking otrs spool dir... done.
Creating cronjobs (source /opt/otrs/var/cron/*) ... done.

-->> http://linux.example.com/otrs/index.pl <<--

done
done

linux:~ #

```

Script: Restarting the OTRS service.

Congratulations! Now the installation of OTRS is finished and you should be able to work with the system (see Figure below). To log into the web interface of OTRS, use the address <http://localhost/otrs/index.pl> from your web browser. Log in as OTRS administrator, using the username 'root@localhost' and the password 'root'. After that you can configure the system for your needs.

Предупреждение

Постарайтесь как можно быстрее изменить пароль для аккаунта 'root@localhost'.

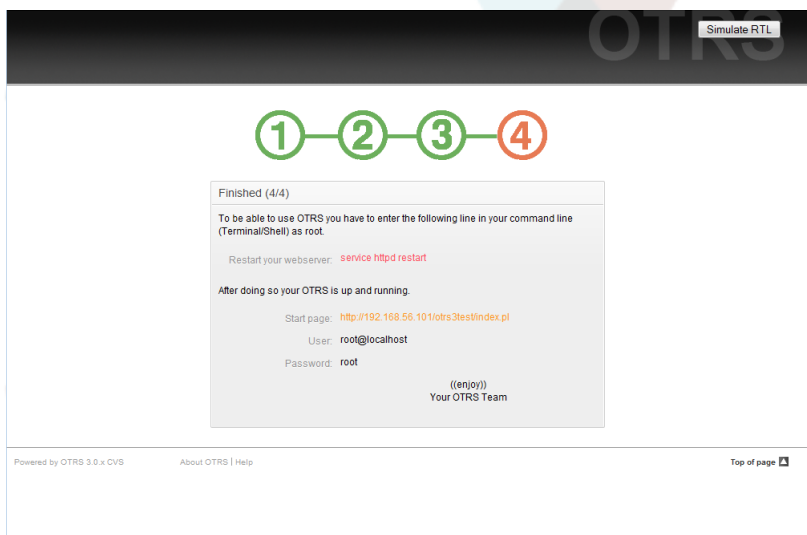


Figure: Final steps to install OTRS.

2.4.2. Установка базы данных OTRS вручную.

If you can't use the web installer to setup the OTRS database, you have to set it up manually. Scripts with the SQL statements to create and configure the database are located in `scripts/database`, in the home directory of the 'otrs' user (see Script below).

```

linux:~# cd /opt/otrs/scripts/database/
linux:/opt/otrs/scripts/database# ls
otrs-initial_insert.db2.sql      otrs-schema.mysql.sql
otrs-schema.oracle.sql          otrs-schema-post.db2.sql
otrs-initial_insert.mssql.sql    otrs-schema.postgresql.sql
otrs-initial_insert.mysql.sql    otrs-schema.postgresql.sql
otrs-initial_insert.oracle.sql   otrs-schema-post.mssql.sql
otrs-initial_insert.postgresql.sql otrs-schema-post.mysql.sql
otrs-schema.db2.sql             otrs-schema-post.oracle.sql
otrs-schema-post.postgresql.sql
otrs-schema.mssql.sql          otrs-schema.xml
linux:/opt/otrs/scripts/database#

```

Script: Files needed to create and configure the database.

При установке базы данных, для различных СУБД существует свой, определенный порядок обработки .sql-файлов.

Создание базы данных OTRS вручную, шаг за шагом

1. Создание базы данных: Используя интерфейс базы данных или свой любимый менеджер баз даных, создайте базу данных, которую планируете использовать для OTRS.
2. Создание таблиц: Используя файлы `otrs-schema.DatabaseType.sql` (например `otrs-schema.oracle.sql`, `otrs-schema.postgresql.sql`) можно создать таблицы в базе данных для OTRS.
3. Inserting the initial system data: OTRS needs some initial system data to work properly (e.g. the different ticket states, ticket and notification types). Depending on the type of your database, use one of the files `otrs-initial_insert.mysql.sql`, `otrs-initial_insert.oracle.sql`, `otrs-initial_insert.postgresql.sql` or `otrs-initial_insert.mssql.sql`.
4. Создание связей между таблицами: Последний шаг - создание связей между различными таблицами базы данных OTRS Для этого используйте файл `otrs-schema-post.DatabaseType.sql` (например `otrs-schema-oracle.post.sql`, `otrs-schema-post.postgresql.sql`).

После завершения установки базы данных необходимо проверить и установить соответствующие права доступа для базы данных OTRS. Сделать это необходимо так, чтобы только один пользователь имел соответствующие права доступа. Настройка прав доступа отличается в зависимости от выбранного вами сервера базы данных и должна производиться с помощью графического интерфейса базы данных или с помощью программы клиента.

If your database and the access rights are configured properly, you have to tell OTRS which database back-end you want to use and how the ticket system can connect to the database. Open the file `Kernel/Config.pm` located in the home directory of the 'otrs' user, and change the parameters shown in the script below according to your needs.

```
# DatabaseHost
# (The database host.)
$self->{'DatabaseHost'} = 'localhost';

# Database
# (The database name.)
$self->{'Database'} = 'otrs';

# DatabaseUser
# (The database user.)
$self->{'DatabaseUser'} = 'otrs';

# DatabasePw
# (The password of database user.)
$self->{'DatabasePw'} = 'some-pass';
```

Script: Parameters to be customized.

2.5. Настрой планировщика задач (cron jobs) для OTRS

Для правильной работы системы OTRS необходим планировщик задач (cron jobs). Планировщик задач (cron jobs) должен запускаться с теми же правами, что и модули OTRS. Именно по этому cron jobs должен быть внесен в crontab-файл пользователя 'otrs'.

All scripts with the cron jobs are located in var/cron, in the home directory of the 'otrs' user (see Script below).

```
linux:~# cd /opt/otrs/var/cron
linux:/opt/otrs/var/cron# ls
aaa_base.dist          generic_agent.dist      rebuild_ticket_index.dist
cache.dist             pending_jobs.dist      session.dist
fetchmail.dist        postmaster.dist        unlock.dist
generic_agent-database.dist  postmaster_mailbox.dist
linux:/opt/otrs/var/cron#
```

Script: Files needed to create the cron jobs.

These scripts have a suffix of '.dist'. You should copy them to files with the suffix removed. If you use bash, you might want to use the command listed in Script below.

```
linux:/opt/otrs/var/cron# for foo in *.dist; do cp $foo `basename $foo .dist`; done
linux:/opt/otrs/var/cron# ls
aaa_base          generic_agent-database.dist  rebuild_ticket_index
aaa_base.dist    generic_agent.dist          rebuild_ticket_index.dist
cache            pending_jobs                session
cache.dist      pending_jobs.dist          session.dist
fetchmail        postmaster                  unlock
fetchmail.dist  postmaster.dist            unlock.dist
generic_agent    postmaster_mailbox
generic_agent-database  postmaster_mailbox.dist
linux:/opt/otrs/var/cron#
```

Script: Copying and renaming all the files needed to create the cron jobs.

В Таблице 3-2 приводятся различные задания cron jobs.

Таблица 3.2. Описание некоторых сценариев планировщика задач cron job.

Сценарий	Назначение
aaa_base	Определяет основные настройки для crontab пользователя 'otrs'.
cache	Удаляет из диска устаревшие ("просроченные") кэш-записи. Очищает кэш-погрузчик для CSS и JavaScript файлов.
fetchmail	Этот сценарий может быть использован в том случае, если новые емейлы будут поступать в систему обработки заявок через fetchmail.
generic_agent	Выполняет задания из GenericAgent, которые не сохраняются в базе данных, а в собственных конфиг-файлах.
generic_agent-database	Выполняет задания из GenericAgent, которые хранятся в базе данных.
pending_jobs	Проверяет систему на наличие заявок, ожидающих решения, закрывает их или отправляет напоминание, если это необходимо.
postmaster	Проверяет очередь сообщений системы обработки заявок, и доставляет те сообщения, которые находятся в этой очереди.

Сценарий	Назначение
postmaster_mailbox	Получает почту с POP3-счетов, которые были указаны в админке, в разделе "Учетные записи PostMaster".
rebuild_ticket_index	Восстанавливает индекс заявки, что значительно повышает скорость просмотра заявок в разделе QueueView.
session	Удаляет старые и больше не используемые ID-сессий (session IDs).
unlock	Открывает заявки, которые были ранее заблокированы в системе.

Чтобы настроить все задания cron jobs используйте скрипт bin/Cron.sh, который находится в домашнем каталоге пользователя 'otrs'. При запуске, скрипту нужно передать один из трех параметров, указывающий на выполняемое действие: создание, удаление, переустановка заданий. Допустимы следующие параметры:

Cron.sh {start} {stop} {restart} [OTRS user]

Because the cron jobs need to be installed in the crontab file of the 'otrs' user, you need to be logged in as 'otrs'. If you are logged in as root, you can switch to 'otrs' with the command **su otrs**. Execute the commands specified in Script below to install the cron jobs.

Предупреждение

Обратите внимание, что при использовании файла Cron.sh другие задания, установленные в crontab-файле для пользователя 'otrs' будут перезаписаны или удалены. Внесите все необходимые изменения в файл Cron.sh, чтобы сохранить другие crontab-задания.

```
linux:/opt/otrs/var/cron# cd /opt/otrs/bin/
linux:/opt/otrs/bin# su otrs
linux:~/bin$ ./Cron.sh start
/opt/otrs/bin
Cron.sh - start/stop OTRS cronjobs
Copyright (C) 2001-2009 OTRS AG, http://otrs.org/
(using /opt/otrs) done
linux:~/bin$ exit
exit
linux:/opt/otrs/bin#
```

Script: Installing the cron jobs.

The command **crontab -l -u otrs**, which can be executed as root, shows you the crontab file of the 'otrs' user, and you can check if all entries are placed correctly (see Script below).

```
linux:/opt/otrs/bin# crontab -l -u otrs
# --
# cron/aaa_base - base crontab package
# Copyright (C) 2001-2014 OTRS AG, http://otrs.com/
# --
# This software comes with ABSOLUTELY NO WARRANTY.
# --
# Who gets the cron emails?
MAILTO="root@localhost"
# --
# cron/cache - delete expired cache
# Copyright (C) 2001-2014 OTRS AG, http://otrs.com/
```

```

# This software comes with ABSOLUTELY NO WARRANTY.
# --
# delete expired cache weekly (Sunday mornings)
20 0 * * 0 $HOME/bin/otrs.CacheDelete.pl --expired >> /dev/null
30 0 * * 0 $HOME/bin/otrs.LoaderCache.pl -o delete >> /dev/null

# --
# cron/fetchmail - fetchmail cron of the OTRS
# Copyright (C) 2001-2014 OTRS AG, http://otrs.com/
# --
# This software comes with ABSOLUTELY NO WARRANTY.
# --
# fetch every 5 minutes emails via fetchmail
*/5 * * * * /usr/bin/fetchmail -a >> /dev/null

# --
# cron/generic_agent - otrs.GenericAgent.pl cron of the OTRS
# Copyright (C) 2001-2014 OTRS AG, http://otrs.com/
# --
# --
# This software comes with ABSOLUTELY NO WARRANTY.
# --
# start generic agent every 20 minutes
*/20 * * * * $HOME/bin/GenericAgent.pl >> /dev/null
# example to execute GenericAgent.pl on 23:00 with
# Kernel::Config::GenericAgentMove job file
#0 23 * * * $HOME/bin/otrs.GenericAgent.pl -c "Kernel::Config::GenericAgentMove" >> /dev/
null
# --
# cron/generic_agent - GenericAgent.pl cron of the OTRS
# Copyright (C) 2001-2014 OTRS AG, http://otrs.com/
# --
# This software comes with ABSOLUTELY NO WARRANTY.
# --
# start generic agent every 10 minutes
*/10 * * * * $HOME/bin/otrs.GenericAgent.pl -c db >> /dev/null
# --
# cron/pending_jobs - pending_jobs cron of the OTRS
# Copyright (C) 2001-2014 OTRS AG, http://otrs.com/
# --
# This software comes with ABSOLUTELY NO WARRANTY.
# --
# check every 120 min the pending jobs
45 */2 * * * $HOME/bin/otrs.PendingJobs.pl >> /dev/null
# --
# cron/postmaster - postmaster cron of the OTRS
# Copyright (C) 2001-2014 OTRS AG, http://otrs.com/
# --
# This software comes with ABSOLUTELY NO WARRANTY.
# --
# check daily the spool directory of OTRS
#10 0 * * * test -e /etc/init.d/otrs & /etc/init.d/otrs cleanup >> /dev/null; test -e /
etc/rc.d/init.d/otrs && /etc/rc.d/init.d/otrs cleanup >> /dev/null
10 0 * * * $HOME/bin/otrs.CleanUp.pl >> /dev/null
# --
# cron/postmaster_mailbox - postmaster_mailbox cron of the OTRS
# Copyright (C) 2001-2014 OTRS AG, http://otrs.com/
# --
# This software comes with ABSOLUTELY NO WARRANTY.
# --
# fetch emails every 10 minutes
*/10 * * * * $HOME/bin/otrs.PostMasterMailbox.pl >> /dev/null
# --
# cron/rebuild_ticket_index - rebuild ticket index for OTRS
# Copyright (C) 2001-2014 OTRS AG, http://otrs.com/
# --
# This software comes with ABSOLUTELY NO WARRANTY.
# --
# just every day
01 01 * * * $HOME/bin/otrs.RebuildTicketIndex.pl >> /dev/null

# --

```



```
# cron/session - delete old session ids of the OTRS
# Copyright (C) 2001-2014 OTRS AG, http://otrs.com/
# --
# This software comes with ABSOLUTELY NO WARRANTY.
# --
# delete every 120 minutes old/idle session ids
55 */2 * * * $HOME/bin/otrs.DeleteSessionIDs.pl --expired >> /dev/null

# --
# cron/unlock - unlock old locked ticket of the OTRS
# Copyright (C) 2001-2014 OTRS AG, http://otrs.com/
# --
# This software comes with ABSOLUTELY NO WARRANTY.
# --
# unlock every hour old locked tickets
35 * * * * $HOME/bin/otrs.UnlockTickets.pl --timeout >> /dev/null

linux:/opt/otrs/bin#
```

Script: Crontab file.

3. Обновление OTRS-фреймворка

These instructions are for people upgrading OTRS from version 3.0 to 3.1, and apply both for RPM and source code (tarball) upgrades.

If you are running a lower version of OTRS you have to follow the upgrade path to 3.0 first (1.1->1.2->1.3->2.0->2.1->2.2->2.3->2.4->3.0->3.1 ...)!

Please note that if you upgrade from OTRS 2.2 or earlier, you have to take an extra step; please read http://bugs.otrs.org/show_bug.cgi?id=6798.

Within a single minor version you can skip patch level releases if you want to upgrade. For instance you can upgrade directly from OTRS 3.1.1 to version 3.1.4. If you need to do such a "patch level upgrade", you should skip steps 9, 11 and 13-19.

Please note that for upgrades from 3.1.beta1 or 3.1.beta2, an additional step 21 is needed!

If you are using Microsoft SQL Server as the DBMS for OTRS, please refer to the manual, chapter "Upgrading Microsoft SQL Server Data Types" for instructions how to upgrade the data types used by OTRS (<http://doc.otrs.org/3.1/en/html/upgrading-mssql-datatypes.html>).

1. Остановите все соответствующие службы.

в т.ч. (зависящие от используемых услуг):

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

2. Сделайте резервные копии всех данных из \$OTRS_HOME (по умолчанию OTRS_HOME=/opt/otrs):

- Kernel/Config.pm
- Kernel/Config/GenericAgent.pm
- Kernel/Config/Files/ZZZAuto.pm
- var/*

- as well as the database

3. Make sure that you have backed up everything ;-)

4. Setup new system (optional)

Если есть возможность, сначала попробуйте произвести установку на отдельном тестовом компьютере.

5. Установка нового релиза (из tar-архива или с помощью RPM-пакетов).

- С помощью тарбола (tarball):

```
shell> cd /opt
shell> tar -xzf otrs-x.x.x.tar.gz
shell> ln -s otrs-x.x.x otrs
```

Восстановите старые конфигурационные файлы.

- Kernel/Config.pm
- Kernel/Config/GenericAgent.pm
- Kernel/Config/Files/ZZZAuto.pm
- С помощью RPM-пакетов:

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

В этом случае обновление из RPM автоматически восстанавливает старые конфигурационные файлы.

6. Собственные темы

Note: The OTRS themes between 3.0 and 3.1 are NOT compatible, so don't use your old themes!

Themes are located under `$OTRS_HOME/Kernel/Output/HTML/*/*.dtl` (default: `OTRS_HOME=/opt/otrs`).

7. Установка прав доступа к файлам.

В случае использования тарбола (tarball), выполните

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

с правами, необходимыми для установки системы.

8. Примените изменения к базе данных (часть 1/2):

```
shell> cd /opt/otrs/
# MySQL:
shell> cat scripts/DBUpdate-to-3.1.mysql.sql | mysql -p -f -u root otrs
# PostgreSQL 8.2+:
shell> cat scripts/DBUpdate-to-3.1.postgresql.sql | psql otrs
# PostgreSQL, older versions:
```

```
shell> cat scripts/DBUpdate-to-3.1.postgresql_before_8_2.sql | psql otrs
```

NOTE: If you use PostgreSQL 8.1 or earlier, you need to activate the new legacy driver for these older versions. Do this by adding a new line to your Kernel/Config.pm like this:

```
$Self->{DatabasePostgresqlBefore82} = 1;
```

Запустите сценарий перемещения (как пользователь 'otrs', а НЕ root):

Сценарий перемещения (миграции) нужно запускать для перемещения некоторых данных из старой базы данных в новую. Пожалуйста, запустите:

```
shell> scripts/DBUpdate-to-3.1.pl
```

Примените изменения к базе данных (часть 2/2):

```
# MySQL:
shell> cat scripts/DBUpdate-to-3.1-post.mysql.sql | mysql -p -f -u root otrs
# PostgreSQL 8.2+:
shell> cat scripts/DBUpdate-to-3.1-post.postgresql.sql | psql otrs
# PostgreSQL, older versions:
shell> cat scripts/DBUpdate-to-3.1-post.postgresql_before_8_2.sql | psql otrs
```

9. Обновите конфигурацию системы и удалите все данные из кэша. Пожалуйста, введите:

```
shell> bin/otrs.RebuildConfig.pl
shell> bin/otrs.DeleteCache.pl
```

10 Update your web server configuration

Note: this applies only if you use the Apache web server together with mod_perl2, and do not use the configuration file directly from the OTRS installation directory (e. g. with a symlink from the Apache configuration directory).

Please add a new setting to the Apache configuration file for OTRS:

```
# set mod_perl2 option for generic interface
<Location /otrs/nph-genericinterface.pl>
  PerlOptions -ParseHeaders
</Location>
```

Please see the file /opt/otrs/scripts/apache2-httpd.include.conf for an example of where this new option needs to be added (inside the <IfModule mod_perl.c> block).

In this file, you will also note a new section on caching:

```
<IfModule mod_headers.c>
  <Directory "/opt/otrs/var/httpd/htdocs/skins/*/*/css-cache">
    <FilesMatch "\.(css|CSS)$">
      Header set Cache-Control "max-age=2592000 must-revalidate"
    </FilesMatch>
  </Directory>

  <Directory "/opt/otrs/var/httpd/htdocs/js/js-cache">
```

```
<FilesMatch "\.(js|JS)$">
  Header set Cache-Control "max-age=2592000 must-revalidate"
</FilesMatch>
</Directory>
</IfModule>
```

Please activate this in your local installation too, and make sure that mod_headers is installed and active.

11Перезапустите сервисы.

e. g. (depends on used services):

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

Now you can log into your system.

12Check installed packages

In the package manager, check if all packages are still marked as correctly installed or if any require reinstallation or even a package upgrade.

13Check for encoding issues

OTRS 3.1 only allows UTF-8 as internal charset. Non-UTF-8 installations of OTRS must switch to UTF-8.

14Escalation events

If you want to use the new escalation events in your system, you need to activate the corresponding GenericAgent job in Kernel/Config/GenericAgent.pm. Please look into Kernel/Config/GenericAgent.pm.dist for an example of how to do this.

15TicketHistory

The TicketHistory type TicketFreeTextUpdate was renamed to TicketDynamicFieldUpdate. If you have any custom reporting using this, please adjust it.

16Ticket event handlers

The Event name TicketFreeTextUpdate_\$\$Counter was renamed to TicketDynamicFieldUpdate_\$\$FieldName. If you have any custom event handlers for these events, please adapt them.

17DynamicField user preferences module

If you had one or more active custom settings for "PreferencesGroups###Freetext", you need to adapt them to work with the new DynamicFields engine. The PrefKey setting must be changed to "UserDynamicField_DynamicField", where the part after the _ is the name of the dynamic field. Existing values would need to be renamed in the database as well.

18Custom free field default value event handler

If you used the event handler Ticket::EventModulePost###TicketFreeFieldDefault (not active by default), you'll need to migrate its configuration to the new setting Ticket::EventModulePost###TicketDynamicFieldDefault.

The configuration of this is slightly different; where you had to specify a Counter indicating the TicketFreeText number previously, now you need to specify the name of the DynamicField (for migrated fields, this will be DynamicField_TicketFreeKey\$Counter and DynamicField_TicketFreeText\$Counter. You need two separate entries now if you want to set both the key and the text field.

19 ACL changes

If you have any ACLs defined which involve freetext or freetime fields, you need to adjust these ACL definitions.

Please have a look at <http://doc.otrs.org/3.1/en/html/acl.html>. There you can find a list of all possible ACL settings. In general, you need to add the prefix "DynamicField_" to existing free field definitions, and you can add a new "DynamicField" section to the "Properties" list for situations when a ticket does not exist yet.

Please also note that since OTRS 3.1.5, it is possible to match either the values of an existing ticket as it is on the database (new setting "PropertiesDatabase"), in addition to the existing match against the current values of the form. Details can be found at <http://doc.otrs.org/3.1/en/html/acl.html>.

20 SMIME file names

This step is ONLY needed if you use SMIME to encrypt/sign your emails. You need to take this step if you upgrade from 3.0.x OR if you upgrade from 3.1.x before 3.1.6 while using OpenSSL 1.x to encrypt/sign your emails.

Between OTRS 3.0.x and 3.1.x, the file naming of private secret file names changed and need to be updated. In addition, OpenSSL 0.9.x and 1.x use different algorithms to calculate the certificate hash. To update your private secrets and certificate file names, please run this command:

```
shell> bin/otrs.RefreshSMIMEKeys.pl
```

This step must also be taken if you change a running installation from OpenSSL 0.9 to 1.x.

21 Database Upgrade During Beta Phase

This step is ONLY needed if you upgrade from 3.1.beta1 or 3.1.beta2! Please apply the required database changes as follows:

```
MySQL:  
shell> cat scripts/DBUpdate-3.1.beta.mysql.sql | mysql -p -f -u root otrs  
  
PostgreSQL 8.2+:  
shell> cat scripts/DBUpdate-3.1.beta.postgresql.sql | psql otrs  
  
PostgreSQL, older versions:  
shell> cat scripts/DBUpdate-3.1.beta.postgresql_before_8_2.sql | psql otrs
```

22 Молодцы!

4. Обновление с помощью Windows Installer

В настоящее время нету механизма автоматического обновления копии OTRS, которая была установлена с использованием инсталлятора Windows (Windows Installer). В основном, процесс обновления состоит из таких шагов: архивирование базы дан-

ных и всей файловой системы, деинсталляция OTRS, установка новой версии, восстановление базы данных и запуск процедуры обновления (если она необходима).

Upgrading is described in [FAQ# 4200351](#), and there is also an informative [YouTube video](#) available.

5. Upgrading Microsoft SQL Server Data Types

Starting OTRS version 3.1, OTRS uses the *NVARCHAR* data type rather than *VARCHAR* or *TEXT*, to store textual data. This is because the *NVARCHAR* type has full support for Unicode, whereas the old data types store data in UCS-2 format, which is a sub-set of Unicode. Also, the *TEXT* data type is deprecated since *SQL Server 2005*. Due to this, starting with OTRS version 3.1, the minimal SQL Server version required for operation with OTRS is now *Microsoft SQL Server 2005*.

Because dropping and re-creating these indexes is a time-consuming operation, especially on large databases, please plan enough time for performing the upgrade. We would recommend that you perform the upgrade on a copy of the database prior to doing the actual conversion to test the upgrade procedure and to time how much time will be needed on your specific environment.

Please make sure that, before you start, there is enough space available on the database server. Make sure the free space on your database server is at least 2.5x the current size of the database.

Важно

This upgrade procedure will upgrade all fields of the mentioned data types to the new types. This procedure first removes any indexes and constraints in which these fields are referenced, upgrades the fields, and then adds the indexes and constraints back. It will do so on all tables found in the SQL Server database that OTRS uses. If you would have stored non-OTRS tables in the OTRS database, and these tables contain columns of the data types *VARCHAR* or *TEXT*, these will also be updated.

1. Open a Command Line on the OTRS server.
2. Change directory to the OTRS root directory. If you're using the default OTRS installer this would be C:\Program Files\OTRS\OTRS.
3. Run the following command:

```
shell> perl scripts/DUpdate-to-3.1.mssql-datatypes.pl
```

4. This will generate three scripts in the specified directory `scripts\database\update`. Run these scripts on the SQL Server database, via SQL Server Management Studio or `isql`.

Глава 4. Первые шаги

В этой главе приводится общий обзор OTRS и структура веб-интерфейса. Дается объяснение таким терминам как: агенты, клиенты и администраторы. В этом разделе вы войдете в систему с правами OTRS-администратора и более внимательно ознакомитесь с настройками пользователей, доступными для каждого из типов аккаунта.

1. Веб-интерфейс для агентов

Агентский веб-интерфейс позволяет агентам отвечать на вопросы клиентов, создавать новые заявки для клиентов и агентов, создавать заявки на основе телефонных звонков клиентов, писать и редактировать записи FAQ-модуля, редактировать данные клиентов и так далее.

Supposing your OTRS host is reachable via the URL <http://www.example.com> , then the OTRS login screen can be reached by using the address <http://www.example.com/otrs/index.pl> in a web browser (see Figure below).

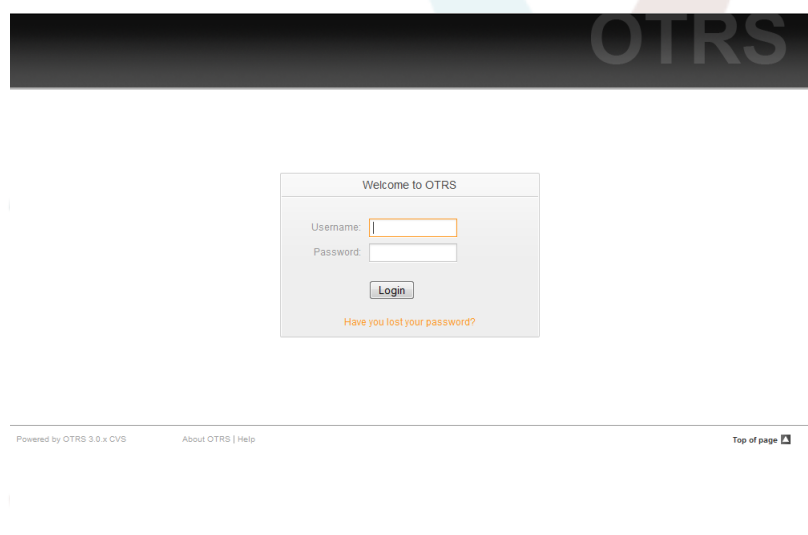


Figure: Login screen of the agent interface.

2. Пользовательский веб-интерфейс

В OTRS клиенты имеют отдельный веб-интерфейс, с помощью этого веб-интерфейса они могут создавать новые учетные записи, просматривать собственные заявки, создавать и редактировать заявки, изменять настройки своей учетной записи и т.д.

Continuing with the above example, the customer login screen can be reached by using the URL <http://www.example.com/otrs/customer.pl> with a web browser (see Figure below).

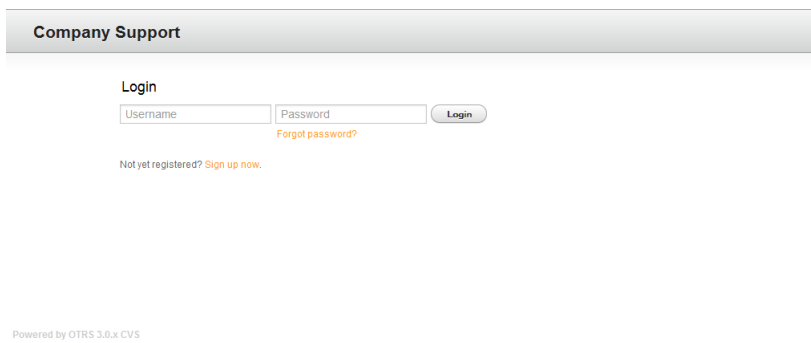


Figure: Login screen of the customer interface.

3. Общедоступный веб-интерфейс

В дополнение к веб-интерфейсу для агентов и пользователей, OTRS имеет открытый веб-интерфейс (см. ниже Рисунок 4.3), доступ к которому возможен посредством FAQ-модуля (который устанавливается отдельно). Он предоставляет открытый доступ к FAQ-системе и позволяет пользователям без авторизации искать информацию.

В нашем примере общий веб-интерфейс будет доступен перейдя по ссылке: <http://www.example.com/otrs/faq.pl> , <http://www.example.com/otrs/public.pl>

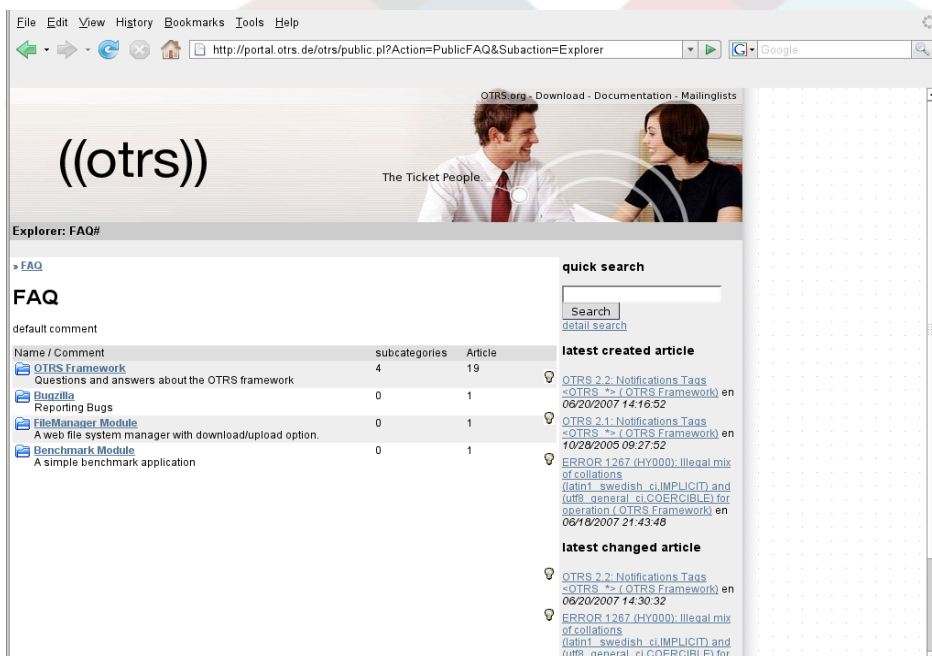


Figure: Public web interface.

4. Первое применение

Перейдите к странице входа, как это описано в разделе Агентский веб-интерфейс . Введите имя пользователя и пароль. Ввиду того что после новой установки в си-

стеме нету ни одного зарегистрированного пользователя, войдите с правами OTRS-администратора, используя для этого имя пользователя 'root@localhost' и пароль 'root'.

Предупреждение

Данные доступа этого аккаунта действительны для каждой копии новоустановленной системы OTRS. Именно по этой причине пароль OTRS-администратора необходимо изменить как можно быстрее. Это можно сделать после входа в систему с правами администратора перейдя по ссылке preferences.

Если не хотите входить в систему с правами OTRS-администратора, просто введите имя и пароль своего обычного агентского аккаунта.

In case you have forgotten your password, you can request the system for a new password. Simply press the link below the Login button, enter the mail address that is registered for your OTRS account into the input field, and press the Submit button (see Figure).

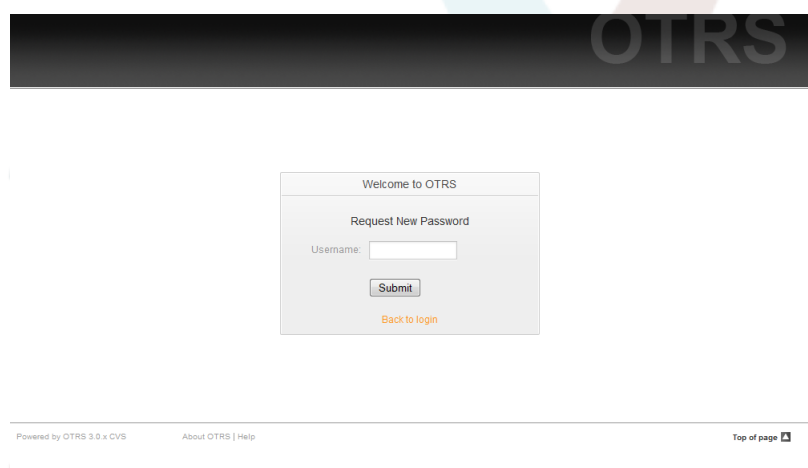


Figure: Request new password.

5. Веб-интерфейс - обзор

On successfully logging into the system, you are presented with the Dashboard page (see Figure below). The Dashboard is completely customizable. It shows your locked tickets, allows direct access through menus to the queue, status and escalation views, and also holds options for creation of new phone and e-mail tickets. It also presents a quick summary of the tickets which are pending, escalated, new and open.

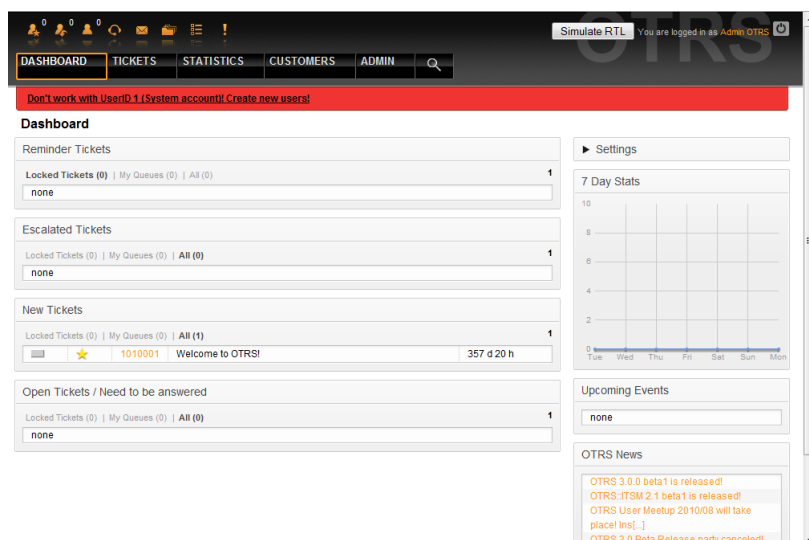


Figure: Dashboard of the agent interface.

Для большей ясности, общий веб-интерфейс разделен на несколько частей (областей). В верхней части окна отображается строка с общей информацией: имя пользователя, кнопка выхода, иконки, отображающие количество заблокированных заявок (позволяют прямой доступ к ним), ссылки для создания новой заявки на основе телефонного звонка или сообщения электронной почты. Также есть иконки для перехода к очереди, просмотра информации о статусах и эскалациях.

Под иконками находится навигационная строка. Она показывает меню, позволяющие переходить к разным модулям системы и позволяет выполнять некоторые глобальные действия. При нажатии на кнопку Панель Управления вы перейдете на страницу Панели Инструментов, которая по умолчанию отображается после входа в систему. Если нажать ссылку меню Заявки, откроется подменю, предлагающее различные действия, допустимые над заявками: создание новой заявки (на основе телефонного звонка или сообщения электронной почты), поиск конкретных заявок. Кнопка Статистика также является частью меню и позволяет просматривать существующую статистическую информацию, создать новую а также производить импорт существующей статистики. Кнопка Клиенты позволяет перейти на страницу Управления Клиентами. Для администрирования системы нажмите кнопку Администрирование. На этой странице можно создавать аккаунты для других агентов, очереди и т.д. Для поиска заявок используйте кнопку Поиск.

Если установлены дополнительные приложения, такие как, например, Файловый Менеджер или Менеджер Емейл-Рассылки, кнопки для запуска этих приложений будут также отображаться.

Красная полоса, которая находится под панелью навигации предназначена для отображения различных системных сообщений. Если войти в систему с правами OTRS-администратора, будет выведено сообщение о том, что в системе не следует работать с правами администратора.

Под названием раздела, в котором вы находитесь в данный момент, есть несколько подразделов (каждый из них находится в отдельном блоке), которые можно перетаскивать и перемещать в пределах блока или перенести их в другое место.

В левой колонке находится информация о заявках: напоминания, эскалации, новые или открытые. В каждой из категорий можно просматривать все заявки, к которым у вас есть доступ, сколько заявок заблокировано и сколько находится в "Мои Очереди". "Мои Очереди" - очередь, которая определяется в настройках пользовательского аккаунта предпочтения. Она используется в том случае, когда нужно отслеживать заявки с определенными параметрами (характеристиками).

In the right column is the Settings button. Click on it to expand the section and see the various settings, as shown in Figure. You can then check or uncheck the individual settings options, and save your changes. This section is fixed, so you can not drag and drop it.

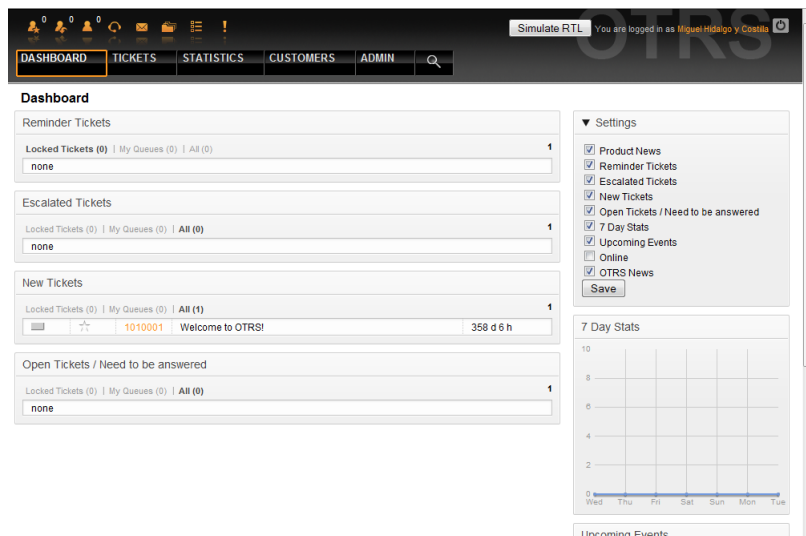


Figure: Dashboard Settings.

Ниже, под настройками, можно увидеть раздел с статистическим графиком по заявкам за последние 7 дней. Ниже располагается секция, в которой отображаются Предстоящие События и новости OTRS.

Finally at the bottom of the page, the site footer is displayed (see Figure below). It contains links to directly access the OTRS official website, or go to the Top of the page.

Powered by OTRS 3.0.x CVS About OTRS | Help Top of page

Figure: Footer.

6. Что такое очередь?

На физическом уровне для почтового ящика создается специальный файл (иногда его еще называют почтовый спул), в который помещаются и хранятся все сообщения. Новые сообщения электронной почты просто добавляются в конец этого файла. Почтовая программа, которая используется для чтения и написания писем (такие программы еще называются пользовательскими почтовыми агентами - MUA (Mail User Agent) читает этот файл и отображает его содержимое.

Очереди OTRS можно сравнить с Inbox-файлом, в котором хранятся входящие сообщения электронной почты, так как в них тоже может храниться множество сообщений. Очереди имеют много функций, схожих с теми, которые присущи Inbox-файлу. Как агенты так и пользователи, все должны помнить о том, в какой очереди содержится заявка. Агенты могут открывать, редактировать заявки, размещенные в очереди, а также, перемещать их из одной очереди в другую. Но зачем перемещать заявки?

Чтобы объяснить это с более практической точки зрения давайте вспомним пример с компанией Максима, которая была описана в разделе пример системы обработки заявок. Максим установил OTRS чтобы избавиться от существующего хаоса в системе поддержки и предоставить возможность своей команде лучше управлять службой поддержки клиентов, покупающих видеомагнитофоны.

На данный момент достаточно одной очереди, чтобы хранить в ней все запросы клиентов. Но через некоторое время Максим решает начать продажи DVD-магнито-

фонов. Теперь у клиентов есть вопросы касающиеся не только видеомagneтофонов, но и вопросы относительно нового продукта. С течением времени в очередь, ранее созданную Максимом, начинает приходить все больше и больше писем и управляться с ними становится все труднее и труднее.

Однажды Максим принимает решение немного оптимизировать структуру системы поддержки и добавляет две новых очереди. Таким образом в системе используется уже три очереди. Новые заявки, только поступившие в систему хранятся в очереди "raw". Две новые очереди: одна для хранения заявок касающихся видеомagneтофонов и другая - для dvd-magneтофонов.

Максим просит Маргариту просматривать очередь "raw" и сортировать емейлы либо в очередь "видеомagneтофоны" или в "dvd-magneтофоны", в зависимости от того, что запросил клиент. Андрей имеет доступ только к очереди "видеомagneтофоны", а Сергей может отвечать на заявки из очереди "dvd-magneтофоны". Максим имеет возможность редактировать заявки, находящиеся во всех очередях.

OTRS поддерживает управление доступом для пользователей, групп и ролей, таким образом можно очень легко настроить очереди, которые доступны только для некоторых учетных записей. Чтобы все запросы приходили в различные очереди, Максим может использовать правила фильтрации. Иначе, если используются два различных email-адреса, Маргарита будет сортировать и направлять в другие очереди только те заявки, которые не были отсортированы автоматически.

Сортировка входящих сообщений в различные очереди позволяет сохранять систему поддержки в структурированном и аккуратном виде. Поскольку ваши агенты находятся в разных группах и имеют различные права для доступа к очередям, система может быть оптимизирована еще больше. Очереди могут быть использованы для определения структуры всей компании а также для определения рабочего потока процессов. Максим может к примеру создать еще одну очередь и назвать ее "продажи", которая будет содержать подочереди "запросы", "предложения", "заказы", "оплаты" и т.д. Такой подход поможет Максиму структурировать и оптимизировать процесс работы с транзакциями.

Чем лучше структурирована система поддержки, тем меньше времени требуется для различных задач. Это приводит к уменьшению рабочего времени, соответственно и к уменьшению денежных затрат. Очереди могут помочь оптимизировать бизнес-процессы вашей компании.

7. Пользовательские настройки

OTRS users such as customers, agents and the OTRS administrator can configure their account preferences as per their needs. Agent can access the configuration screen by clicking on their login name at the top right corner of the web interface (see Figure below), and customers must click on the "Preferences" link (see Figure below).

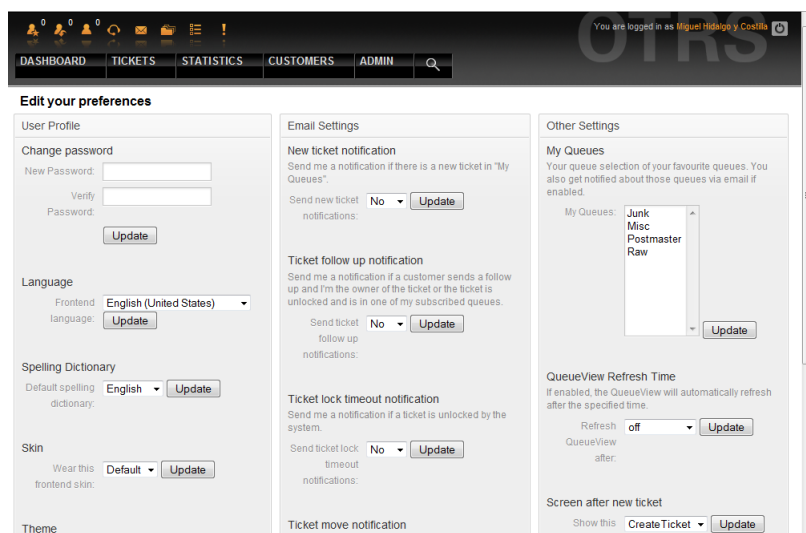


Figure: Agent's personal preferences.

Агент может настраивать три различных категории настроек: профиль пользователя, настройки электронной почты и другие параметры. Возможности по умолчанию:

Профиль пользователя

- Изменение текущего пароля.
- Выбор языка интерфейса.
- Переключение между темами интерфейса.
- Изменение темы веб-интерфейса.
- Активация и настройка функции "время вне офиса".

Настройки электронной почты

- Выберите события, при наступлении которых, система должна отправлять уведомления агентам на их емайлы.

Другие настройки

- В меню "Мои Очереди" выберите те очереди, которые хотите отслеживать более внимательно.
- Установите время обновления страницы для шаблона отображения очередей.
- Выберите страницу, которая должна отображаться после создания заявки.

Company Personal Support Simulate RTL

New Ticket | My Tickets | Company Tickets | Search Preferences | Logout

Interface language

Your language: English (United States)

Number of displayed tickets

Max. displayed tickets: 25

Ticket overview

Refresh interval: off

Change password

New Password:

Verify Password:

SIMME Certificate

SIMME Certificate Upload:

Powered by OTRS 3.0.x CVS [About OTRS Help](#)

Figure: Customer's personal preferences.

Клиент может выбрать язык веб-интерфейса системы, изменять время обновления страницы в шаблоне просмотра очередей, выбирать максимальное количество отображаемых заявок. Кроме этого также можно установить новый пароль.

Глава 5. Панель управления OTRS

1. Основы

Администраторы используют страницу администрирования веб-интерфейса OTRS для настройки системы - добавления агентов, клиентов и очередей, заявок и настройки почты, установки дополнительных пакетов, таких как FAQ (ЧАВО) и ITSM и многое другое.

Agents who are members of the *admin* group can access the Admin area by clicking the *Admin* link in the navigation bar (see Figure below). The rest of the agents won't see this link.

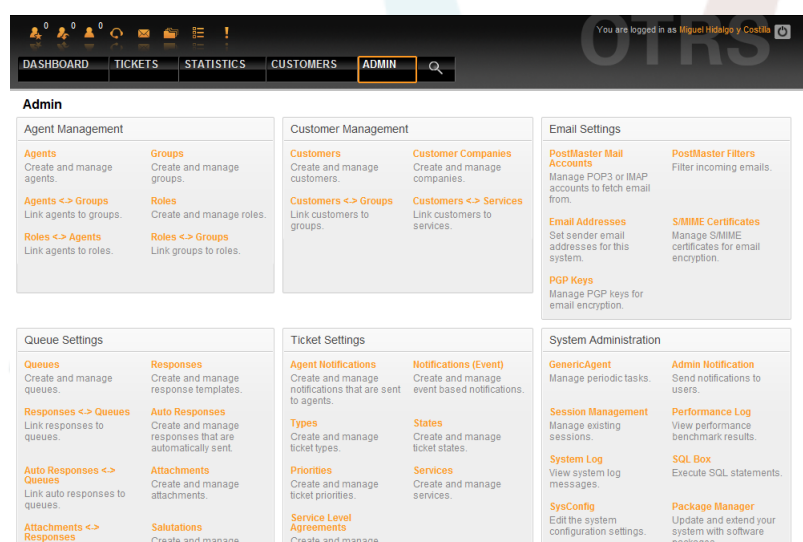


Figure: OTRS Admin screen.

2. Агенты, Группы и Роли

2.1. Агенты

By clicking the link *Agents*, you get access to the agent management screen of OTRS (see Figure below). Administrators can add, change or deactivate agent accounts. Administrators can also manage agent preferences, for instance the language and notification settings for their interface.

Примечание

Учетная запись (аккаунт) агента может быть деактивирована, но не удалена. Деактивация осуществляется путем установки параметра Действительный в значение *не действительный* или *не действительный-временно*.

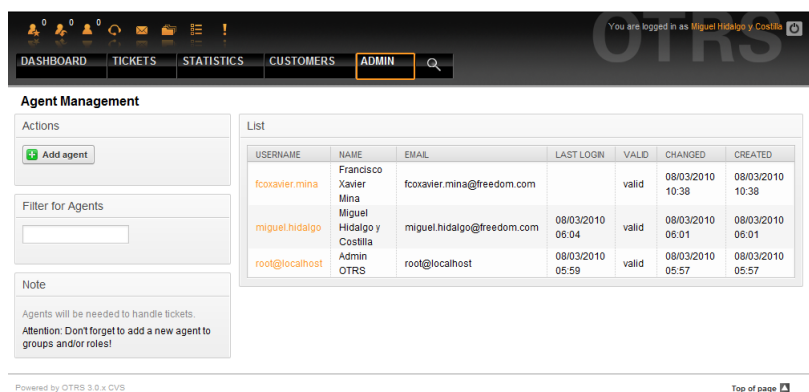


Figure: Agent management.

To register an agent, click on the "Add agent" button, type all the needed data and press the Submit button at the bottom of the screen, as shown in Figure.

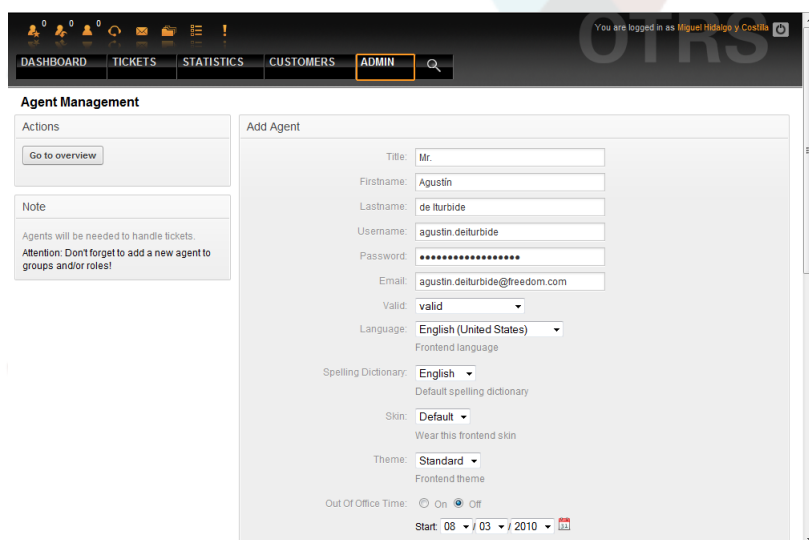


Figure: Adding a new agent.

После того, как новая учетная запись агента создана, его нужно сделать членом одной или нескольких групп или ролей. За дополнительной информацией о группах и ролях обращайтесь к разделу Группы и Роли данной главы.

2.2. Группы

Каждая учетная запись агента должна принадлежать по крайней мере к одной группе или роли. После установки системы есть три предопределенных группы, которые приведены в Таблице 5-1.

Таблица 5.1. Предопределенные группы, доступны для новой (только что) установленной системы OTRS

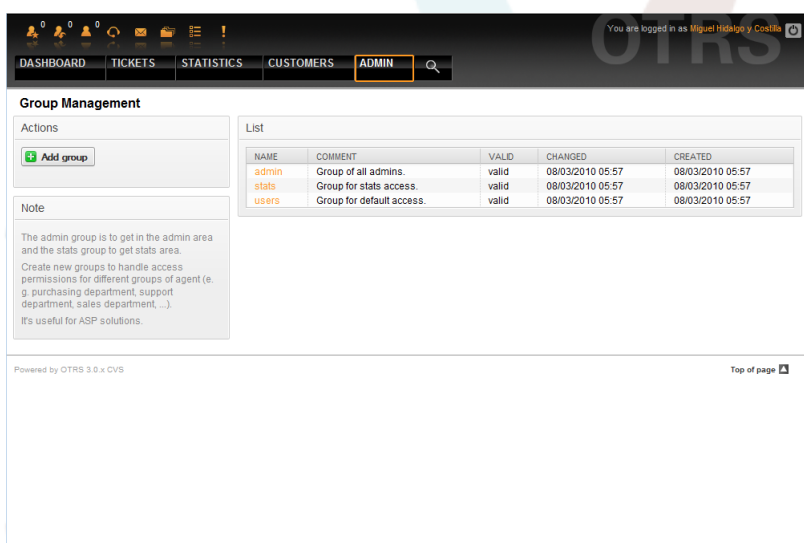
Группа	Описание
admin	Допускается для выполнения административных задач в системе.

Группа	Описание
stats	Компетентные для генерации статистики и доступа к модулю статистики OTRS.
users	Агенты должны входить в эту группу с правами для чтения и записи. Тогда они смогут получить доступ ко всем функциям системы обработки заявок.

Примечание

В новой установке OTRS, группа *users* изначально пуста. Агент 'root@localhost' по умолчанию принадлежит к группам *admin* и *stats*.

You can access the group management page (see Figure below) by clicking the *Groups* link in the admin area.



The screenshot shows the OTRS Group Management interface. At the top, there is a navigation bar with tabs for DASHBOARD, TICKETS, STATISTICS, CUSTOMERS, and ADMIN (which is highlighted). Below the navigation bar, the page title is 'Group Management'. On the left side, there is an 'Actions' section with a green 'Add group' button and a 'Note' section containing instructions on how to use the admin and stats groups. On the right side, there is a 'List' table with the following data:

NAME	COMMENT	VALID	CHANGED	CREATED
admin	Group of all admins.	valid	08/03/2010 05:57	08/03/2010 05:57
stats	Group for stats access.	valid	08/03/2010 05:57	08/03/2010 05:57
users	Group for default access.	valid	08/03/2010 05:57	08/03/2010 05:57

At the bottom of the page, it says 'Powered by OTRS 3.0.x CVS' and 'Top of page'.

Figure: Group management.

Примечание

Как в случае с агентами, группа в OTRS может быть только деактивирована, но не удалена. Деактивация осуществляется путем установки флага Действительный в значение *не действительный* или *не действительный-временно*.

To add an agent to a group, or to change the agents who belong to a group, you can use the link *Agents* <-> *Groups* from the Admin page (see Figure below).

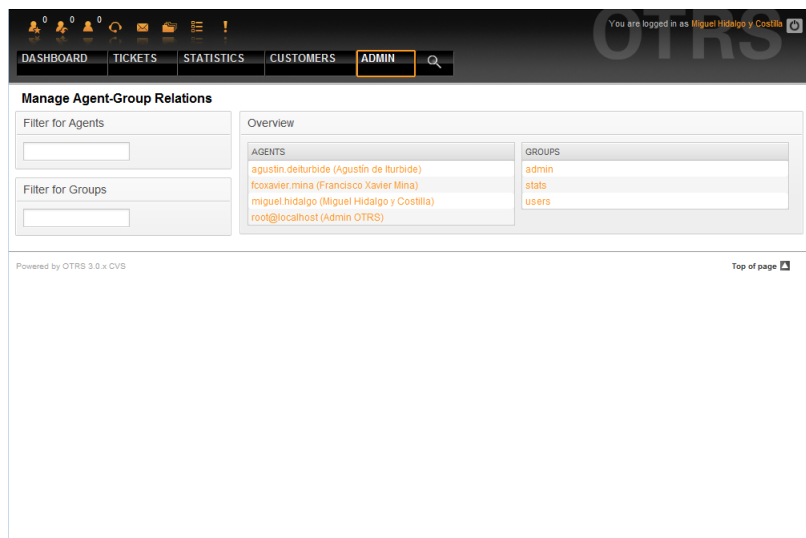


Figure: Group management.

An overview of all groups and agents in the system is displayed. You can also use the filters to find a specific entity. If you want to change the groups that an agent is member of, just click on the agent's name (see Figure below). To change the agents associated with a group, just click on the group you want to edit (see Figure below).

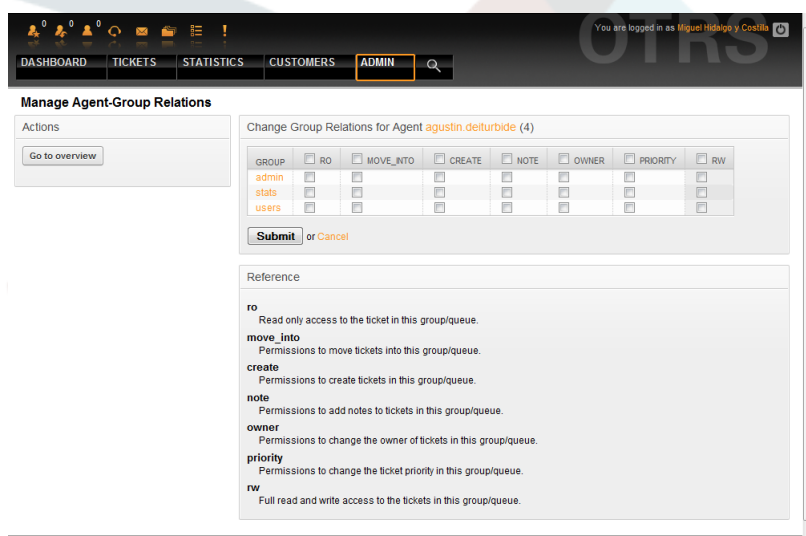


Figure: Change the groups an agent belongs to.

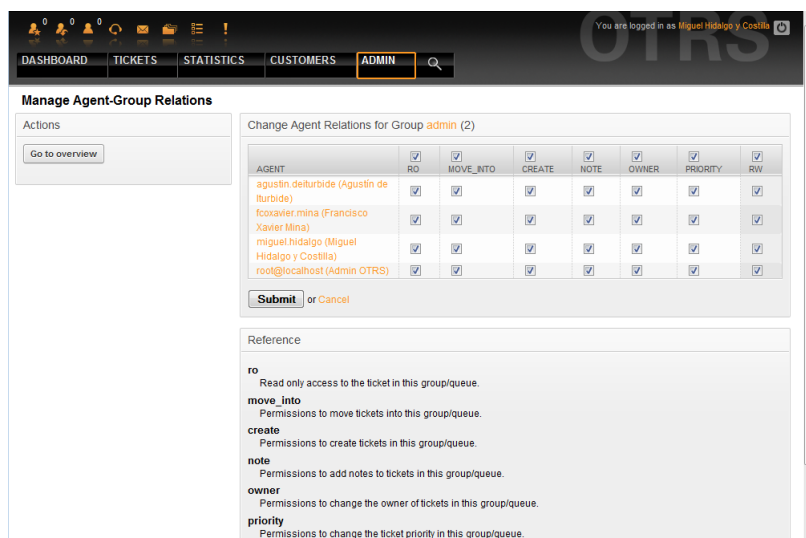


Figure: Change the agents that belong to a specific group.

Каждая группа имеет набор прав, связанный с ней и, каждый агент, член этой группы может иметь некоторую собственную комбинацию прав. Список разрешений/прав показано в Таблице 5-2.

Таблица 5.2. Права, связанные с OTRS-группами

Права	Описание
ro	Права "только для чтения" заявок, записей и очередей этой группы.
move into	Права для перемещения заявок или записей между очередями или для области, принадлежащей группе.
create	Права для создания заявок или записей в очередях или области группы.
owner	Права для изменения владельца заявок или записей в очередях принадлежащих этой группе.
priority	Права для изменения приоритета заявок или записей в очереди или области, принадлежащей этой группе.
rw	Полный доступ к заявкам или записям в очереди или области, принадлежащей к этой группе.

Примечание

По умолчанию, Список Очередей (QueueView) отображает только заявки в очередях, к которым агент имеет права *rw*, т.е к заявкам, с которыми он должен работать. Если вы хотите изменить это поведение, вы можете установить параметр `Ticket::Frontend::AgentTicketQueue###ViewAllPossibleTickets` в значение *Да*.

2.3. Роли

Роли являются мощным средством для простого и быстрого управления правами доступа множества агентов. Они особенно полезны для больших и сложных систем

поддержки, с большим количеством агентов, групп и очередей. В приведенном ниже примере объясняется как и когда они могут использоваться.

Давайте представим, что у нас есть система, в которой зарегистрировано 100 агентов, 90 из них имеют доступ только к очереди "поддержка", в которой хранятся все запросы о поддержке. Очередь "поддержка" содержит несколько под очередей. Остальные 10 агентов имеют права доступа ко всем очередям, существующим в системе. Эти 10 агентов работают с заявками, просматривают "raw"-очередь и перемещают спам-сообщения в очередь "junk".

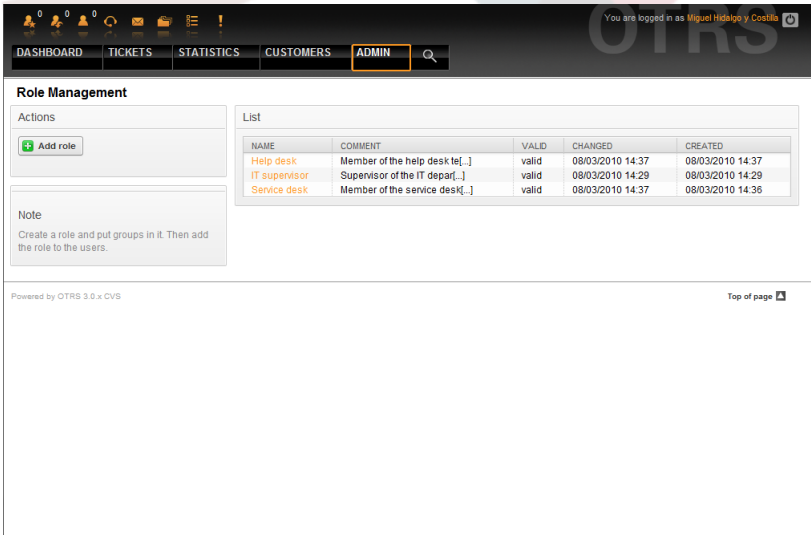
В один день компания открывает новый отдел для продажи товаров. Запрос/принятие заказа, подтверждение, счета и т.д. должны быть обработаны, и некоторые из агентов компании должны делать это используя OTRS. Разным агентам нужно предоставить доступ к очередям, которые планируются создать.

Поскольку изменения вручную прав доступа для разных агентов займет много времени, то для определения различных уровней доступа можно создать роли. Затем, агенты могут быть добавлены к одной или нескольким ролям, таким образом их права будут изменены автоматически. Если будет создана новая учетная запись агента, ее также можно с легкостью добавить к одной или нескольким ролям.

Примечание

Роли действительно полезны для управления большими OTRS--системами. Однако, используя роли, нужно проявлять осторожность. Смешивание Агента входящего в группу с Агентом входящим в Роль может сделать схему контроля сложной для понимания и управления. Если вы решили использовать только роли и отключить опцию Агенты <-> Группы в Панели Администрирования, то это можно сделать изменив параметр `Frontend::Module###AdminUserGroup` в `SysConfig`. Помните, это действие не удалит существующих Агентов из Групп, в которых они находятся!

You can access the role management section (see Figure below) by clicking the *Roles* link on the Admin page.



The screenshot shows the OTRS Admin interface. At the top, there is a navigation bar with tabs for DASHBOARD, TICKETS, STATISTICS, CUSTOMERS, and ADMIN (which is highlighted). Below the navigation bar, the 'Role Management' section is visible. On the left, there is an 'Actions' panel with a green 'Add role' button and a 'Note' box that says 'Create a role and put groups in it. Then add the role to the users.' On the right, there is a 'List' table with the following data:

NAME	COMMENT	VALID	CHANGED	CREATED
Help desk	Member of the help desk []	valid	08/03/2010 14:37	08/03/2010 14:37
IT supervisor	Supervisor of the IT depart []	valid	08/03/2010 14:29	08/03/2010 14:29
Service desk	Member of the service desk []	valid	08/03/2010 14:37	08/03/2010 14:36

At the bottom of the page, it says 'Powered by OTRS 3.0.x CVS' and 'Top of page'.

Figure: Role management.

Примечание

Как в случае с агентом и группами, однажды созданные роли могут быть деактивированы, но не удалены. Чтобы отключить, установите опцию Действительный в значение *не действительный* или *не действительный-временно*.

An overview of all roles in the system is displayed. To edit a role's settings, click on the role's name. In a fresh new OTRS installation, there are no roles defined by default. To register one, click on the "Add role" button, provide the needed data and submit it (see Figure below).

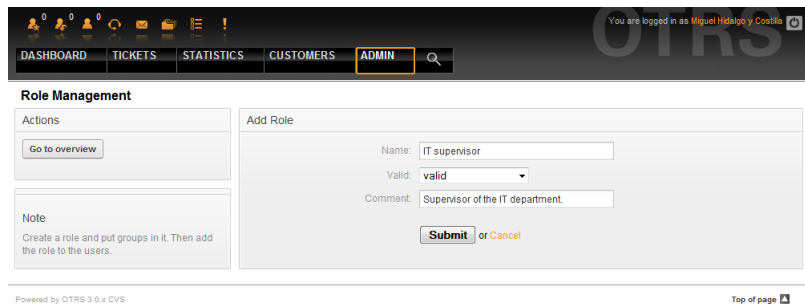


Figure: Adding a new role.

To get an overview of all roles and agents in the system, click on the link Roles <-> Agents on the Admin page. You can also use filters to find a specific element. If you want to change the roles associated with an agent, just click on the agent's name (see Figure below). To change the agents associated with a role, click on the role you want to edit (see Figure below).

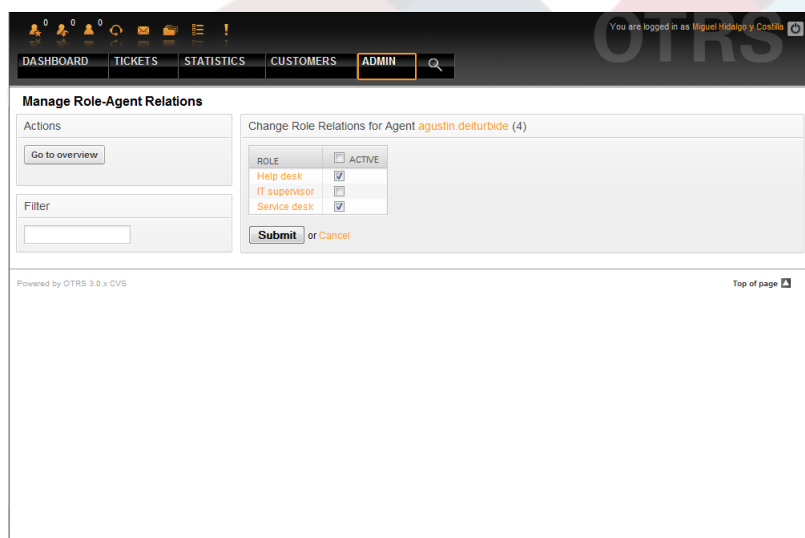


Figure: Change the Roles associated with an Agent.

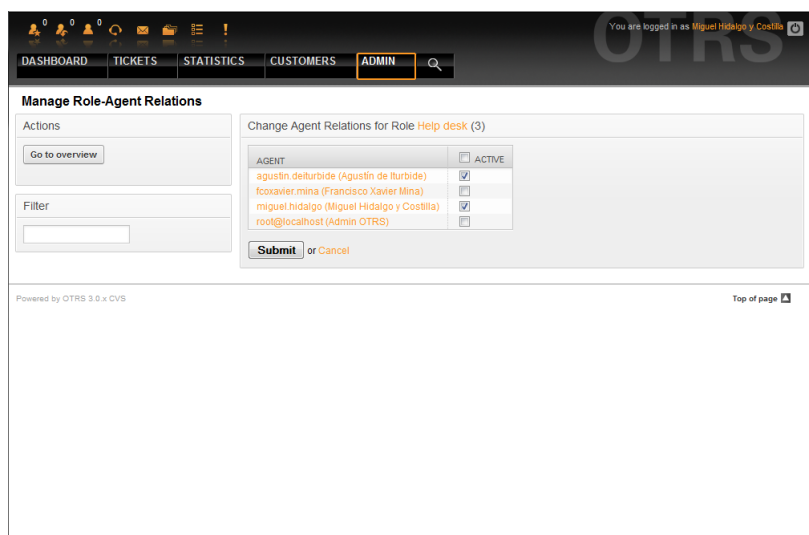


Figure: Change the Agents associated with a specific Role.

To get an overview of all roles and groups in the system, click on the link Roles <-> Groups on the Admin page. You will see a similar screen as the one shown in the Figure. You can also use filters to find a specific entity.

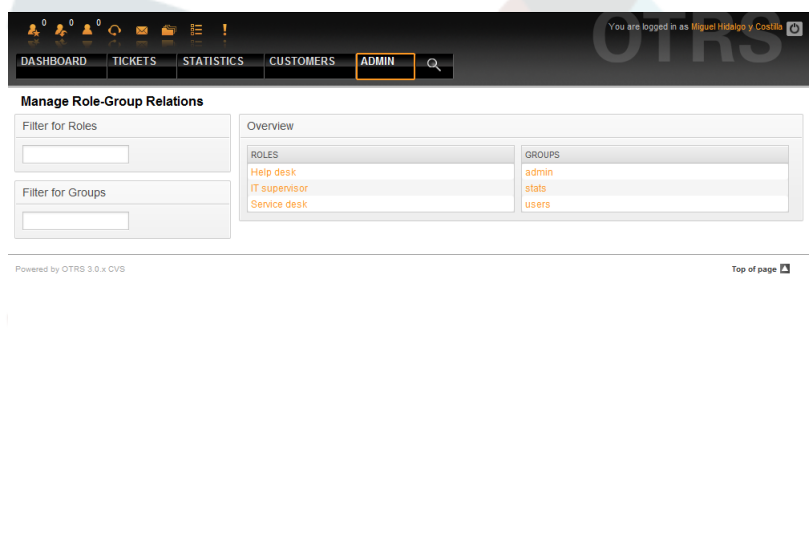
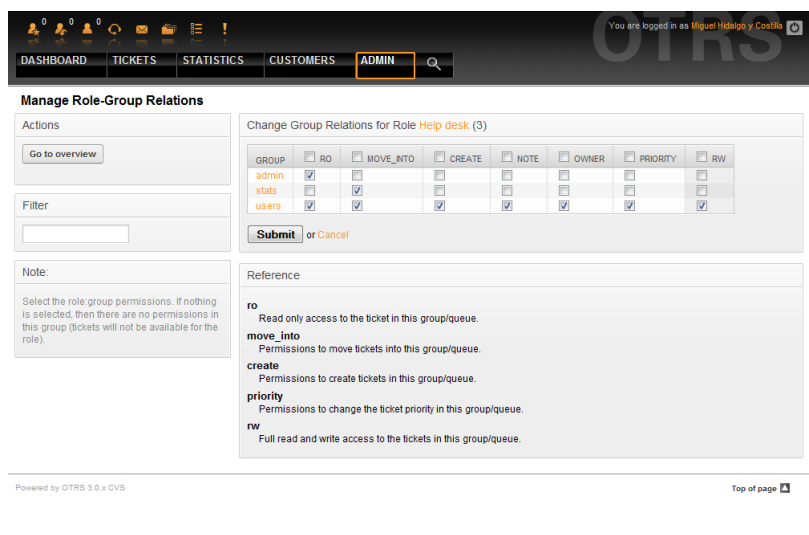


Figure: Manage Roles-Groups relations.

Чтобы определить различные права доступа для роли, нажмите на имя роли или группы (см. ниже Рисунок 5.13 и 5.14, соответственно).



Manage Role-Group Relations

Change Group Relations for Role **Help desk** (3)

GROUP	<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
admin	<input checked="" 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 checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
users	<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"/>

or

Note:
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).

Reference

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

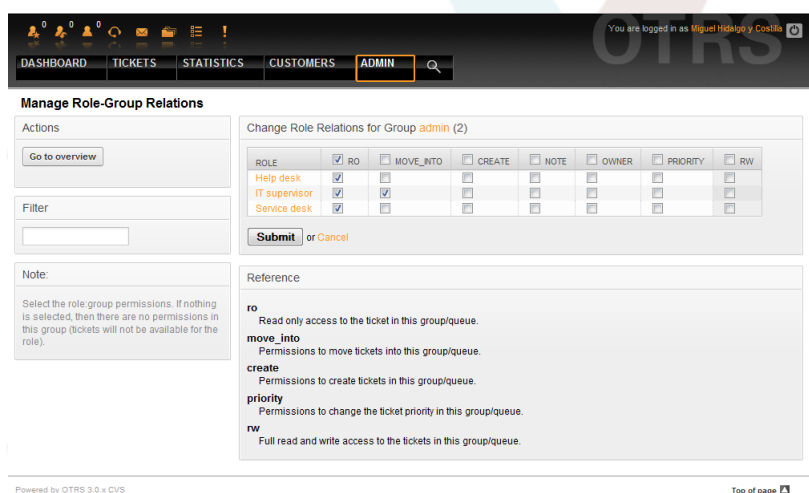
move_into
Permissions to move tickets into this group/queue.

create
Permissions to create tickets in this group/queue.

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

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

Figure: Change Group relations for a Role.



Manage Role-Group Relations

Change Role Relations for Group **admin** (2)

ROLE	<input checked="" 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
Help desk	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
IT supervisor	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Service desk	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

or

Note:
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).

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.

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

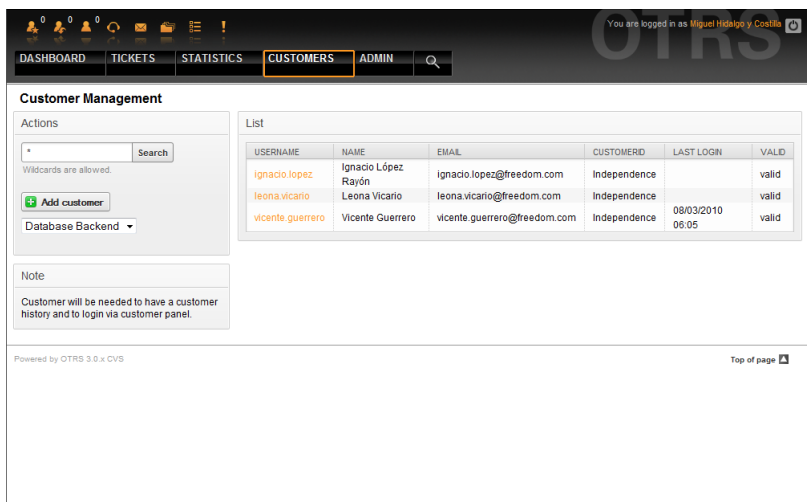
rw
Full read and write access to the tickets in this group/queue.

Figure: Change Role relations for a Group.

3. Клиенты и Группы Клиентов

3.1. Клиенты

OTRS supports different types of users. Using the link "Customers" (via the navigation bar, or the Admin page), you can manage the accounts of your customers (see Figure below), who can log into the system via the Customers interface (customer.pl). Through this interface, your customers can create tickets and access them as they are updated. It is important to know that a customer is needed for the ticket history in the system.



The screenshot shows the OTRS Customer Management interface. At the top, there is a navigation bar with tabs for DASHBOARD, TICKETS, STATISTICS, CUSTOMERS (selected), and ADMIN. Below the navigation bar, the page title is "Customer Management". On the left, there is an "Actions" section with a search box, a note that "Wildcards are allowed.", an "Add customer" button, and a "Database Backend" dropdown menu. Below this is a "Note" box stating: "Customer will be needed to have a customer history and to login via customer panel." The main area contains a "List" table with the following data:

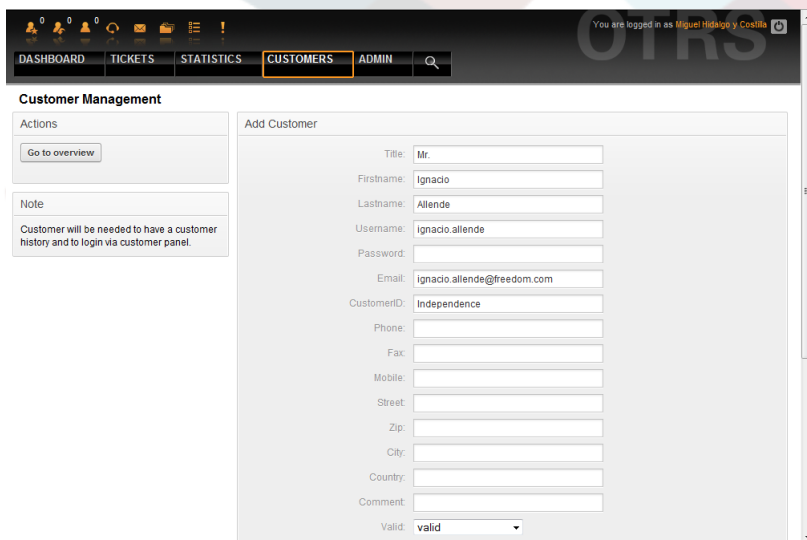
USERNAME	NAME	EMAIL	CUSTOMERID	LAST LOGIN	VALID
ignacio.lopez	Ignacio López Rayón	ignacio.lopez@freedom.com	Independence		valid
leona.vicario	Leona Vicario	leona.vicario@freedom.com	Independence		valid
vicente.guerrero	Vicente Guerrero	vicente.guerrero@freedom.com	Independence	08/03/2010 08:05	valid

At the bottom of the page, it says "Powered by OTRS 3.0.x CVS" and "Top of page".

Figure: Customer management.

Нажав на имя клиента, по зарегистрированным клиентам можно производить поиск, редактировать их данные. Также есть возможность изменить клиентский backend, за более подробной информацией обращайтесь к главе внешние хранилища данных.

To create a new customer account, click on the "Add customer" button (see Figure below). Some of the fields are mandatory, i.e., they have to contain values, so if you leave one of those empty, it will be highlighted in red.



The screenshot shows the OTRS Customer Management interface with the "Add Customer" form open. The form contains the following fields:

- Title: Mr.
- Firstname: Ignacio
- Lastname: Allende
- Username: ignacio.allende
- Password: (empty)
- Email: ignacio.allende@freedom.com
- CustomerID: Independence
- Phone: (empty)
- Fax: (empty)
- Mobile: (empty)
- Street: (empty)
- Zip: (empty)
- City: (empty)
- Country: (empty)
- Comment: (empty)
- Valid: valid

At the bottom of the form, there is a "Valid" dropdown menu set to "valid".

Figure: Adding a customer.

Клиент может получить доступ к системе путем ввода логина и пароля. CustomerID нужен для идентификации клиента и его заявок. Поскольку адрес электронной почты является уникальным, его можно использовать в качестве уникального идентификатора (ID).

Примечание

Как и в случае с агентами, группами и ролями, система не позволяет удалять клиентов, а только деактивировать их, установив опцию Действительный в значение *недействительный* или *недействительный-временно*.

3.2. Группы Клиентов

Клиенты также могут быть добавлены в группу, что может быть полезно, если вы хотите объединить нескольких клиентов той же компании для доступа к одной или нескольким очередям. Сначала создайте группу, в которую будут входить клиенты, для этого перейдите по ссылке Модуль Управления Группами . Затем добавьте очереди и выберите новую группу для этих очередей.

Следующим шагом будет включение поддержки клиентов группы. Это можно сделать используя конфигурационный параметр CustomerGroupSupport в SysConfig. Используя параметр CustomerGroupAlwaysGroups можно указать группы по умолчанию, к которым будет принадлежать новый пользователь, таким образом каждая новая учетная запись будет добавляться в эту группу.

Through the link "Customers <-> Groups" you can manage which customer shall belong to the different groups (see Figure below).

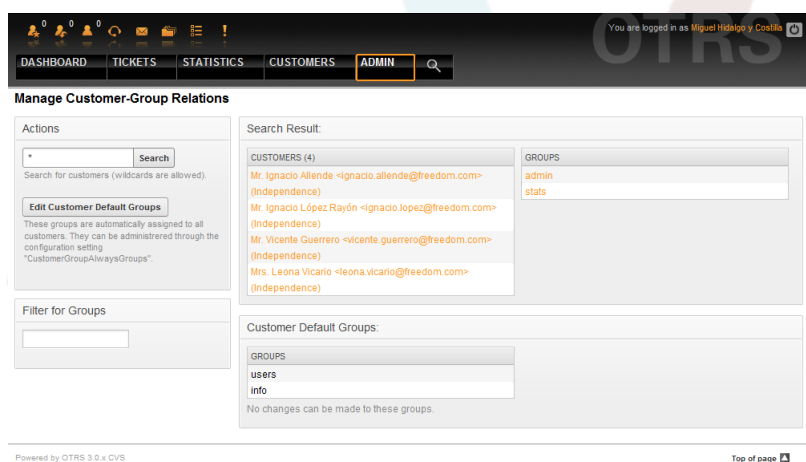


Figure: Customer-Group relations management.

Для определения различных групп, в которые должен входить клиент, и наоборот, нажмите на соответствующее имя пользователя или группы (см. ниже Рисунок 5.16 и 5.17, соответственно).

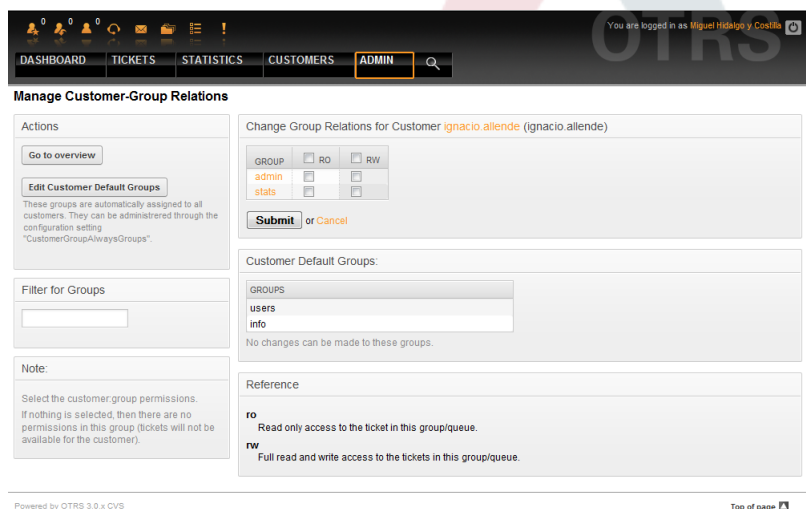
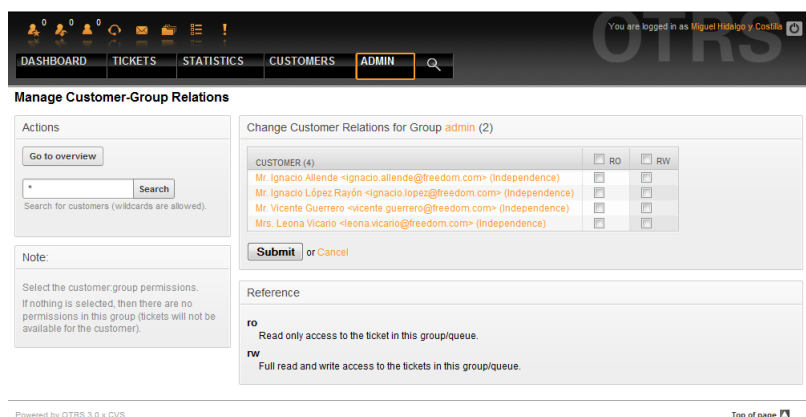


Figure: Change Group relations for a Customer.

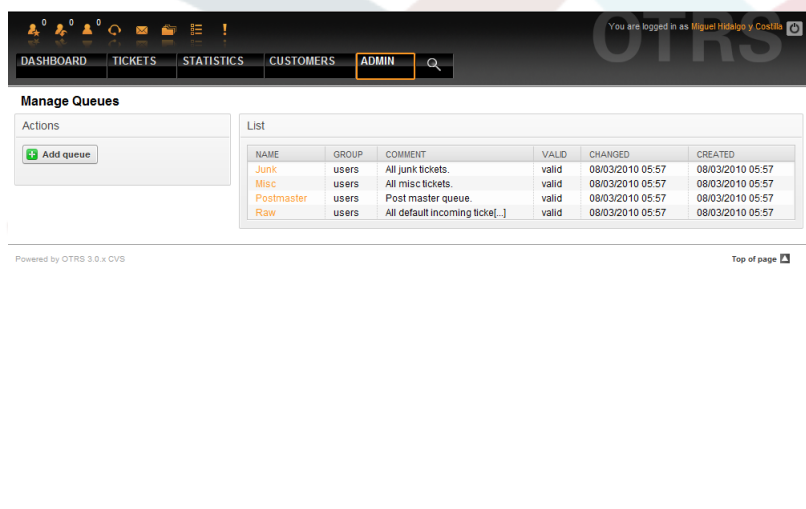


The screenshot shows the 'Manage Customer-Group Relations' page in OTRS. The top navigation bar includes 'DASHBOARD', 'TICKETS', 'STATISTICS', 'CUSTOMERS', and 'ADMIN'. The page title is 'Manage Customer-Group Relations'. On the left, there are 'Actions' (Go to overview, Search) and a 'Note' section. The main content area is titled 'Change Customer Relations for Group admin (2)'. It contains a table with columns for 'CUSTOMER (4)', 'RO', and 'RW'. The table lists four customers: Mr. Ignacio Allende, Mr. Ignacio López Rayón, Mr. Vicente Guerrero, and Mrs. Leona Vicario. Below the table are 'Submit' and 'Cancel' buttons. A 'Reference' section explains the permissions: 'ro' (Read only access to the ticket in this group/queue) and 'rw' (Full read and write access to the tickets in this group/queue).

Figure: Change Customer relations for a Group.

4. Очереди

Clicking on the link "Queues" of the Admin page, you can manage the queues of your system (see Figure below). In a new OTRS installation there are 4 default queues: Raw, Junk, Misc and Postmaster. All incoming messages will be stored in the "Raw" queue if no filter rules are defined. The "Junk" queue can be used to store spam messages.

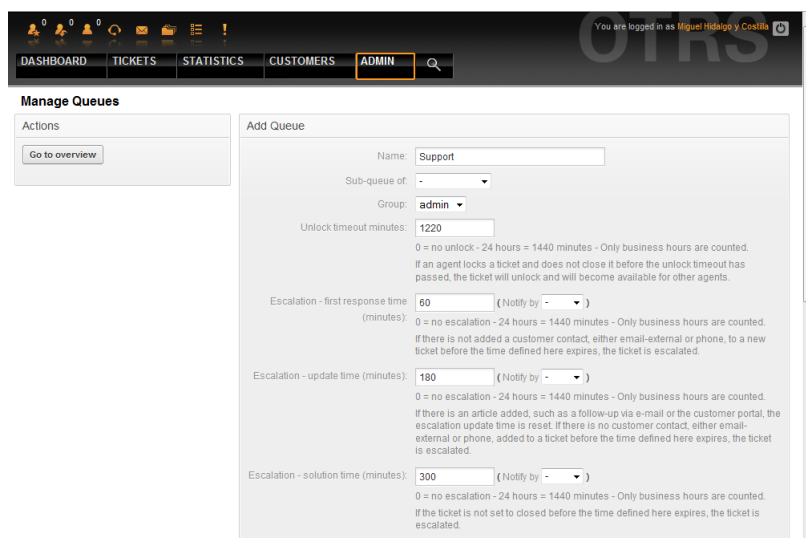


The screenshot shows the 'Manage Queues' page in OTRS. The top navigation bar includes 'DASHBOARD', 'TICKETS', 'STATISTICS', 'CUSTOMERS', and 'ADMIN'. The page title is 'Manage Queues'. On the left, there are 'Actions' (Add queue) and a 'List' section. The main content area is titled 'List' and contains a table with columns for 'NAME', 'GROUP', 'COMMENT', 'VALID', 'CHANGED', and 'CREATED'. The table lists four queues: Junk, Misc, Postmaster, and Raw.

NAME	GROUP	COMMENT	VALID	CHANGED	CREATED
Junk	users	All junk tickets.	valid	08/03/2010 05:57	08/03/2010 05:57
Misc	users	All misc tickets.	valid	08/03/2010 05:57	08/03/2010 05:57
Postmaster	users	Post master queue.	valid	08/03/2010 05:57	08/03/2010 05:57
Raw	users	All default incoming ticket[...]	valid	08/03/2010 05:57	08/03/2010 05:57

Figure: Queue management.

Here you can add queues (see Figure below) and modify them. You can specify the group that should use the queue. You can also set the queue as a sub-queue of an existing queue.



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 e-mail 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.

Figure: Adding a new queue.

Для очереди можно определить время разблокировки - если агент заблокировал заявку и не закрыл ее до истечения времени блокировки, заявка будет автоматически разблокирована и доступна для других агентов.

Есть три типа настроек эскалации времени, которые могут быть связаны на уровне очередей:

Эскалация - Время Первого Отклика

- После создания заявки, если истекает оговоренное время без какой-либо связи с клиентом как по электронной почте так и по телефону, - заявка будет эскалирована.

Эскалация - Время обновления

- Если какой-нибудь клиент прислал ответ на заявку по электронной почте или изменил ее через пользовательский портал, время эскалации будет обнулено. Если до истечения времени ожидания контакта с клиентом не было, заявка эскалируется.

Эскалация - Время решения

- Если до истечения оговоренного времени заявка не закрыта - она эскалируется.

Используя опцию 'Ticket lock after a follow-up', можно определить будет ли заявка установлена в значение 'заблокирована' для старого владельца, если она была закрыта а потом открыта заново. Такое поведение гарантирует, что последующая заявка будет обрабатываться агентом, который работал с этой заявкой ранее.

Параметром системный адрес определяется адрес электронной почты, который будет использоваться для исходящих заявок этой очереди. Также, для ответов по электронной почте, есть возможность ассоциировать (связать) очереди с приветствиями и подписями. За более подробной информацией перейдите по ссылке адреса электронной почты , приветствия и подписи .

Примечание

Как в случае с агентами, группами и клиентами, очереди не могут быть удалены, а только отключены, путем установки опции "Действительная" в значение *недействительная* или *недействительная-временно*.

5. Приветствия, подписи, вложения и ответы

5.1. Приветствия

A salutation is a text module for a response. Salutations can be linked to one or more queues, as described in the section about queues. A salutation is used only if a ticket from a queue the salutation is linked to, is answered. To manage the different salutations of your system, use the "Salutations" link of the admin area (see Figure below).

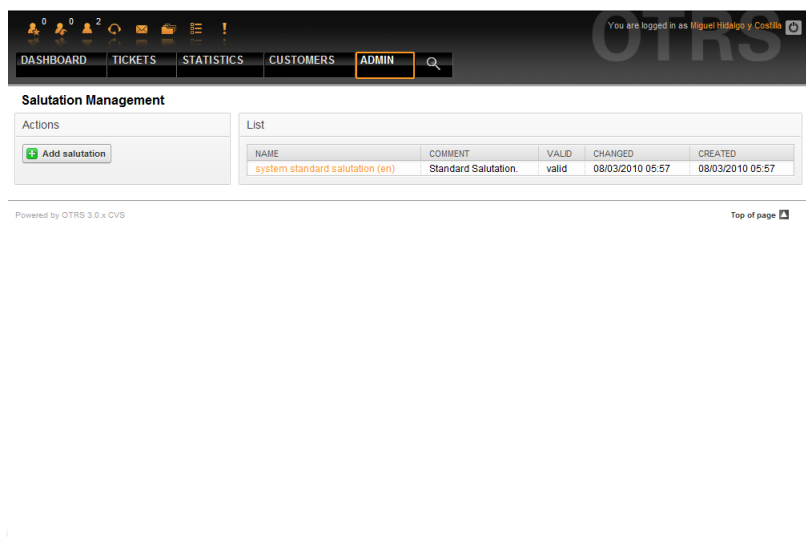


Figure: Salutation management.

После установки в системе есть одно предопределенное приветствие: "system standard salutation (en)".

To create a new salutation, press the button "Add salutation", provide the needed data and submit it (see Figure below).

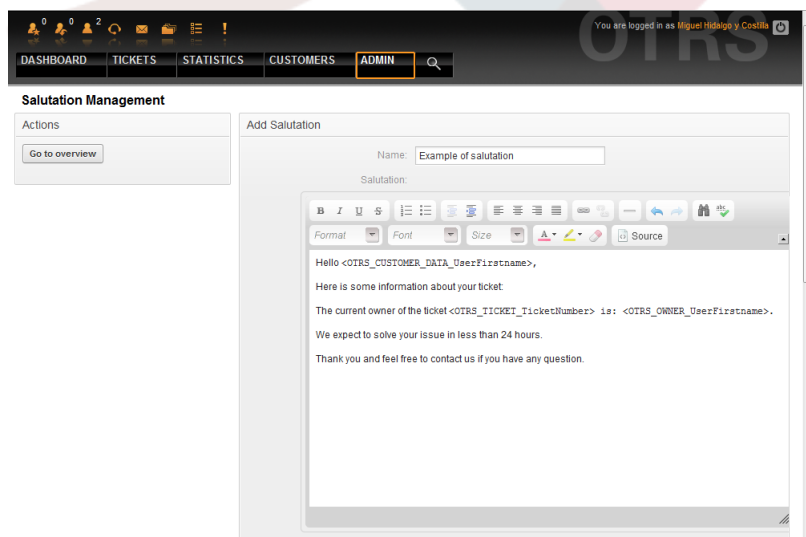


Figure: Adding a new salutation.

В приветствиях можно использовать переменные. Когда вы отвечаете на заявку, имена переменных будут заменены на их значения.

Различные переменные, которые можно использовать в ответах приведены в нижней части экрана приветствия. Если вы используете, например, переменную <OTRS_LAST_NAME> то в ответ будет включена фамилия отправителя заявки.

Примечание

Как и в случае с другими сущностями системы OTRS, приветствия не можно удалять, только деактивировать их, установив параметр Действительный в значение *не действительный* или *не действительный-временно*.

5.2. Подписи

Another text module for a response is the signature. Signatures can be linked to a queue, as described in the section about the queues. Only if a signature is linked to a queue will it be included into the response text. Through the "Signatures" link of the Admin page, you can manage the signatures in your system (see Figure below).

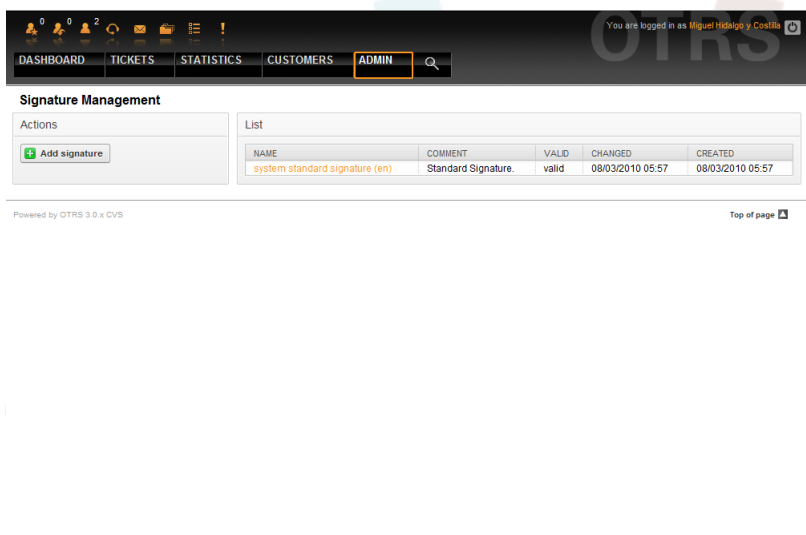


Figure: Signatures management.

В новой, только установленной системе OTRS есть одна предустановленная подпись: "system standard signature (en)".

To create a new signature, press the button "Add signature", provide the needed data and submit it (see Figure below).

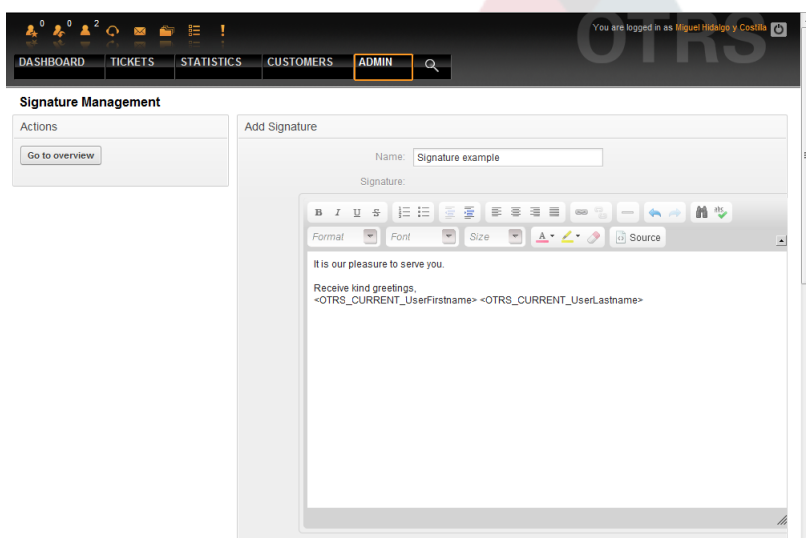


Figure: Adding a new signature.

Как и приветствия, подписи также могут содержать динамический контент, например, имя и фамилию агента, который отвечает на заявку, а также могут использо-

ваться переменные для замены текста подписи каждой заявки. Переменные, которые можно использовать ищите в нижней части страницы с подписями. Если вы, к примеру, включаете переменную <OTRS_LAST_NAME> то она будет заменена фамилией агента, который отвечает на заявку.

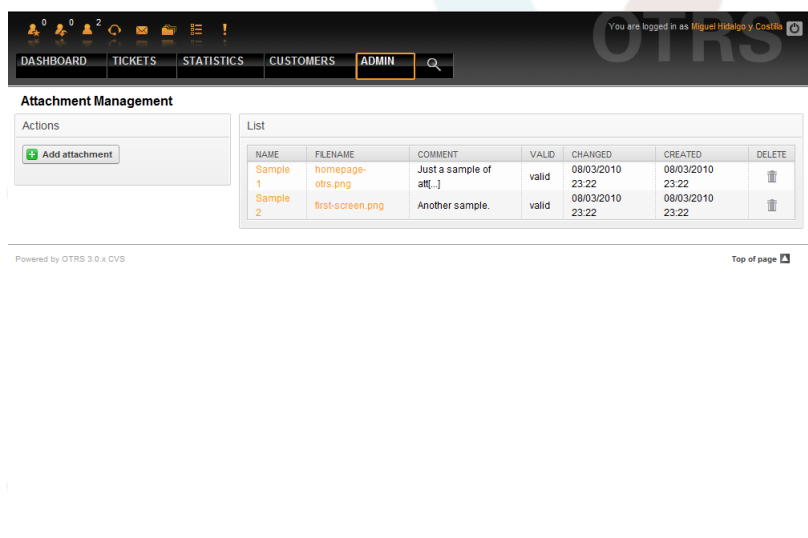
Примечание

Как и в случае с приветствиями, подписи не могут быть удалены, только деактивированы путем установки параметра Действительный в значения *не действительный* или *не действительный-временно*.

5.3. Вложения

При желании к ответу можно добавить одно или несколько вложений. Если ответ выбран, вложения будут прикреплены к сообщению. При необходимости, непосредственно перед отправкой ответа клиенту, агент может удалить вложения.

Through the "Attachment" link of the Admin page, you can load the attachments into the database of the system (see Figure below).

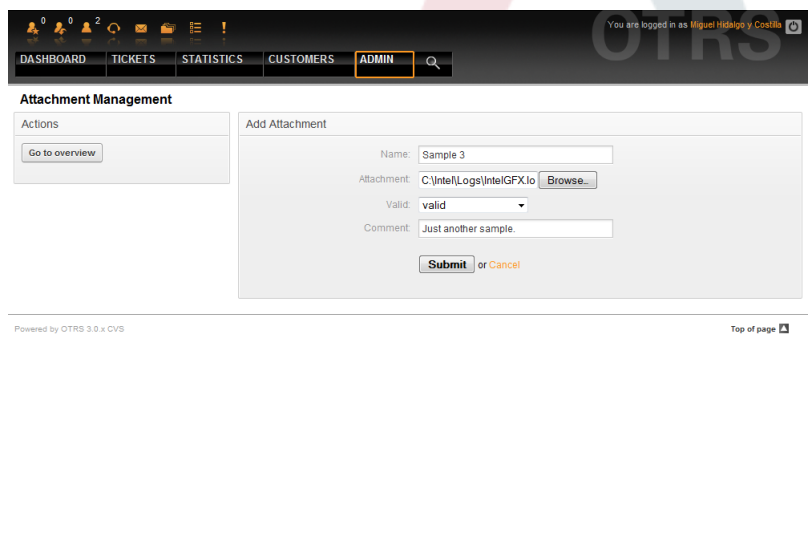


The screenshot shows the OTRS Admin interface for Attachment Management. The top navigation bar includes DASHBOARD, TICKETS, STATISTICS, CUSTOMERS, and ADMIN. The main content area is titled "Attachment Management" and features an "Add attachment" button. Below this is a table with the following data:

NAME	FILENAME	COMMENT	VALID	CHANGED	CREATED	DELETE
Sample 1	homepage-otrs.png	Just a sample of att[...]	valid	08/03/2010 23:22	08/03/2010 23:22	[Delete icon]
Sample 2	first-screen.png	Another sample.	valid	08/03/2010 23:22	08/03/2010 23:22	[Delete icon]

Figure: Attachments management.

To create a new attachment, press the button "Add attachment", provide the needed data and submit it (see Figure below).



The screenshot shows the OTRS Admin interface for adding a new attachment. The "Add Attachment" form is displayed with the following fields:

- Name: Sample 3
- Attachment: C:\Intel\Logs\Intel\GFX.log
- Valid: valid
- Comment: Just another sample.

The "Submit" button is highlighted in red, indicating it is the next step to create the attachment.

Figure: Adding a new attachment.

If an attachment is stored it can be linked to one or more responses. Click on the "Attachment <-> Responses" link of the Admin page (see Figure below).

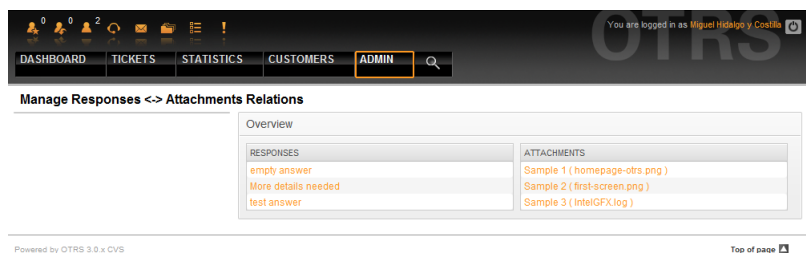


Figure: Linking Attachments to Responses.

Чтобы связать различные вложения с конкретным ответом и наоборот, нажмите на название соответствующего ответа или вложения (см. ниже Рисунки 5.27 и 5.28, соответственно).

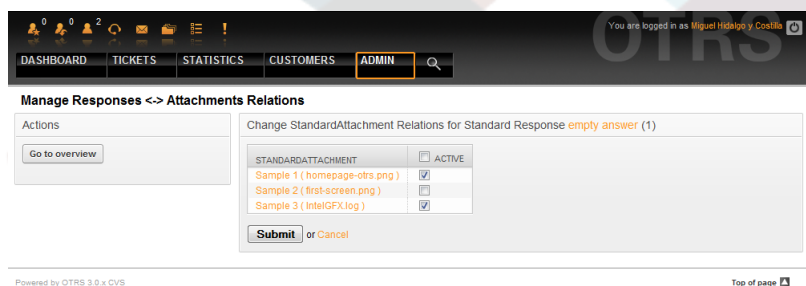


Figure: Change Attachment relations for a Response.

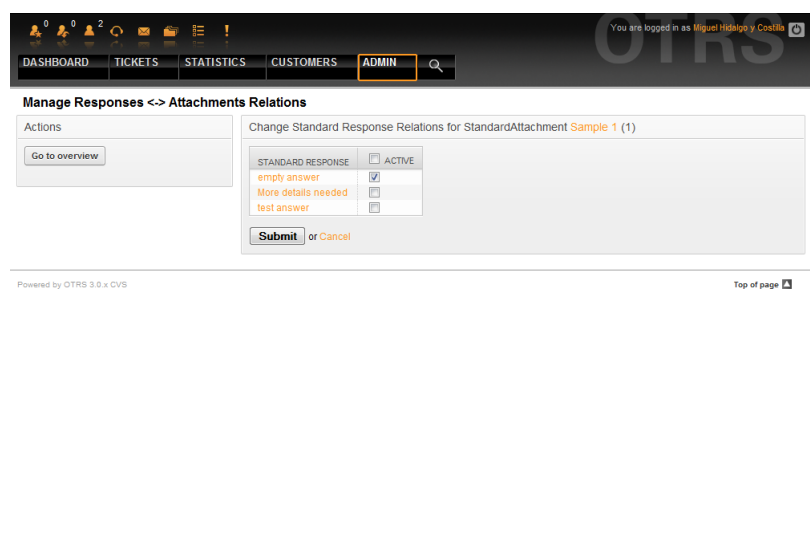


Figure: Change Response relations for an Attachment.

5.4. Ответы

В OTRS можно определить ответы для того, чтобы быстрее отвечать на заявки и как то стандартизировать вид ответов. Ответ может быть связан с одной или несколькими очередями и наоборот. Для того чтобы можно было быстро использовать заранее заготовленные ответы, различные ответы отображаются в очереди "Мои Очереди", внизу под каждой заявкой.

On a fresh OTRS installation, the "empty answer" response is defined for every queue. Clicking the "Responses" link on the Admin page brings you to the Responses management page (see Figure below).

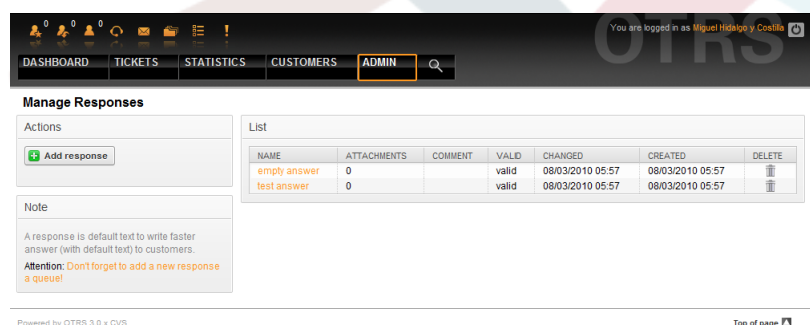


Figure: Responses management.

To create a new response, click on the "Add response" button, provide the required data and submit it (see Figure below).

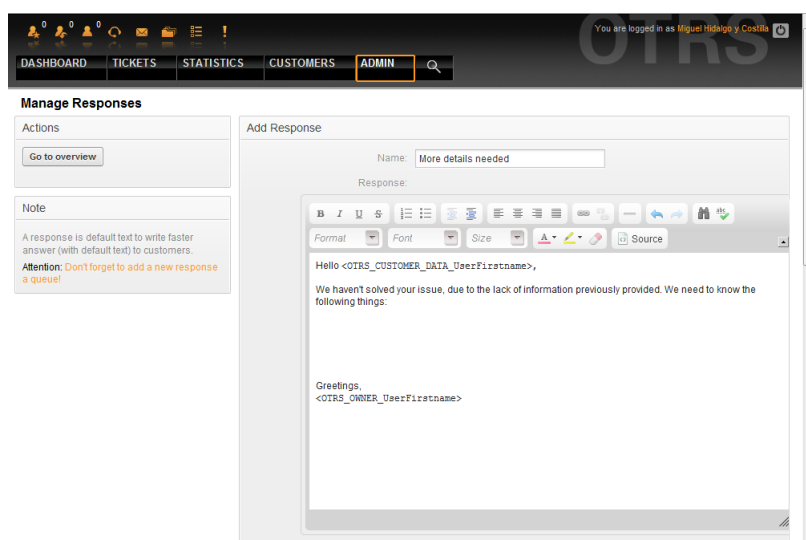


Figure: Adding a response.

To add/remove responses to one or more queues, click on the "Responses <-> Queues" link on the Admin page (see Figure below). You can also use filters to get information on a specific entity.

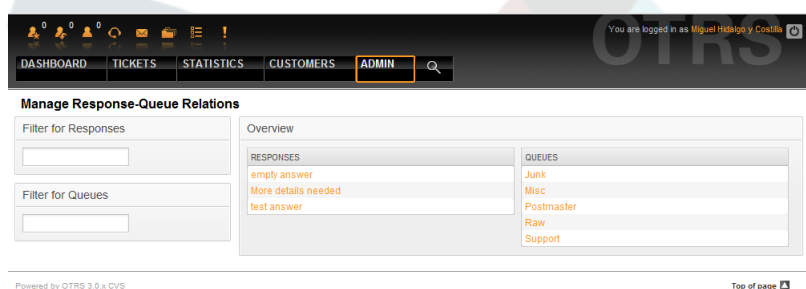


Figure: Response-Queue relations management.

Чтобы определить различные ответы для очереди и наоборот, нажмите на соответствующий ответ или очередь (см. ниже Рисунко 5.32 и 5.33, соответственно).

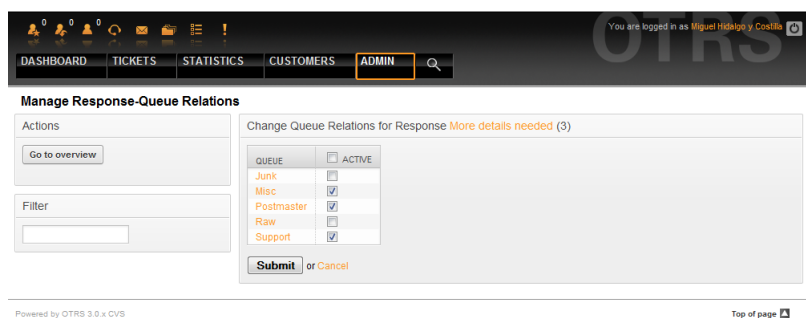


Figure: Change Queue relations for a Response.

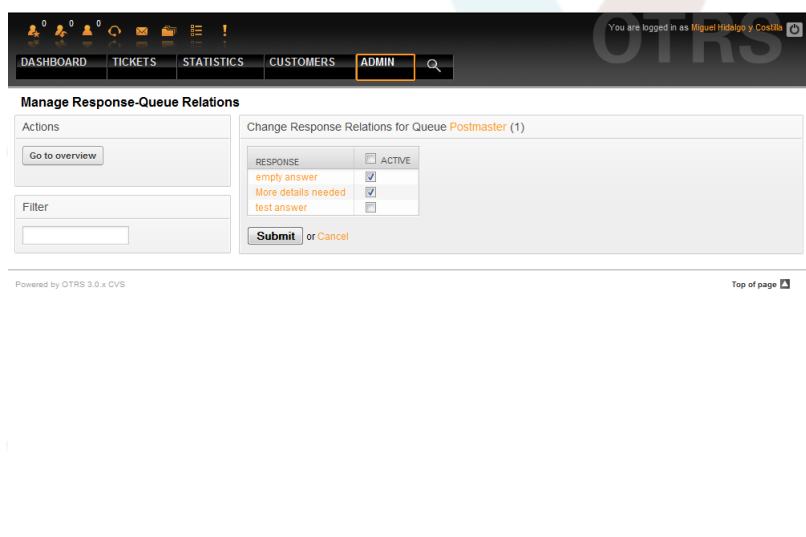


Figure: Change Response relations for a Queue.

Структура ответа понятна на интуитивном уровне. Она состоит из приветствия, которое ассоциируется с очередью, потом идет текст ответа, затем цитируемый текст заявки и, наконец, подпись, связанная с очередью.

6. Авто ответы

OTRS allows you to send automatic responses to customers on the occurrence of certain events, such as the creation of a ticket in certain queue, reception of a follow-up message on a ticket, closure or rejection of a ticket, etc. To manage such responses, click the link "Auto responses" on the Admin page (see Figure below).

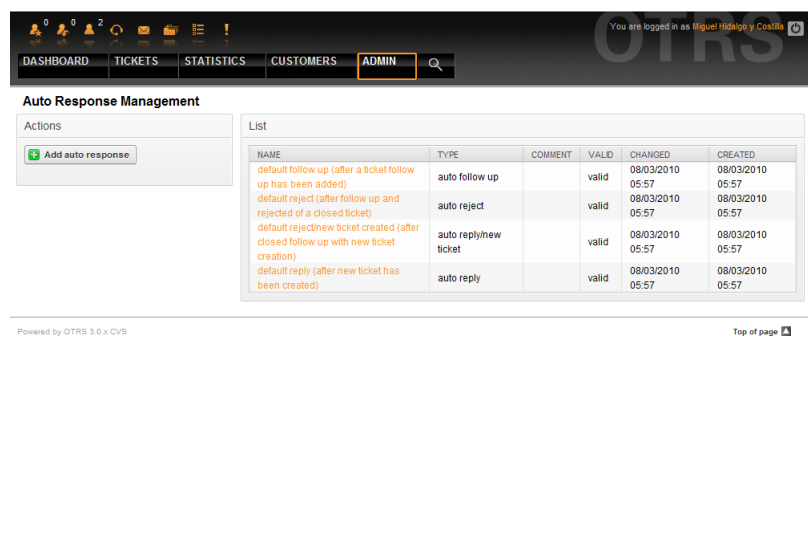


Figure: Auto Response management.

To create an automatic response, click on the button "Add auto response", provide the needed data and submit it (see Figure below).

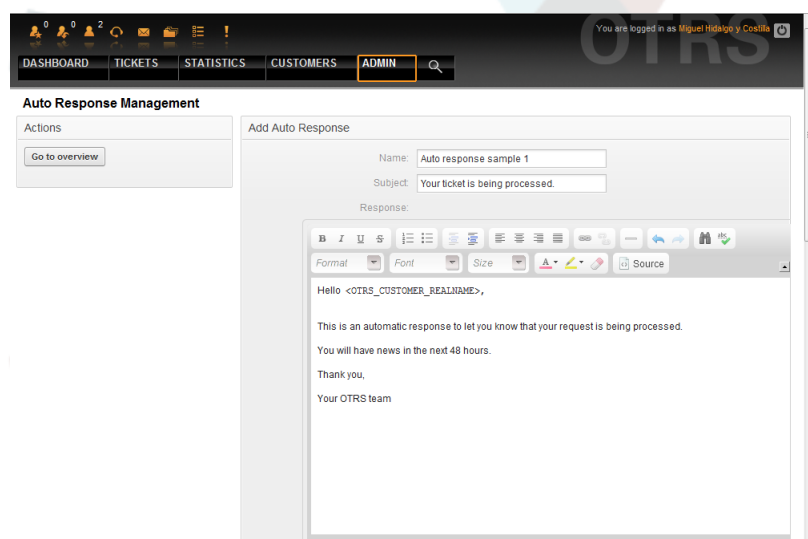


Figure: Adding an Auto Response.

The subject and text of auto responses can be generated by variables, just as in signatures and salutations. If you insert, for example, the variable `<OTRS_CUSTOMER_EMAIL[5]>` into the body of the auto answer, the first 5 lines of the customer mail text will be inserted into the auto answer. You will find more details about the valid variables that can be used at the bottom of the screen shown in the Figure.

Для каждого авто-ответа можно указать специальное событие, при наступлении которого этот авто-ответ будет отправляться. Предопределенные события, доступные после новой установки приведены в Таблицу 5-3.

Таблица 5.3. События для Автоматических ответов

Название	Описание
auto reply	Создание заявки в определенной очереди.

Название	Описание
auto reply/new ticket	Открытие уже закрытой заявки, например если клиент отвечает на определенную заявку.
auto follow up	Прием входящих заявок.
auto reject	Автоматический отказ от заявки, сделанный системой.
auto remove	Удаление заявок, завершенных системой.

Примечание

Как и в случае с другими сущностями OTRS, Авто ответы также не могут быть удалены, только деактивированы путем установки опции Действительный в значения *не действительный* или *не действительный-временно*.

To add an auto response to a queue, use the "Auto Response <-> Queues" link on the Admin page (see Figure below). All system events are listed for every queue, and an auto answer with the same event can be selected or removed via a listbox.

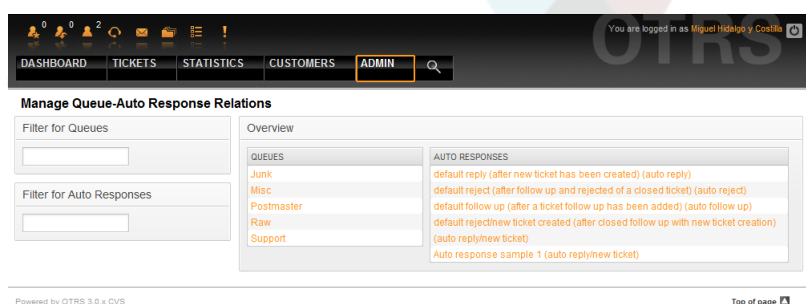


Figure: Queue-Auto Response relations management.

To define the different auto responses that will be available for a queue, click on the corresponding queue name (see Figure below). It is also possible to edit an existing auto response - to do so, click on the response and edit in the same manner as editing a new auto response.

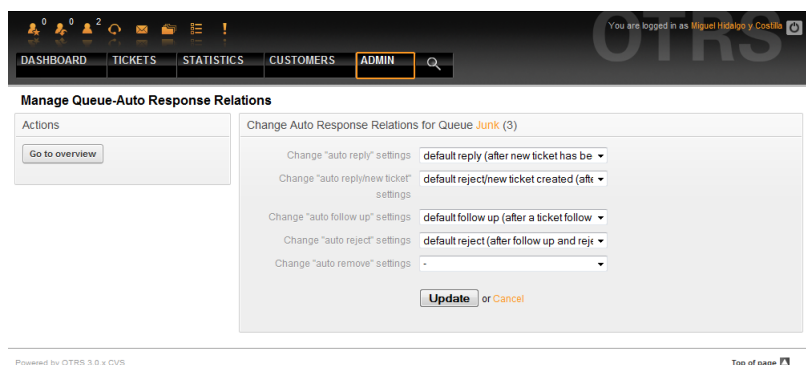


Figure: Change Auto Response relations for a Queue.

7. Адреса электронной почты

To enable OTRS to send emails, you need a valid email address to be used by the system. OTRS is capable of working with multiple email addresses, since many support installations need to use more than one. A queue can be linked to many email addresses, and vice versa. The address used for outgoing messages from a queue can be set when the queue is created. Use the "Email Addresses" link from the Admin page to manage all email addresses of the system (see Figure below).

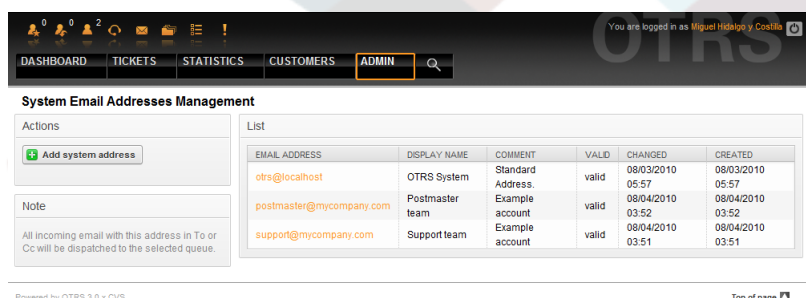


Figure: System Email Addresses management.

If you create a new mail address (see Figure below) you can select the queue or sub queue to be linked with it. This link enables the system to sort incoming messages via the address in the To: field of the mail into the right queue.

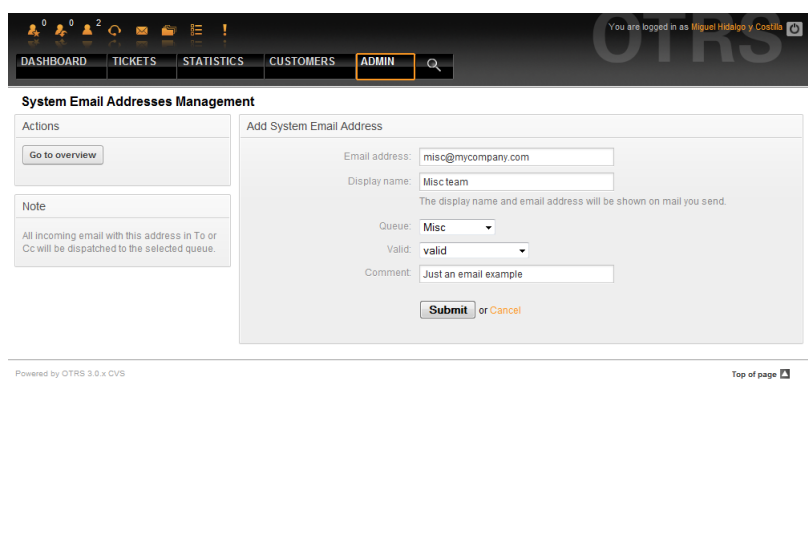


Figure: Adding a system Email Address.

Примечание

Как и другие сущности OTRS, емейл адреса не могут быть удалены, только деактивированы, установив опцию Действительный в значение не действительный или не действительный-временно.

8. Уведомления

OTRS позволяет отправлять уведомления агентам и клиентам при наступлении в системе определенных событий. Агенты могут выбрать системные события для своих собственных уведомлений перейдя по ссылке предпочтения

Through the "Agent Notifications" link on the Admin page, you can manage the notifications of your system (see Figure below). You can use filters to list only certain notifications.

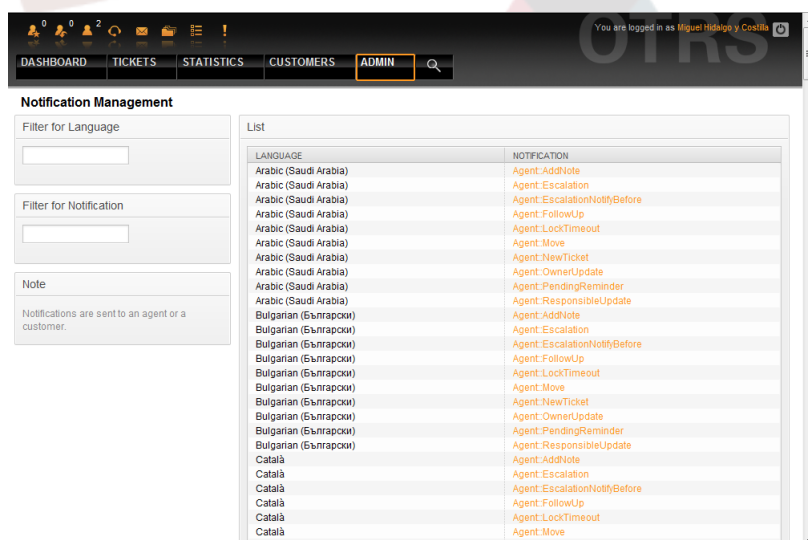


Figure: Notification management.

You can customize the subject and the text of the notifications. Click on the notification you want to change from the list, and its content will get loaded for editing (see Figure). Please note that there is a notification with the same name for each of the available languages.

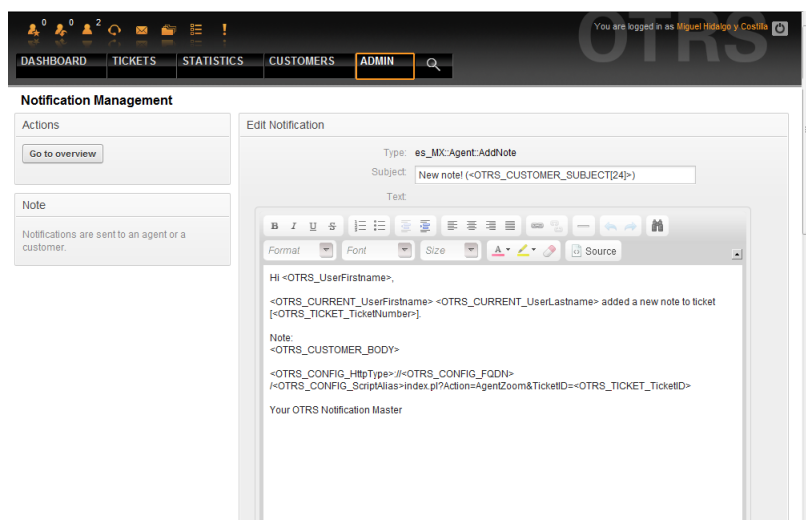


Figure: Customizing a Notification.

Just as with signatures and salutations, it is possible to dynamically create the content of a notification, by using special variables. You can find a list of variables at the bottom of the screen shown in the Figure.

Также есть возможность создавать уведомления основанные на системных событиях. Можно подробно указать когда и кому такое уведомление будет отправлено. Выбрать можно из большого количества параметров, таких как: группа(ы) получателя, агент(ы), роль(и), адресс(а) электронной почты, тип события, который генерирует уведомление, тип заявки, состояние, приоритет, очередь, блокировка, сервис, SLA и т.д.

In order to see a list of all event based notifications, click on the link "Notifications (Event)" on the Admin page (see Figure).

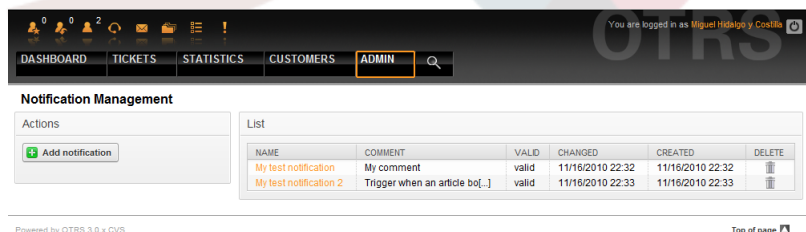


Figure: Event based Notification management.

As shown in Figure, you can create a new notification by clicking on the Add button (see Figure).

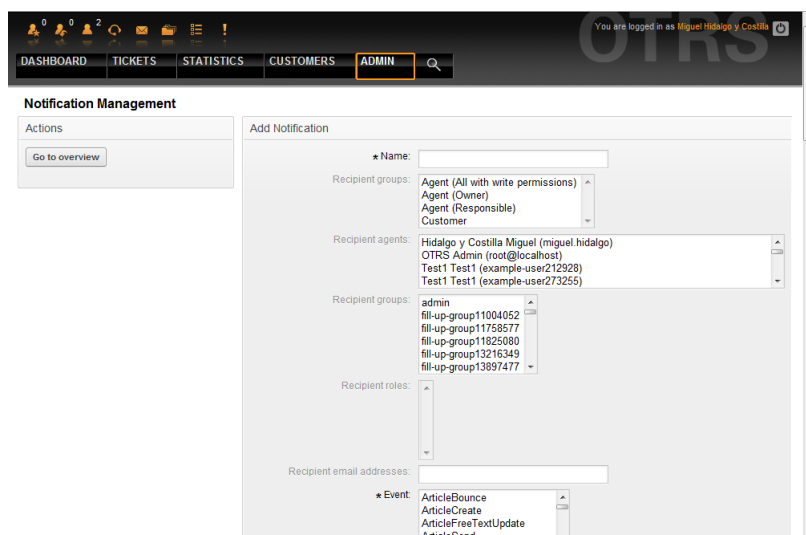


Figure: Registering an Event based Notification management.

Please note that the content of the event based notifications can also be dynamically created by using the special variables listed at the bottom of the screen shown in the Figure.

9. SMIME

OTRS может обрабатывать входящие S/MIME-кодированные сообщения и подписывать исходящие сообщения электронной почты. Перед тем как использовать эту возможность, ее необходимо активировать и настроить путем изменения некоторых параметров конфигурации в файле SysConfig.

The "S/MIME Certificates" link on the Admin page allows you to manage your S/MIME certificates (see Figure below). You can add or remove certificates, and also search through the SMIME data.

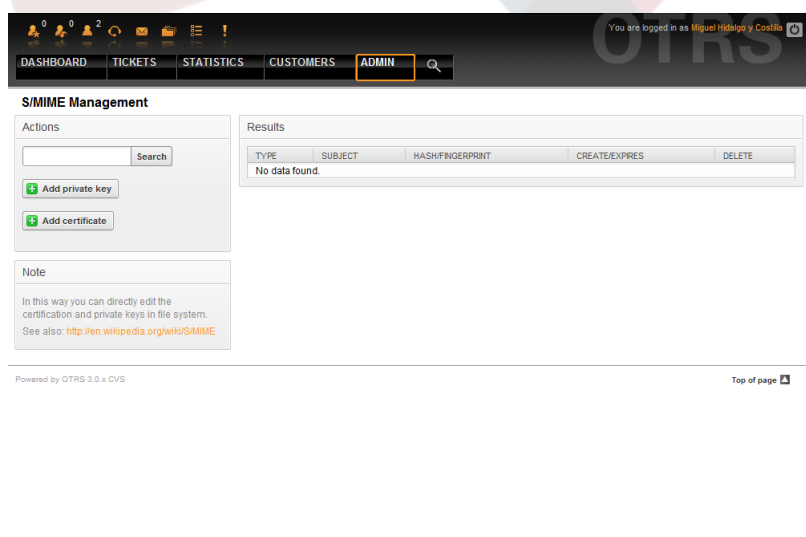


Figure: S/MIME management.

10. PGP

OTRS обрабатывает PGP-ключи, которые позволяют кодировать/декодировать сообщения и подписывать исходящие сообщения. Перед использованием этой функ-

ции ее необходимо активировать и изменить соответствующие конфигурационные параметры в файле SysConfig.

Through the "PGP Keys" link on the Admin page, it is possible to manage the key ring of the user who shall be used for PGP with OTRS (see Figure below), e.g. the local OTRS user or the web server user. It is possible to add and remove keys and signatures, and you can search through all data in your key ring.

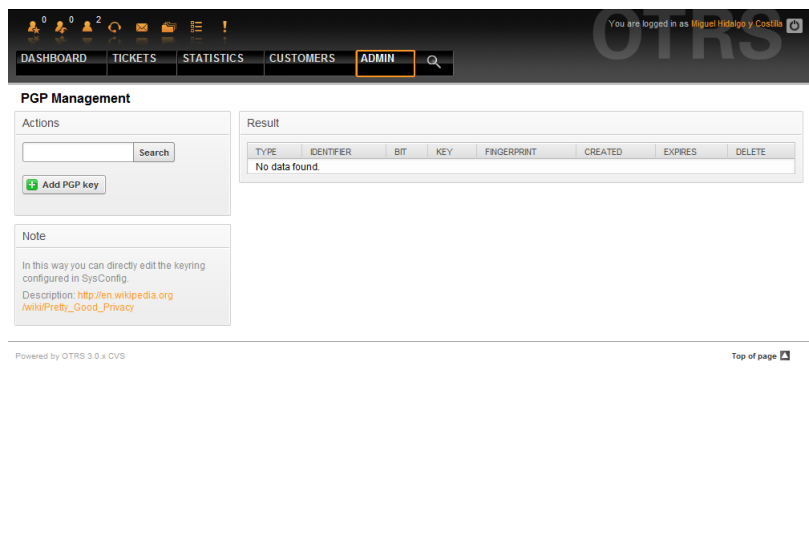


Figure: PGP management.

11. Состояния

Through the "States" link on the Admin page, you can manage the different ticket states you want to use in the system (see Figure below).

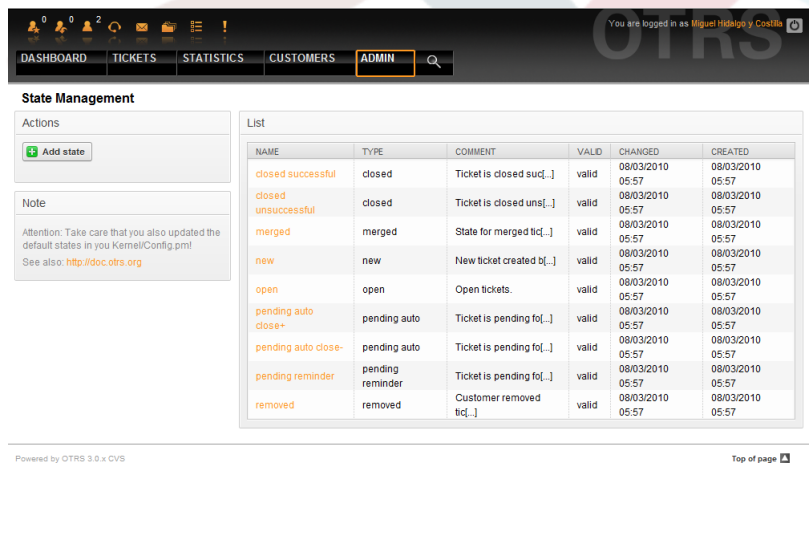


Figure: State management.

После установки в системе будут следующие предустановленные состояния:

- закрыта успешно
- закрыта неуспешно
- объединенные

- новая
- открытая
- ожидание авто закрытие+
- ожидание авто закрытие-
- ожидание напоминание
- удаленная

Каждое состояние связано с типом, который нужно указать когда создается новое состояние. По умолчанию есть такие типы состояний:

- закрытая
- объединенные
- новая
- открытая
- ожидание авто
- ожидание напоминание
- удаленная

12. SysConfig

Ссылка SysConfig - указывает на место, где хранится большинство конфигурационные параметров OTRS.

The SysConfig link on the Admin page loads the graphical interface for system configuration (see Figure below). You can upload your own configuration files for the system, as well as backup all your current settings into a file. Almost all configuration parameters of the OTRS framework and installed applications can be viewed and changed through this interface. Since all configuration parameters are sorted into groups and sub groups, it is possible to navigate quickly through the multitude of the parameters. It is also possible to perform a full-text search through all the configuration parameters.

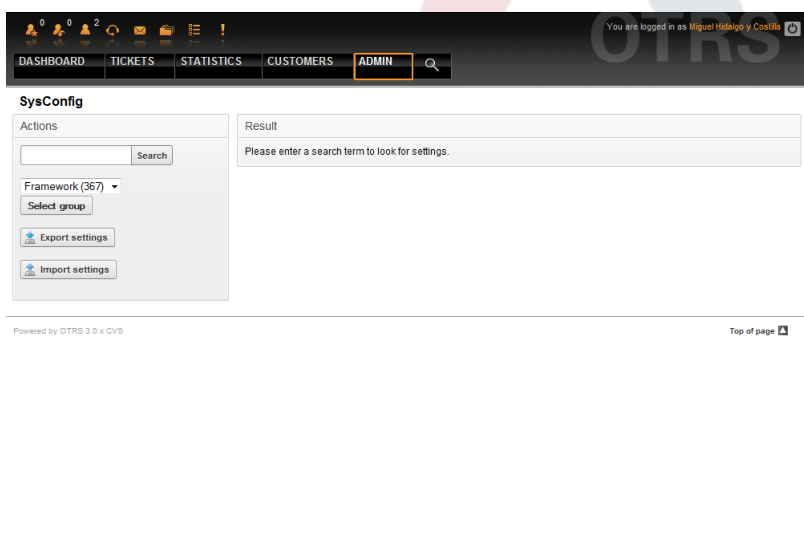


Figure: The graphical interface for system configuration (SysConfig).

Более подробно графический интерфейс для настройки системы рассматривается в главе "Настройка системы через веб-интерфейс" .

13. Использование почтовый ящиков

There are several possibilities to transport new emails into the ticket system. One way is to use a local MTA and the `otrs.PostMaster.pl` script that pipes the mails directly into the system. Another possibility is the use of mail accounts which can be administrated through the web interface. The "PostMaster Mail Accounts" link on the Admin page loads the management console for the mail accounts (see Figure below). OTRS supports the mail protocols: POP3, POP3S, IMAP and IMAPS.

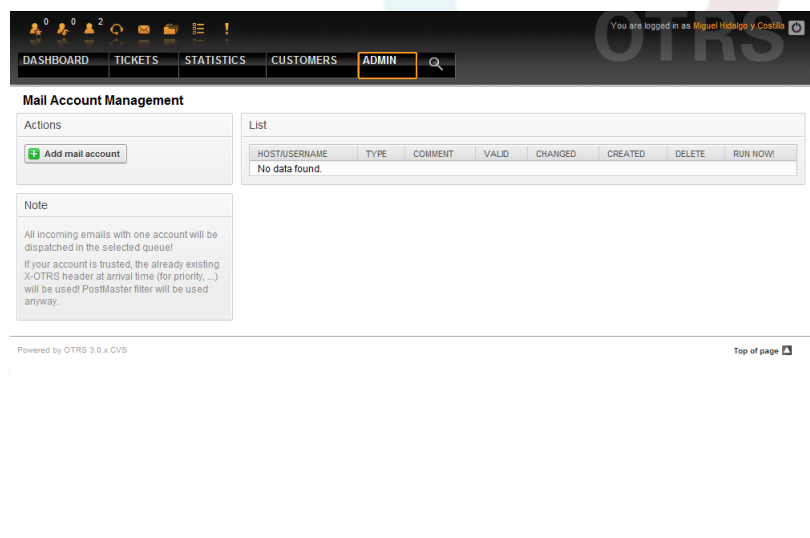


Figure: Mail account management.

See the section about PostMaster Mail Accounts for more details.

14. Filtering incoming email messages

OTRS has the capability to filter incoming email messages. As an example, it is possible to put certain emails automatically into specified queues, or set a specific state or ticket type for some mails. The filters apply to all incoming mail. You can maintain your filters via the link "PostMaster Filter" on the Admin page (see Figure below).

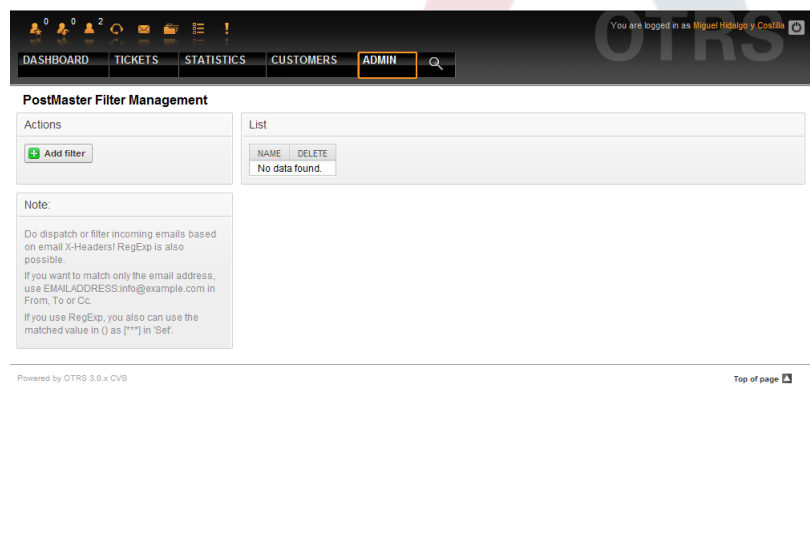


Figure: PostMaster filter management.

A filter consists of one or more criteria that must match for the defined actions to be executed on the email. Filter criteria may be defined for the headers or the body of an email, e.g. search for specific header entries, such as a sender address, or on strings in the body. Even regular expressions can be used for extended pattern matching. If your filter matches, you can set fields using the X-OTRS headers in the GUI. These values will be applied when creating the ticket or follow-up message in OTRS. The Table 5-4 lists the different X-OTRS headers and their meaning.

Note: You also can use X-OTRS-FollowUp-* headers to set values for follow up emails.

Таблица 5.4. Функции различных X-OTRS-заголовков

Название	Возможные значения:	Описание
X-OTRS-Priority:	1 очень низкий, 2 низкий, 3 нормальный, 4 высокий, 5 очень высокой	Устанавливает приоритет заявки.
X-OTRS-Queue:	Название очереди в системе.	Sets the queue where the ticket shall be sorted. If set in X-OTRS header, all other filter rules that try to sort a ticket into a specific queue are ignored. If you use a sub-queue, specify it as "Parent::Sub".
X-OTRS-Lock:	блокировать, разблокировать	Устанавливает состояние блокирования заявки.
X-OTRS-Ignore:	Yes или True	Если этот X-OTRS заголовок установлен в "Yes", входящие сообщения будут полностью игнорироваться и никогда не попадут в систему.
X-OTRS-State:	новая, открытая, закрытая успешно, закрытая неуспешно, ...	Устанавливает следующее состояние заявки.
X-OTRS-State-PendingTime:	например 2010-11-20 00:00:00	Устанавливает время ожидания для заявки (вы также можете отправить состояние ожидания с помощью X-OTRS-State).
X-OTRS-Type:	по умолчанию (в зависимости от настроек)	Sets the type of a ticket (if Ticket::Type is activated).
X-OTRS-Service:	(в зависимости от настроек)	Sets the service of a ticket (if Ticket::Service is active). If you want to set a sub-service you should specify it as "Parent::Sub".
X-OTRS-SLA:	(в зависимости от настроек)	Устанавливает SLA для заявки (если активирована поддержка Ticket::Service).
X-OTRS-CustomerUser:	CustomerUser	Устанавливает клиентского пользователя для заявки.

Название	Возможные значения:	Описание
X-OTRS-CustomerNo:	CustomerNo	Устанавливает ID пользователя для заявки.
X-OTRS-SenderType:	агент, система, клиент	Устанавливает тип отправителя заявки.
X-OTRS-ArticleType:	email-external, email-internal, email-notification-ext, email-notification-int, phone, fax, sms, webrequest, note-internal, note-external, note-report	Устанавливает тип статьи для входящих заявок.
X-OTRS-DynamicField-<DynamicFieldName>:	Depends on Dynamic Field configuration (Text: Notebook, Date: 2010-11-20 00:00:00, Integer: 1)	Saves an additional info value for the ticket on <DynamicFieldName> Dynamic Field.
X-OTRS-Loop:	True	Если этот X-OTRS заголовок установлен, то отправителю сообщение не доставляется ни одного автоматического ответа (защита от зацыкливания).

You should specify a name for every filter rule. Filter criteria can be specified in the section "Filter Condition". Choose via the listboxes for "Header 1", "Header 2" and so on for the parts of the messages where you would like to search, and specify on the right side the values you wish to filter on. In the section "Set Email Headers", you can choose the actions that are triggered if the filter rules match. You can select for "Header 1", "Header 2" and so on to select the X-OTRS-Header and set the associated values (see Figure below).

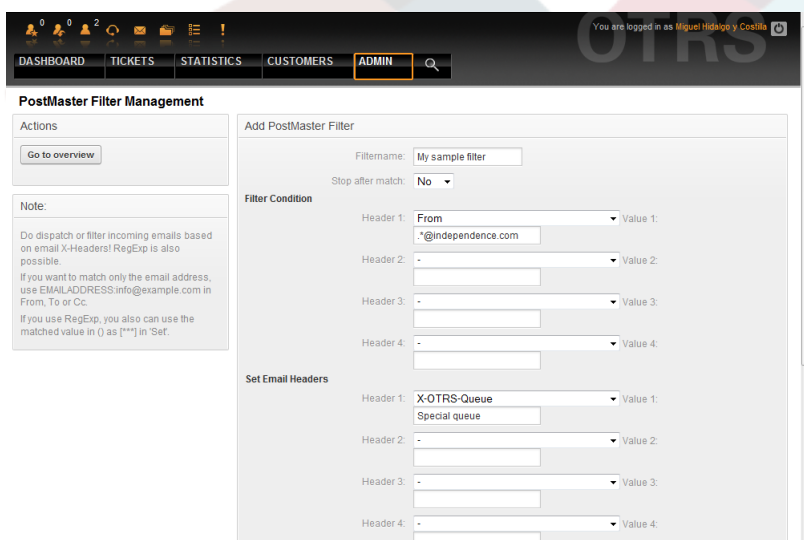


Figure: Add a PostMaster filter.

Пример 5.1. Сортирование спама в определенную очередь

Полезные правила фильтрации помогают OTRS автоматически сортировать почту помеченную как спам с инструментом обнаружения спама, таких как SpamAssassin, в "Нежелательная" очередь. SpamAssassin добавляет "X-Spam-Flag" заголовок для каждого проверить почту. Если письмо помечено как спам, заголовков установлена в "да". Так критерии фильтра будет "X-Spam-Flag: Yes". Чтобы создать правило

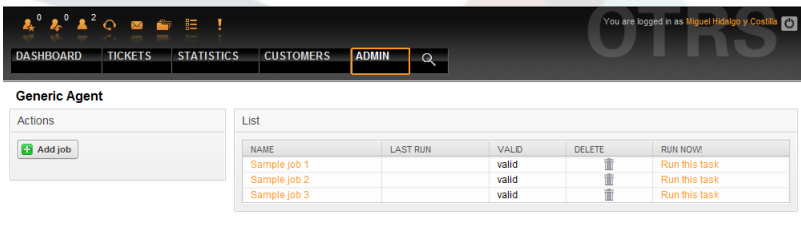
фильтрации с этим критериям можно вставить имя, например "спам-почты". Также выбрать в разделе "Матч" для "Заголовок 1" из списка "X-Spam-Flag:". Вставить "Да" в качестве значения для этого заголовка. Теперь критерии фильтра указан. Чтобы убедиться, что все спам-письма встанет в "Нежелательная" очереди выбрать в разделе "Установить" для "Заголовок 1" "X-OTRS-очереди:" запись. Укажите "Нежелательная" в качестве значения для этого заголовка. Наконец добавить новые правила фильтрации, чтобы активировать его на следующий новых сообщениях в системе.

Есть дополнительные модули, которые можно использовать для более подробной фильтрации входящих сообщений. Эти модули будут полезны для больших и сложных систем.

15. Выполнение автоматизированных заданий с помощью GenericAgent

GenericAgent является инструментом для автоматического выполнения задач. Если бы его небыло, то эти задачи пришлось бы выполнять живому человеку - агенту. GenericAgent, например, может закрывать или перемещать заявки, отправлять уведомления для эскалированных заявок и т.д.

Click the link "GenericAgent" on the Admin page (see Figure below). A table with currently automated jobs is displayed which can be edited to switch to executing jobs manually, or removing them.



NAME	LAST RUN	VALID	DELETE	RUN NOW
Sample job 1		valid		Run this task
Sample job 2		valid		Run this task
Sample job 3		valid		Run this task

Figure: Job list for the GenericAgent.

Click the "Add job" button to create a new job. You first need to supply a name for the job, as also the times when the job should be executed. Different criteria to select the tickets to work on and the new properties of those tickets can also be set (see Figure below).

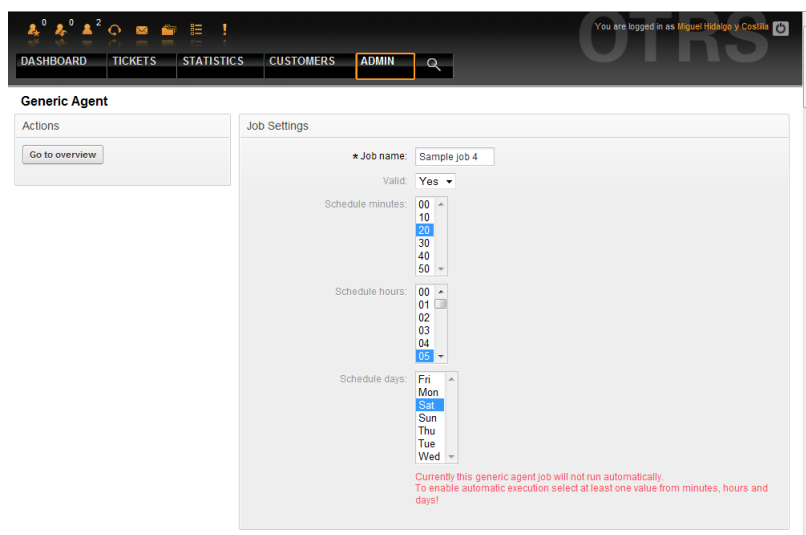


Figure: Creating a job for the GenericAgent.

По завершению создания заданий, будут отображены все заявки, затронутые этим заданием. Этот список поможет проверить правильность выполнения задания. На этом этапе не происходит никаких изменений в заявках. Задание будет активировано только после его сохранения в списке заданий.

16. Электронная почта администратора

OTRS administrators can send messages to specific users or groups. The "Admin Notification" link on the Admin page opens the screen where the agents and groups that should be notified can be selected (see Figure below).

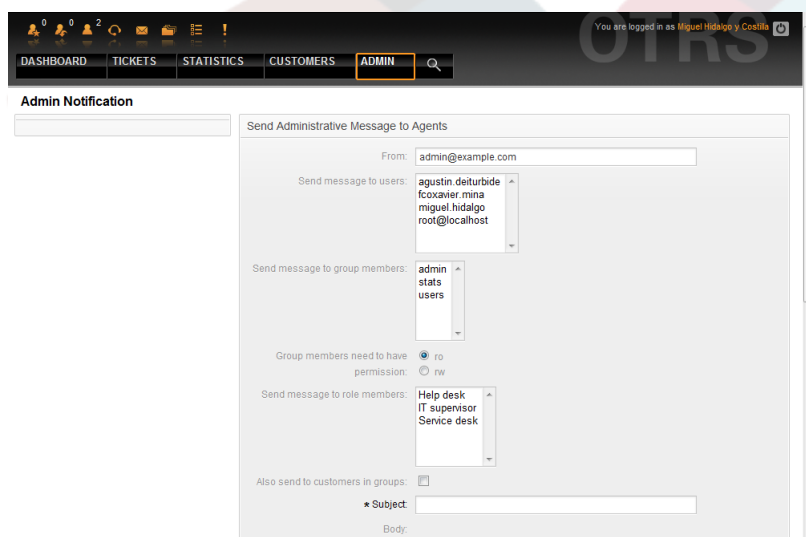
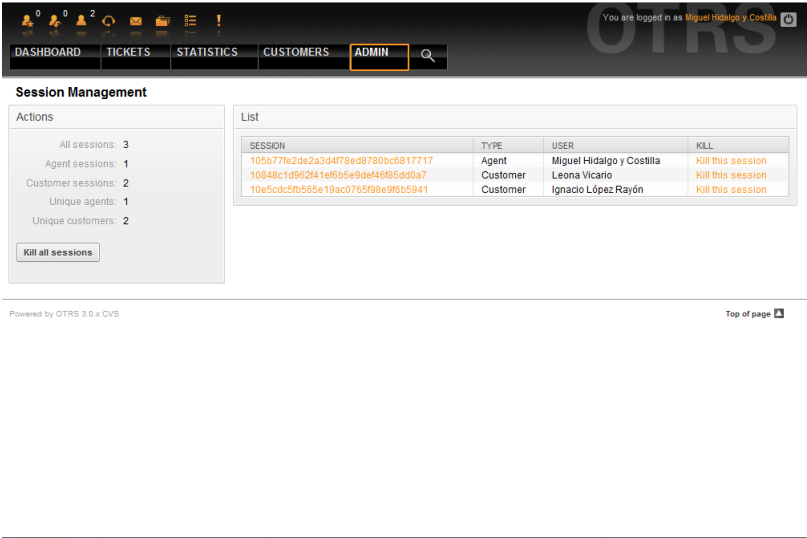


Figure: Admin notification.

Можно указать отправителя, тему и текст уведомления. Также можно выбрать агентов, группы и роли, которые должны получить сообщение.

17. Управление сеансами

You can see all logged in users and their session details by clicking the "Session Management" link in the admin area (see Figure below).



Session Management

Actions

- All sessions: 3
- Agent sessions: 1
- Customer sessions: 2
- Unique agents: 1
- Unique customers: 2

[Kill all sessions](#)

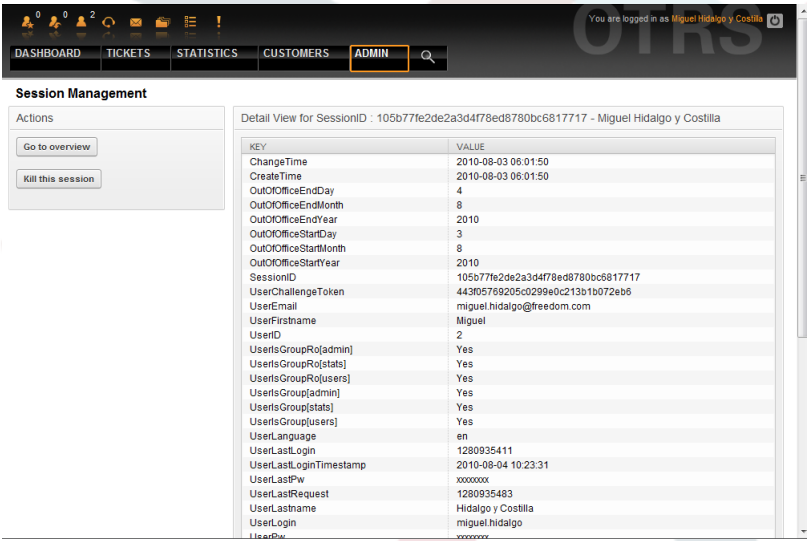
List

SESSION	TYPE	USER	KILL
105b77fe2de2a3d479ed8780bc6817717	Agent	Miguel Hidalgo y Costilla	Kill this session
10848c1d962f41ef8b5e9de4895d0a7	Customer	Leona Vicaria	Kill this session
10e5cdc5fb555e19ac076598e9f6b5941	Customer	Ignacio López Rayón	Kill this session

Powered by OTRS 3.0 x CVS Top of page

Figure: Session management.

Some statistics about all active sessions are displayed, e.g. how many agents and customer users are logged in, number of active sessions. Any individual session can be removed by clicking on the *Kill this session* link on the right-hand side of the list. You also have the option to *Kill all sessions*, which can be useful if you'd like to bring the system down. Detailed information for every session is available, too (see Figure below).



Session Management

Actions

- [Go to overview](#)
- [Kill this session](#)

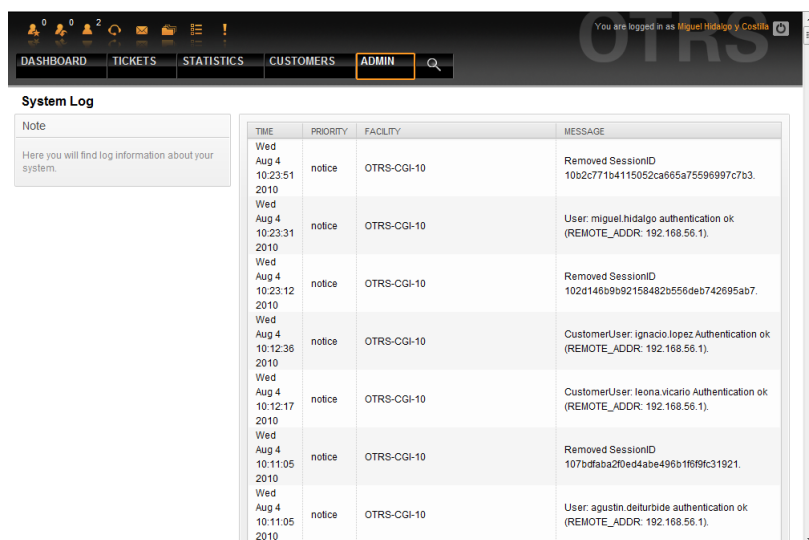
Detail View for SessionID : 105b77fe2de2a3d479ed8780bc6817717 - Miguel Hidalgo y Costilla

KEY	VALUE
ChangeTime	2010-08-03 06:01:50
CreateTime	2010-08-03 06:01:50
OutOfficeEndDay	4
OutOfficeEndMonth	8
OutOfficeEndYear	2010
OutOfficeStartDay	3
OutOfficeStartMonth	8
OutOfficeStartYear	2010
SessionID	105b77fe2de2a3d479ed8780bc6817717
UserChallengeToken	4430579205c0299e0c213b1b072eb6
UserEmail	miguel.hidalgo@freedom.com
UserFirstname	Miguel
UserID	2
UsersGroupRo(admin)	Yes
UsersGroupRo(stats)	Yes
UsersGroupRo(users)	Yes
UsersGroup(admin)	Yes
UsersGroup(stats)	Yes
UsersGroup(users)	Yes
UserLanguage	en
UserLastLogin	1280935411
UserLastLoginTimestamp	2010-08-04 10:23:31
UserLastPw	xxxxxxx
UserLastRequest	1280935483
UserLastname	Hidalgo y Costilla
UserLogin	miguel.hidalgo
UserPw	xxxxxxx

Figure: Session details.

18. Системный журнал

The "System Log" link on the Admin page shows the log entries of the system, reverse chronologically sorted with most recent first (see Figure below).



The screenshot shows the OTRS Admin interface with the 'System Log' tab selected. The log table contains the following entries:

TIME	PRIORITY	FACILITY	MESSAGE
Wed Aug 4 10:23:51 2010	notice	OTRS-CGI-10	Removed SessionID 10b2c771b4115052ca665a75596997c7b3.
Wed Aug 4 10:23:31 2010	notice	OTRS-CGI-10	User: miguel.hidalgo authentication ok (REMOTE_ADDR: 192.168.56.1).
Wed Aug 4 10:23:12 2010	notice	OTRS-CGI-10	Removed SessionID 102d146b9b92158482b556deb742695ab7.
Wed Aug 4 10:12:36 2010	notice	OTRS-CGI-10	CustomerUser: ignacio.lopez Authentication ok (REMOTE_ADDR: 192.168.56.1).
Wed Aug 4 10:12:17 2010	notice	OTRS-CGI-10	CustomerUser: leona.vicario Authentication ok (REMOTE_ADDR: 192.168.56.1).
Wed Aug 4 10:11:05 2010	notice	OTRS-CGI-10	Removed SessionID 107bdfaba2f0ed4abe496b1f8f9fc31921.
Wed Aug 4 10:11:05 2010	notice	OTRS-CGI-10	User: agustn.delrubi authentication ok (REMOTE_ADDR: 192.168.56.1).

Figure: System Log.

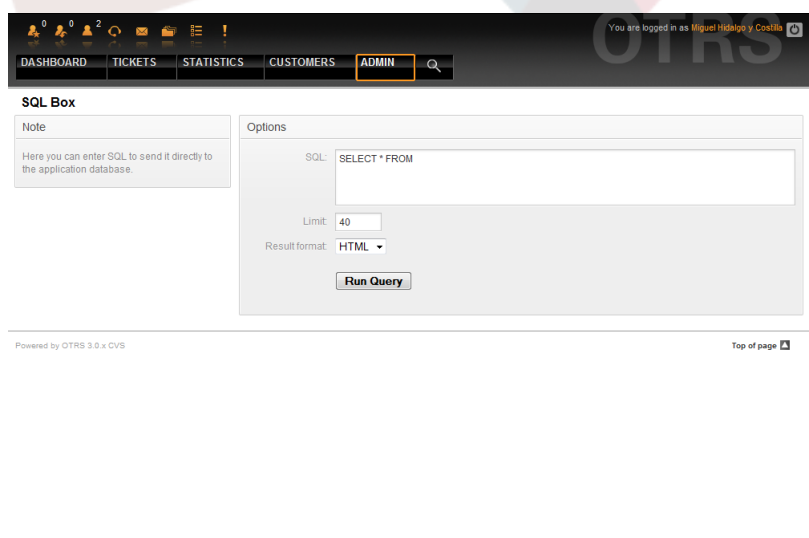
Каждая строка в журнале содержит: время, приоритет, компонент системы и непосредственно саму запись.

Примечание

Системный журнал доступен только через веб-интерфейс на Linux/UNIX системах. В Windows-системах, журнал можно увидеть, открыв в текстовом редакторе файл `[install_dir]otrs\var\log\otrs.log`.

19. SQL запросы

The "SQL Box" link on the Admin page opens a screen that lets you query the content of the tables in the OTRS database (see Figure below). It is not possible to change the content of the tables, only select queries are allowed.



The screenshot shows the OTRS Admin interface with the 'SQL Box' tab selected. The interface includes a text area for entering SQL queries, a 'Limit' field set to 40, and a 'Result format' dropdown menu set to 'HTML'. A 'Run Query' button is located below the form.

Figure: SQL Box.

20. Менеджер пакетов

Using the "Package Manager" link on the Admin page, you can install and manage packages that extend the functionality of OTRS (see Figure below). See the Additional

applications section for a discussion on the extensions that are available from the OTRS repositories.

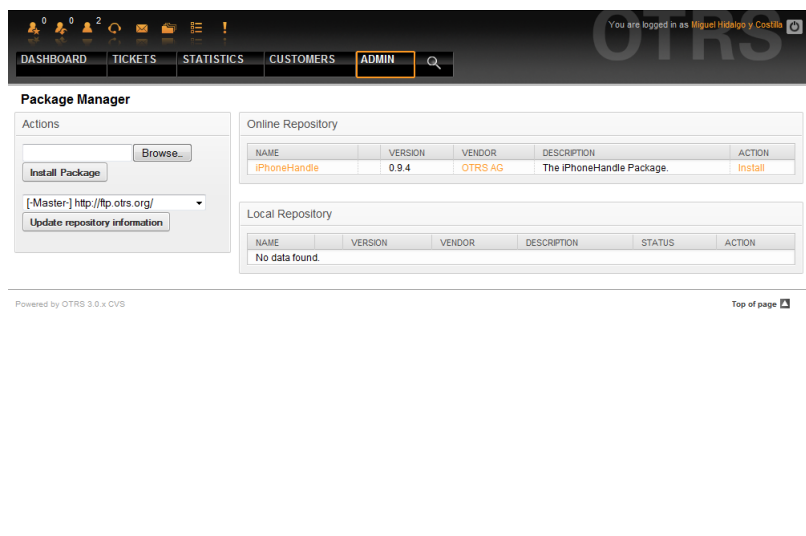


Figure: Package Manager.

Менеджер Пакетов позволяет просмотреть пакеты расширения OTRS вместе с номерами их версий, установленные на сервере в данный момент.

Пакеты можно установить из удаленного хоста, для этого, в разделе *Онлайн Репозиторий* нужно выбрать репозиторий и нажать кнопку *Обновить информацию из репозитория*. Все доступные пакеты будут отображены в соответствующей таблице. В правой части страницы отображаются доступные пакеты. Чтобы установить пакет, нажмите на ссылку *Установить*. После установки пакет будет отображаться в разделе *Local Repository*.

Для обновления установленных пакетов можно воспользоваться списком доступных пакетов из он-лайн репозитории. В колонке "действие" для каждого из пакетов, который имеет более высокую версию (чем установленная сейчас) нажмите кнопку *Обновить*. Просто нажмите на кнопку "Обновить", и новый пакет будет обновлен.

В некоторых случаях, например, когда ваша OTRS-система не подключена к Интернету, можно установить пакеты, которые были заранее загружены на локальный диск. Нажмите кнопку *Обзор* на боковой панели и выберите .otm-файл на жестком диске. Нажмите *Открыть*, а затем *Установить пакет*. После установки пакет отображается в разделе *Локальный Репозиторий*. Прделайте те же действия для того чтобы обновить уже установленный пакет.

В особых случаях, можно настроить Менеджер Пакетов, например, для использования прокси-сервера, или локального репозитория. перейдя по ссылке `Framework:Core::Package`, просто посмотрите на имеющиеся варианты в файле `SysConfig`.

21. Web Services

The Web Services link leads to the graphical interface where web services (for the OTRS Generic Interface) are created and maintained (see Figure below).

Figure: The graphical interface for web services.

The graphical interface for web services configuration is described in more detail in the section "Web Service Graphical Interface".

22. Dynamic Fields

Dynamic Fields is the place where you setup an manage custom fields for tickets and articles (see figure below).

Figure: The dynamic fields overview screen with some dynamic fields.

The dynamic fields configuration is described in more detail in the section "Dynamic Fields Configuration".

Each dynamic field type has its own configuration settings and therefore its own configuration screen.

Примечание

In the OTRS framework, dynamic fields can only be linked to tickets and articles by default, but they can be extended to other objects as well.



Глава 6. Настройка Системы

1. Конфигурационные файлы OTRS

Все конфигурационные файлы OTRS хранятся в директории Kernel и ее поддиректориях. Нет необходимости изменять другие файлы кроме Kernel/Config.pm, потому что остальные файлы будут изменены, после обновления системы. Просто скопируйте конфигурационные параметры из других файлов в Kernel/Config.pm и измените их в соответствии с вашими потребностями. Этот файл никогда не будет затронут во время обновления, так что настройки, которые вы произвели вручную будут сохранены.

Файл Kernel/Config/Defaults.pm содержит параметры конфигурации главного фреймворка OTRS. Он определяет все основные параметры системы, конфигурация электронной почты, подключения к базе данных, кодировка по умолчанию, язык интерфейса. Файл Kernel/Config/Files/Ticket.pm содержит все конфигурационные параметры системы обработки заявок.

В каталоге Kernel/Config/Files есть некоторые другие файлы, которые передаются, когда запускается страница входа в OTRS. Если установлены дополнительные приложения, такие как FAQ (ЧАВО) или Менеджер Файлов, то их конфигурационные файлы также могут быть найдены по указанному пути.

Если веб-интерфейс OTRS доступен - все .xml-файлы из директории Kernel/Config/Files анализируются в алфавитном порядке, затем подгружаются настройки главного фреймворка и дополнительных приложений. В последствии, будут вычислены настройки в двух файлах Kernel/Config/Files/ZZZAAuto.pm и Kernel/Config/Files/ZZZAuto.pm. Оба файла используются графическим веб-интерфейсом конфигурации системы и никогда не должны редактироваться вручную. Последним оценивается файл Kernel/Config.pm, который содержит индивидуальные настройки и параметры конфигурации изменяемые вручную. Чтение конфигурационных файлов в таком порядке гарантирует, что специфические параметры вашей конфигурации будут использованы системой.

2. Настройка системы через графический веб-интерфейс

Since OTRS 2.0, nearly all configuration parameters of the central framework or additional installed applications, can be changed easily with the graphical interface for system configuration. Log in as OTRS administrator and follow the SysConfig link on the Admin page to execute the new configuration tool (see Figure below).

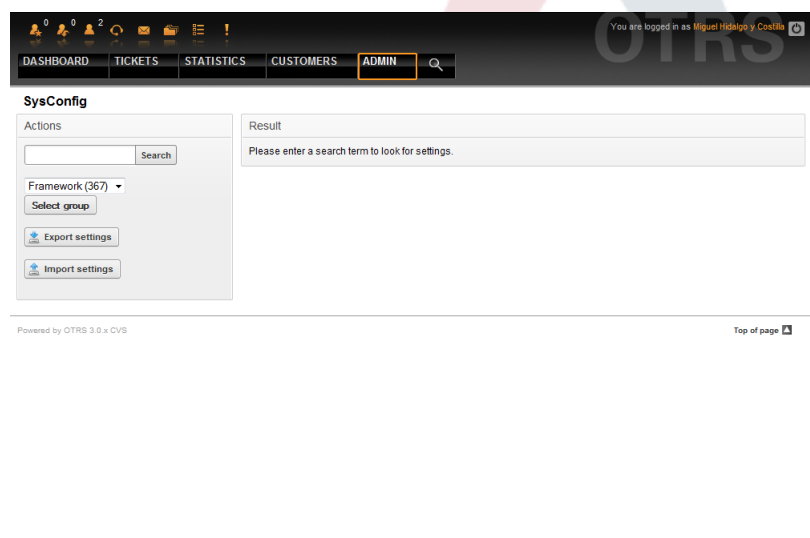


Figure: The graphical interface for system configuration.

Поскольку в настоящее время в OTRS имеется больше 600 параметров конфигурации, есть несколько различных способов быстрого доступа к специфическим настройкам. Используя полнотекстовый поиск можно просмотреть все конфигурационные параметры на предмет совпадения с определенным ключевым словом. Полнотекстовый поиск ищет совпадения не только в названиях конфигурационных параметров, но также и в их описаниях. Это позволяет найти тот или иной элемент даже в том случае если его название неизвестно.

Кроме этого, все параметры конфигурации отсортированы по основным группам и подгруппам. Главная группа представляет собой приложение состоящие из подгрупп "Framework" для главного OTRS-фреймворка, "Ticket" для системы обработки заявок, "FAQ" - для системы ответов и вопросов и так далее. Подгруппы могут быть доступны если приложение выбирается из списка групп "Выбор групп" кнопка нажата.

Каждый параметр конфигурации может быть "включен" или "выключен" с помощью флажка. Если параметр "выключен" (флажок не выбран) система будет игнорировать этот параметр или использовать настройки по умолчанию. Также, воспользовавшись кнопкой "Восстановить" можно переключится между измененным значением конфигурационного параметра и его значением по умолчанию.

Если нужно сохранить все изменения, внесенные в конфигурацию системы, например, для быстрой настройки новой установки, можно воспользоваться кнопкой "Экспортировать настройки", которая создаст .pm-файл. Чтобы восстановить свои собственные настройки, нажмите кнопку "Импортировать настройки" и выберите .pm-файл с настройками созданными ранее.

Примечание

Из за соображений безопасности, параметры конфигурации соединения с базой данных не могут быть изменены в SysConfig. Их нужно установлены вручную в файле Kernel/Config.pm.

Глава 7. Отправка / получение электронной почты

1. Отправка почты

1.1. через Sendmail (по умолчанию)

OTRS может отправлять сообщения электронной почты через [Sendmail](#), [Postfix](#), [Qmail](#) or [Exim](#)). По умолчанию используется Sendmail, он должен работать при установке системы "из коробки".

Параметры sendmail можно настроить через графический веб-интерфейс для конфигурации (Framework::Core::Sendmail)

1.2. Через SMTP-сервер или smarthost

OTRS может отсылать сообщения электронной почты через SMTP ([Simple Mail Transfer Protocol / RFC 821](#)) или Secure SMTP. Возможно вы захотите использовать эту опцию на не-unix-системах (например на Windows).

Параметры настройки SMTP-сервера могут быть сконфигурированы через SysConfig (Framework::Core::Sendmail). Если у вас нету SMTPS в качестве опции, значит вы пропустили установку соответствующих Perl-модулей. В таком случае, обратитесь пожалуйста к этой "Установка Perl-модулей необходимых для работы OTRS" инструкции.

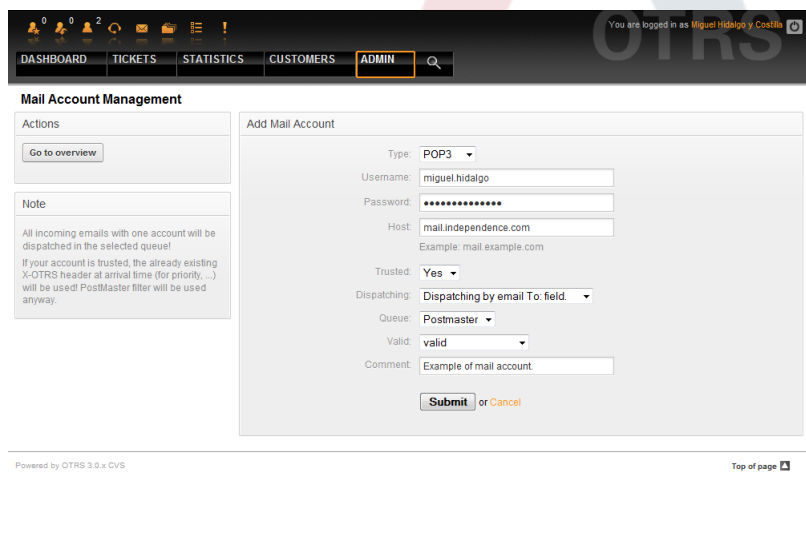
2. Получение сообщений электронной почты

2.1. Учетные записи электронной почты настраиваемые через графический пользовательский интерфейс OTRS

OTRS позволяет получать сообщения электронной почты через почтовые аккаунты POP3, POP3S, IMAP, и IMAPS.

Для настройки почтовых аккаунтов воспользуйтесь ссылкой "Почтовые Аккаунты PostMaster" на странице администрирования.

If a new mail account is to be created (see Figure below), then it's mail server name, login name and password must be specified. Also, you need to select the mail server type, which can be POP3, POP3S, IMAP or IMAPS. If you don't see your server type available as an option, the required Perl modules are missing on your system. In that case, please refer to "Installation of Perl modules required for OTRS" for instructions.



The screenshot shows the OTRS Admin interface for 'Mail Account Management'. The 'Add Mail Account' form is displayed with the following details:

- Type: POP3
- Username: miguel.hidalgo
- Password: [Redacted]
- Host: mail.independencia.com
- Example: mail.example.com
- Trusted: Yes
- Dispatching: Dispatching by email To: field
- Queue: Postmaster
- Valid: valid
- Comment: Example of mail account

Buttons: Submit or Cancel

Figure: Adding a mail account.

Если для опции "Trusted" выбрано значение "Да", будут оцениваться и выполняться любые X-OTRS-заголовки, присоединенные к входящему сообщению. Поскольку X-OTRS-заголовок может выполнять некоторые действия в системе обработки заявок, то для известных отправителей опцию Trusted нужно установить только в значение Да. X-OTRS-заголовки используются в OTRS модулем фильтрации. Более подробно X-OTRS заголовки рассматриваются в этой таблице. Любые созданные и выполненные правила фильтрации независимы от параметров настройки Trusted.

Можно контролировать распределением входящих сообщений, если они должны сортироваться по очередях или содержимому поля "Кому". Если для Диспетчеризации выбрана опция "Диспетчеризация по выбранной очереди", то все входящие сообщения будут отсортированы в указанной очереди. В таком случае адрес, с которого отправили сообщение не учитывается. Если для диспетчеризации выбрана опция "Диспетчеризация емейлов по полю Кому", система для входящих емейлов проверяет, связана ли эта очередь с адресом электронной почты в поле Кому. Перейдя по ссылке Управление E-mail-адресами в Панели Администрирования, можно связать определенный адрес с очередью. Если не найдено никакой связи между адресом в поле "Кому" то сообщение будет сохраняться в системе в "Raw" очереди, которая есть PostmasterDefaultQueue по умолчанию после установки системы.

Все данные для учетных записей электронной почты сохраняются в базе данных OTRS. Скрипт `otrs.PostMasterMailbox.pl`, который находится в директории `bin` вашей системы OTRS, использует настройки в базе данных и получает почту. Вы можете выполнить файл `./bin/otrs.PostMasterMailbox.pl` вручную чтобы проверить что все ваши настройки почты работают правильно.

Для обычной установки, с помощью файла заданий cron job `postmaster_mailbox` выборка емейлов будет выполняться каждые 10 минут. Для получения дополнительной информации об изменении заданий планировщика задач cron jobs, обратитесь к главе "Настройка пралировщика задач cron jobs для OTRS".

Примечание

При получении почты OTRS удаляет почту с POP или IMAP сервера. Нету такой опции, которая бы позволяла хранить копию сообщения на сервере. Если вы все же хотите чтобы такая возможность присутствовала, скорей всего нужно воспользоваться правилами переадресации на почтовый сервер. Оратитесь пожалуйста к документации по вашему почтовому серверу.

2.2. Через командную строку программы и например, procmail (`otrs.PostMaster.pl`)

Если для получения электронной почты в OTRS нету возможности использовать учетные записи, то эту проблему можно решить с помощью программы командной строки `bin/otrs.PostMaster.pl` Она принимает сообщения электронной почты через STDIN и направляет их непосредственно в OTRS. Это значит что емейлы будут доступны в OTRS когда MDA (Mail Delivery Agent - Агент доставки почты) выполняет эту программу.

To test `bin/otrs.PostMaster.pl` without an MDA, execute the command of the following script.

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

Script: *Testing PostMaster without the MDA.*

Если сообщения электронной почты отображаются в QueueView, значит ваши настройки работают.

Procmail - это очень распространенный фильтр электронной почты в среде Linux. Он устанавливается на большинстве систем. Если нет, перейдите по ссылке [procmail homepage](#).

To configure procmail for OTRS (based upon a procmail configured MTA such as sendmail, postfix, exim or qmail), use the `~otrs/.procmailrc.dist` file and copy it to `.procmailrc` and add the lines of the script below.

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

Script: Configuring procmail for OTRS.

Все емейлы отсылаемые локальному OTRS-пользователю будут обрабатываться `bin/otrs.PostMaster.pl` и потом отображаться в QueueView.

2.3. Получение электронной почты по протоколу POP3 или IMAP и обработка для `otrs.PostMaster.pl`

Для того чтобы получить электронную почту с вашего почтового сервера через POP3 или IMAP и сохранить ее на компьютере на котором установлен OTRS, для локального аккаунта или в procmail, перейдите по ссылке [fetchmail](#).

Примечание

Работающий и сконфигурированный SMTP необходим для работы OTRS.

Можно использовать файл `.fetchmailrc.dist` в домашней директории OTRS и скопировать его в `.fetchmailrc`. Изменить его в соответствии с вашими требованиями (см. ниже Пример 7-1).

Пример 7.1. `.fetchmailrc`

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

Не забудьте установить `.fetchmailrc` to 710 ("`chmod 710 .fetchmailrc`")!

Из Листинга 7-1 выше, `.fetchmailrc`, все емейлы будут перенаправлены в локальный OTRS-аккаунт, если выполнена команда **fetchmail -a**. Установите эту команду в планировщике задач `cronjob` если хотите извлекать емейлы постоянно.

2.4. Фильтрация/рассылка модулями OTRS/PostMaster (для более сложной диспетчеризации)

Если вы используете метод `bin/otrs.PostMaster.pl` или `bin/otrs.PostMasterMailbox.pl`, то с модулем фильтрации PostMaster можно вставить или модифицировать X-OTRS заголовок. С помощью X-OTRS-заголовков, система обработки заявок может вызывать некоторые действия для входящих сообщений, сортировать их в определенные очереди, или, например, изменять приоритет или ID-клиента. Более подробную информацию о X-OTRS-заголовках можно найти в главе добавление аккаунтов электронной почты в Панели Администрирования.

Есть некоторые предустановленные модули фильтрации:

Примечание

Название задания (например `$Self->{'PostMaster::PreFilterModule'}->{'JobName'}`) должно быть уникальным!

`Kernel::System::PostMaster::Filter::Match` модуль по умолчанию для проверки совпадения заголовков определенных емейлов (например "От", "Кому", "Тема", ...). Он может устанавливать новые заголовки email (например `X-OTRS-Ignore: да` или `X-OTRS-Queue: spam`) если совпадают правила совпадения. Задания из Примера 7-2 могут быть прописаны в `Kernel/Config.pm`

Пример 7.2. Пример задания для модуля фильтрации `Kernel::System::PostMaster::Filter::Match`

```
# 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` модуль по умолчанию для получения емейлов для внешних команд. Вывод передается в `STDOUT` и если результат истинна, то устанавливается новый заголовок (например `X-OTRS-Ignore: да` или `X-OTRS-Queue: spam`). Пример 7-3 может быть использован в `Kernel/Config.pm`

Пример 7.3. Пример задания для модуля фильтрации `Kernel::System::PostMaster::Filter::CMD`

```
# 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',
    },
};
```

Конечно, также есть возможность разработки своих собственных `PostMaster`-модулей фильтрации.

Глава 8. Функции связанные с временем

1. Настройка периода рабочего времени, праздников и часовых поясов

Некоторые функции в OTRS, например, как эскалация и автоматическая разблокировка заявок зависят от правильной настройки параметра "рабочее время", часовых поясов и праздников. Все эти настройки можно определить перейдя по ссылке интерфейс SysConfig в Framework > Core::Time. Перейдя по ссылке Framework > Core::Time::Calendar1, которая находится под Framework > Core::Time::Calendar9 можно также указать различные, заранее установленные "наборы рабочего времени", праздников и часовых поясов. Перейдя по ссылке настройки очереди можно определить Календари. Для этого также можно использовать SLA-уровни. Например можно создать различные SLA-схемы: 5 x 8 рабочих часов для стандартного SLA, но и вместе с тем создать другой календарь: 7 x 24 - "Золотой SLA"; также можно иметь отдельный календарь для очереди 'Support-USA', с часовыми поясами, различающимися с очередью 'Support-Japan'. В OTRS можно создавать до 99 различных календарей.

1.1. Бизнес Время

Настройте рабочее время для системы в SysConfig Framework > Core::Time::TimeWorkingHours или для определенного календаря в его конфигурации. Проверьте обозначения для пунктов 8, 9, 10 ... 17 соответствующих часам от 8 AM - 6 PM.

Заявки эскалируются, уведомления для эскалированных и ожидающих заявок отправляются и заявки разблокируются только в рабочее время.

1.2. Праздники с фиксированными датами

Праздники, которые с каждым годом не приходятся на фиксированный день, такие как Новый год или четвертое Июля могут быть указаны в Отпускные дни или в соответствующем разделе календаря 1-9.

Заявки не будут эскалироваться, ни разблокироваться в сроки, определенные как "Специально зарезервированные дни".

Примечание

По умолчанию, OTRS работает по *Немецким* выходным.

1.3. TimeVacationDaysOneTime

Праздники, такие как Пасха, которые ежегодно не имеют фиксированной даты могут быть указаны в Отпускные дни (One-Time) .

Заявки не будут эскалироваться и не будут разблокироваться в период времени определенный как TimeVacationDaysOneTime.

Примечание

OTRS поставляется без единого предустановленного One-Time-праздника. Это означает, что во время настройки системы OTRS, нужно самостоятельно добавить праздники, такие как Пасха, День Благодарения.

2. Автоматические Разблокировки

Заблокированные заявки могут быть разблокированы системой автоматически. Эта опция может быть полезной, если, например, агент заблокировал заявки, которые нужно обработать, но не работал с ними по определенной причине, например был

в отпуске. Чтобы убедиться в том, что заблокированные заявки не будут забыты и их обработают другие агенты, автоматическая разблокировка разблокирует заблокированные заявки при истечении определенного времени.

Перейдя по ссылке настройки очереди для каждой заявки можно установить количество времени, при истечении которого она будет разблокирована. Модуль, `bin/otrs.UnlockTickets.pl`, который периодически вызывается как задание планировщика задач `cron job`, выполняет автоматическую разблокировку заявок.

Уведомления о разблокированных заявках отсылаются только тем агентам, которые добавили очередь с разблокированными заявками в "Мои очереди", и активировали в своих настройках функцию уведомлений о разблокированных заявках.

Заявки будут заблокированы в том случае, если выполняются следующие условия:

- Для очереди в которой хранится заявка определяется *время разблокировки* .
- Заявке присвоен статус *заблокирована*.
- Заявке присвоен статус *открытая*.

Если агент добавляет в заявку новую статью - таймер разблокировки будет сброшен. Он может быть любым из следующих типов: *внешний-email*, *телефон*, *факс*, *смс*, или *внешнее примечание*.

Кроме этого, если последняя статья в заявке создана агентом, и клиент добавляет еще одну, либо через веб-интерфейс либо по электронной почте, таймер разблокировки также будет сброшен.

Последнее событие, которое будет сбрасывать таймер разблокировки, когда заявка передается другому агенту.

Глава 9. Ответственность за Заявку & Наблюдение за Заявкой

Начиная с OTRS 2.1 и выше, в дополнение к владельцу заявки можно определить ответственного за нее человека. Кроме того, все мероприятия, связанные с заявкой могут просматриваться не только владельцем но другими людьми. Эти две возможности системы реализованы с помощью функций TicketResponsible и TicketWatcher и также позволяют работать в рамках иерархической структуры команды.

1. Ответственность за Заявку

Функция ответственности за заявку способствует ее полной обработки агентом, который не является владельцем заявки. Таким образом агент, который заблокировал заявку может передать ее другому агенту, который не является владельцем заявки, для того, чтобы второй просто дал ответ на вопрос. После того как запрос был рассмотрен, первый агент может снять ответственность за заявку с второго агента.

С помощью параметров конфигурации Ticket::Responsible, можно активировать функцию ответственности за заявку. Это приведет к появлению на экране трех дополнительных иконок/значков.

Ticket responsibility can be assigned by calling up the ticket content and clicking on the "Responsible" link in the ticket activities menu of a zoomed ticket in the agent interface (see the Figure below).

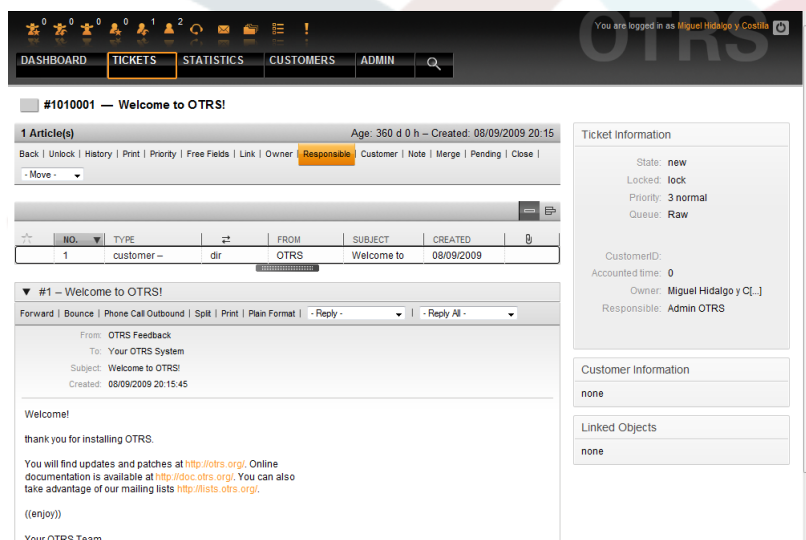


Figure: Changing the Responsibility of a ticket in its zoomed view.

After clicking on "Responsible", a pop-up dialog to change the responsibility of that ticket will open (see Figure below). This dialog can also be used to send a message to the new responsible agent.

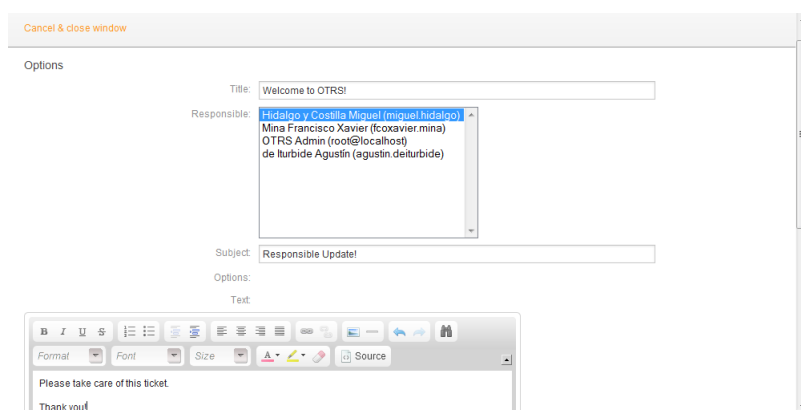


Figure: Pop-up dialog to change a ticket's responsibility.

Если активирована функция ответственности за заявку, то в шаблоне Ответственность, агентского веб-интерфейса OTRS можно просмотреть список всех заявок, за которые агент несет ответственность.

2. Просмотр Заявок

Начиная с OTRS 2.1 и выше с помощью функции TicketWatcher, выбранные агенты, такие как, например, руководители могут просматривать определенные заявки без их обработки.

Функция TicketWatcher может быть активирована с помощью параметра конфигурации Ticket::Watcher, после чего в панели инструментов появятся новые ссылки. Используя Ticket::WatcherGroup можно определить одну или несколько групп пользователей с правами просмотра заявок.

In order to watch a ticket, go to its zoomed view and click on the "Subscribe" link in the ticket activities menu (see Figure below).

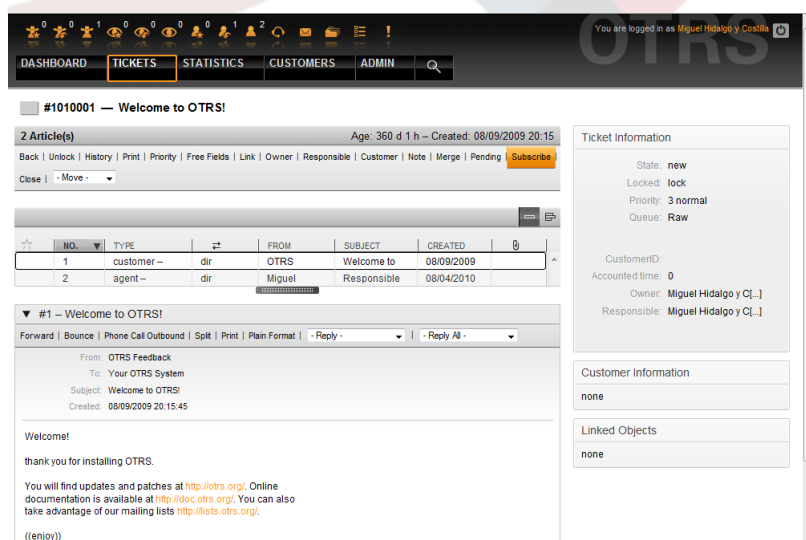
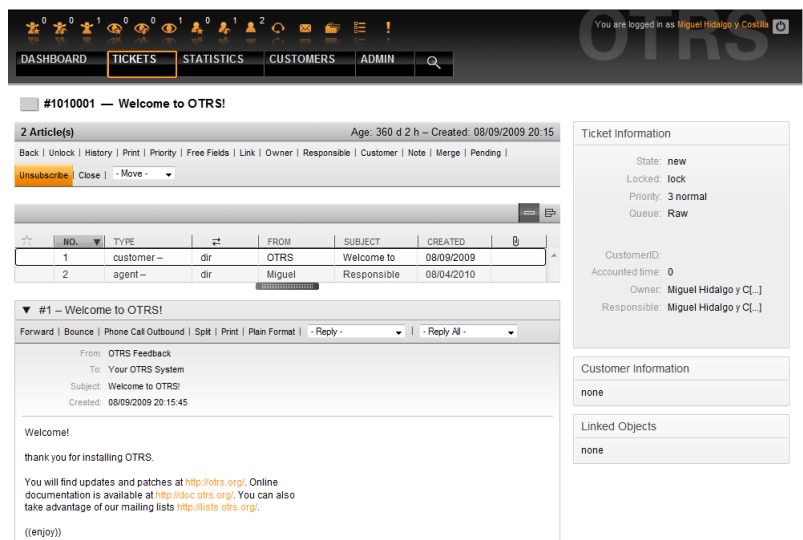


Figure: Subscribing to watching a ticket in its zoomed view.

If you no longer want to watch a specific ticket, go to its zoomed view and click on the "Unsubscribe" link in the ticket activities menu (see Figure below).



DASHBOARD **TICKETS** STATISTICS CUSTOMERS ADMIN

#1010001 — Welcome to OTRS!

2 Article(s) Age: 360 d 2 h — Created: 08/09/2009 20:15

Back | Unlock | History | Print | Priority | Free Fields | Link | Owner | Responsible | Customer | Note | Merge | Pending |

Unsubscribe | Close | - Move -

NO.	TYPE	FROM	SUBJECT	CREATED
1	customer --	dir	OTRS	Welcome to 08/09/2009
2	agent --	dir	Miguel	Responsible 08/04/2010

#1 — Welcome to OTRS!

Forward | Bounce | Phone Call Outbound | Split | Print | Plain Format | - Reply - | - Reply All -


From: OTRS Feedback
 To: Your OTRS System
 Subject: Welcome to OTRS!
 Created: 08/09/2009 20:15:45

Welcome!
 thank you for installing OTRS.
 You will find updates and patches at <http://otrs.org/>. Online documentation is available at <http://doc.otrs.org/>. You can also take advantage of our mailing lists <http://lists.otrs.org/>.
 ((enjoy))

Ticket Information
 State: new
 Locked: lock
 Priority: 3 normal
 Queue: Raw
 CustomerID:
 Accounted time: 0
 Owner: Miguel Hidalgo y C[...]
 Responsible: Miguel Hidalgo y C[...]
Customer Information
 none
Linked Objects
 none

Figure: Unsubscribing from watching a ticket in its zoomed view.

The list of all watched tickets can be accessed through the Watched view of the OTRS agent interface (see Figure below), as soon as the ticket watcher feature gets activated.



DASHBOARD **TICKETS** STATISTICS CUSTOMERS ADMIN

My Watched Tickets: All

All 1 | New Article 0 | Pending 0 | Reminder Reached 0

Bulk 1-1 of 1 - Page: 1 | S M L

TICKET#	AGE	FROM / SUBJECT	STATE	LOCKED	QUEUE	OWNER	CUSTOMERID
1010001	367 d 6 h	OTRS Feedback Welcome to OTRS!	new	lock	Raw	Miguel Hidalgo y Costi[...]	

Powered by OTRS 3.0.x CVS Top of page

Figure: Watched tickets view.

Глава 10. Customizing the PDF output

В этом разделе приводится настройка опций для экспорта данных из OTRS в PDF.

Нажатие кнопки Распечатать из любой точки интерфейса OTRS приведет к генерации PDF-файла. Используя конфигурационный параметр PDF можно изменить такое поведение, заменив его на HTML.

Чтобы изменить внешний вид файлов, сгенерированных системой OTRS, можно создать свой собственный логотип и добавить его перейдя по ссылке PDF::LogoFile. Чтобы определить стандартный размер страниц генерируемых pdf-файлов (DIN-A4 or Letter) воспользуйтесь ссылкой PDF::PageSize. Чтобы указать максимальное количество страниц pdf-файла используйте ссылку PDF::MaxPage. Эта настройка может быть полезна когда пользователь ошибочно генерирует файл большого размера.

Для генерации pdf-файлов должны быть проинсталлированы Perl-модули CPAN: PDF::API2 и Compress::Zlib. В большинстве дистрибутивов они доступны в виде пакетов и могут быть легко установлены с помощью соответствующего менеджера пакетов. В случае, если это не возможно, они должны быть установлены с помощью CPAN. За дополнительной информацией об установке Perl-модулей, обратитесь к главе "Установка Perl-модулей".



Глава 11. Использование внешних хранилищ данных

1. Данные клиентов

OTRS работает с различными атрибутами данных клиентов, такими как имя пользователя, адрес электронной почты, номер телефона и т.д. Эти атрибуты отображаются в обоих фронтэндах, для Агентов и Клиентов. Они также используются для проверки подлинности клиентов.

Данные клиентов, которые используются и отображаются в OTRS очень легко настраиваются. Несмотря на это следующая информация будет всегда необходима для проверки подлинности клиента:

- Вход пользователей
- Адрес электронной почты
- ID Клиента

Use configuration parameters of the following script in your `Kernel/Config.pm` file, if you want to display customer information in your agent interface.

```
# Ticket::Frontend::CustomerInfo*
# (show customer info on Compose (Phone and Email), Zoom and
# Queue view)
$self->{'Ticket::Frontend::CustomerInfoCompose'} = 1;
$self->{'Ticket::Frontend::CustomerInfoZoom'} = 1;
$self->{'Ticket::Frontend::CustomerInfoQueue'} = 0;
```

Script: Kernel/Config.pm configuration parameters.

2. Пользовательский бэк-энд

Вы можете использовать два типа хранилища информации клиентов: Базу Данных DB и LDAP. Если у вас уже есть другой бэк-энд для хранения пользовательской информации (например SAP), также есть возможность написать модуль для использования этой функции.

2.1. База Данных (По умолчанию)

В Примере 11-1 приведена конфигурация базы данных, которая использует данные клиента, хранящиеся в базе данных OTRS.

Пример 11.1. Настройка клиентского хранилища базы данных (DB)

```
# 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',
        # if your frontend is unicode and the charset of your
        # customer database server is iso-8859-1, use these options.
        #     SourceCharset => 'iso-8859-1',
        #     DestCharset => 'utf-8',
    }
}
```

```

    # 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
  [ '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', 'Email', 'email', 1, 1, 'var', '$Env{"CGIHandle"}?
Action=AgentTicketCompose&ResponseID=1&TicketID=$Data{"TicketID"}&ArticleID=
$Data{"ArticleID"}', 0 ],
  [ '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 ],
],
# default selections
Selections => {
  UserTitle => {
    'Mr.' => 'Mr.',
    'Mrs.' => 'Mrs.',
  },
},
},
};

```

If you want to customize the customer data, change the column headers or add new ones to the `customer_user` table in the OTRS database. As an example, the script below shows how to add a new field for room number.

```
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:~#
```

Script: Adding a room field to the customer_user table.

Now add the new column to the MAP array in Kernel/Config.pm, as shown in the following script.

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

Script: Adding a room field to the Kernel/Config.pm file.

Всю эту информацию можно также отредактировать воспользовавшись ссылкой Клиенты.

2.1.1. Клиент с несколькими идентификаторами (Заявки Компании)

Одному клиенту можно назначить больше одного клиентского идентификатора (Customer ID). Это может быть полезно, если клиенту необходимо получить доступ к заявкам других клиентов, например, руководитель хочет посмотреть заявки своих помощников. Если клиент может получить доступ к заявкам другого клиента, то используется особенность OTRS "заявки компании". Заявки компании могут быть доступны перейдя по ссылке "Заявки Компании" в клиентской панели управления.

To use company tickets, a new column with the IDs that should be accessible for a customer, has to be added to the customer_user table in the OTRS database (see Script below).

```
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:~#
```

Script: Adding a customer_ids field to the customer_user table.

Now the new column has to be added to the MAP array in Kernel/Config.pm, as shown in the script below.

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

Script: Adding a UserCustomerIDs field to the Kernel/Config.pm file.

Теперь, новый столбец для мульти-идентификатора (IDs) клиента можно редактировать с помощью веб-интерфейса Агента, в разделе управления клиентами.

Для того, чтобы убедиться что один клиент имеет доступ к заявкам других клиентов нужно добавить идентификаторы (IDs) этих пользователей в новое поле для нескольких идентификаторов клиента. Каждый идентификатор (ID) должен быть отделен точкой с запятой (см. ниже Пример 11-2).

Пример 11.2. Хранение Заявок Компании в базе данных DB

Клиенты А, Б и Ц созданы в вашей системе и А хочет иметь доступ к заявкам Б и Ц используя клиентскую панель. Б и Ц не должны иметь доступа к заявкам других пользователей.

Для реализации этой структуры, измените таблицу customer_user и маппинг (преобразование) в Kernel/Config.pm как это показано выше. С помощью Панели Администрирования или используя ссылку Клиенты в веб-интерфейсе Агента загрузите настройки пользователя А. Если настройки отображаются, добавьте значения "Б;Ц;" в поле для CustomerIDs.

2.2. LDAP

Если у вас есть LDAP--каталог, в котором хранятся данные о клиентах, его можно использовать в OTRS, в качестве хранилища данных о клиентах, как это показано в Примере 11-3.

Пример 11.3. Настройка LDAP в качестве клиентского бэк-енда

```
# 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 both your frontend and your LDAP are unicode, use this:
    SourceCharset => 'utf-8',
    DestCharset => 'utf-8',
    # if your frontend is unicode and the charset of your
    # ldap server is iso-8859-1, use these options.
    # SourceCharset => 'iso-8859-1',
    # DestCharset => 'utf-8',
    # Net::LDAP new params (if needed - for more info see perl doc 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
    [ '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 ],
  ],
};

```

If additional customer attributes are stored in your LDAP directory, such as a manager's name, a mobile phone number, or a department, and if you want to display this information in OTRS, just expand the MAP array in `Kernel/Config.pm` with the entries for these attributes, as shown in the following script.

```

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

```

Script: Adding new fields to the Kernel/Config.pm file.

2.2.1. Клиент с несколькими идентификаторами (Заявки Компании)

При использовании LDAP-бэкэнда клиенту можно присвоить больше одного клиентского айди (Customer ID). Для использования заявок компании, в LDAP-директорию нужно добавить новое поле, которое содержит доступные агенту идентификаторы (IDs).

If the new field in the LDAP directory has been created, the new entry has to be added to the MAP array in `Kernel/Config.pm`, as shown in the script below.

```

# var, frontend, storage, shown (1=always,2=lite), required, storage-type, http-link,
readonly
[...]

```

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

Script: Mapping new fields to the Kernel/Config.pm file.

Клиентские идентификаторы (IDs) можно редактировать напрямую в LDAP-директории. OTRS может только считать информацию из LDAP, но не записывать.

Чтобы убедиться что клиент имеет доступ к заявкам других клиентов, добавьте идентификаторы (IDs) клиентов, к заявкам которых нужен доступ, к новому полю вашей LDAP-директории. Каждый идентификатор ID должен быть отделен точкой с запятой (см. ниже Рисунок 11-4).

Пример 11.4. Использование заявок Компании с LDAP-бэкэндом

Клиенты А, Б и Ц созданы в вашей системе и А хочет иметь доступ к заявкам Б и Ц через панель клиента. Б и Ц не должны иметь доступа к другим пользователям.

Для реализации этой установки измените LDAP-директорию и маппинг (mapping) в Kernel/Config.pm, как это показано выше. Затем добавьте в вашей LDAP-директории значения "Б;Ц;" в поле для CustomerIDs, для клиента "А".

2.3. Использование больше чем одного хранилища информации с OTRS

Если в OTRS нужно использовать больше одного источника данных о клиентах (например LDAP и базу данных), конфигурационный параметр CustomerUser должен быть расширен числом, например "CustomerUser1", "CustomerUser2" (см. ниже Пример 11-5).

Пример 11.5. Использование больше чем одного пользовательского хранилища данных с OTRS

В следующем примере показано применение конфигурации как для LDAP так и для базы данных клиентского бэкэнда с OTRS.

```
# 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 both your frontend and your LDAP are unicode, use this:
        SourceCharset => 'utf-8',
        DestCharset => 'utf-8',
        # if your frontend is e. g. iso-8859-1 and the character set of your
        # ldap server is utf-8, use these options:
        SourceCharset => 'utf-8',
        DestCharset => '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 ],
],
};

```

Есть возможность интегрировать до 10 различных пользовательских бэк-эндов. Используйте интерфейс управления клиентами в OTRS, чтобы просматривать и редактировать данные о них (при условии наличия прав для записи).

3. Хранилища (бэк-енды) для аутентификации Агентов и Клиентов

OTRS предлагает опцию для проверки подлинности агентов и клиентов с использованием различных хранилищ данных (бэкендов).

3.1. Хранилища данных (бэк-енды) для аутентификации Агентов

3.1.1. База Данных (DB, по умолчанию)

В качестве бэк-энда для аутентификации агентов в OTRS, по умолчанию, используется база данных. Чтобы добавлять агентов, редактировать данные о них, перейдите на страницу Панель Администрирования и нажмите ссылку Интерфейс для управления агентами (см. ниже Пример 11.6).

Пример 11.6. Проверка подлинности агентов путем использования Базы Данных (DB) в качестве хранилища информации.

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

3.1.2. LDAP

Если данные всех ваших агентов хранятся в LDAP-директории, то для аутентификации пользователей в OTRS можно использовать LDAP-модуль (см. ниже Пример 11-7). Этот модуль имеет права только на чтение дерева LDAP-каталогов, что озна-

чает что нету возможности редактировать данные пользователей используя ссылку веб-интерфейс для управления пользователями .

Пример 11.7. Проверка подлинности агентов при использовании LDAP в качестве хранилища данных

```
# 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.
# $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,
};
```

The configuration parameters shown in the script below can be used to synchronize the user data from your LDAP directory into your local OTRS database. This reduces the number of requests to your LDAP server and speeds up the authentication with OTRS. The data synchronization is done when the agent authenticates the first time. Although the data can be synchronized into the local OTRS database, the LDAP directory is the last instance for the authentication, so an inactive user in the LDAP tree can't authenticate to OTRS, even when the account data is already stored in the OTRS database. The agent data in the LDAP directory can't be edited via the web interface of OTRS, so the data has to be managed directly in the LDAP tree.

```
# 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',
];
```

Script: Synchronizing the user data from the LDAP directory into the OTRS database.

3.1.3. HTTPBasicAuth-аутентификация для Агентов

Если вы хотите реализовать решение "single sign on" для всех агентов, вы можете использовать базовую аутентификацию (для всех систем) и HTTPBasicAuth-модуль для OTRS (см. ниже Пример 11-8).

Пример 11.8. Аутентификация Агентов с помощью HTTPBasic

```
# 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';
```

3.1.4. Radius

Параметры конфигурации приведенные в Примере 11-9 могут быть использованы для аутентификации агентов с использованием Radius-сервера.

Пример 11.9. Аутентификация (проверка подлинности) агентов с использованием Radius-сервера в качестве хранилища информации

```
# 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';
```

3.2. Хранилища информации для аутентификации (проверки подлинности) клиентов

3.2.1. База Данных (По умолчанию)

Для аутентификации клиентов в OTRS, по умолчанию, используется база данных. Используя базу данных в качестве хранилища, все данные клиентов можно редактировать через веб-интерфейс OTRS (см. ниже Пример 11-10).

Пример 11.10. Аутентификация Клиента в Базе Данных

```
# 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";

```

3.2.2. LDAP

Если у вас есть LDAP-каталог со всеми данными о клиентах, можно использовать модуль LDAP для аутентификации клиентов в OTRS (см. Пример 11-11 ниже). Поскольку этот модуль имеет права только для чтения для данных из LDAP-бэкенда, то нету возможности изменить данные клиента через веб.

Пример 11.11. Аутентификация пользователей с помощью LDAP-бэкенда

```

# 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.
#$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,
};

```

3.2.3. HTTPBasicAuth аутентификация для клиентов

Если для пользователей нужно внедрить "single sign on"-решение, можно использовать базовую, HTTPBasic аутентификацию (для всех систем) и использовать модуль HTTPBasicAuth с OTRS (больше не нужно логинится в OTRS). См. ниже Пример 11-12.

Пример 11.12. Аутентификация клиентов с помощью HTTPBasic

```

# 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';
```

3.2.4. Radius

Настройки приведенные в примере 11-13 могут быть использованы для аутентификации ваших клиентов с помощью Radius-сервера.

Пример 11.13. Аутентификация клиентов с использованием Radius

```
# 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';
```

4. Customizing the customer self-registration

Есть возможность настроить самостоятельную регистрацию новых клиентов, доступную через панель customer.pl. Можно добавить новые дополнительные или обязательные поля, такие как номер комнаты, адрес или состояние.

В следующем примере показано каким образом можно указать обязательные для заполнения поля в базе данных клиентов, в данном случае, для хранения номера клиента.

4.1. Настройка веб-интерфейса

To display the new field for the room number in the customer.pl web interface, the .dtl file responsible for the layout in this interface has to be modified. Edit the Kernel/Output/HTML/Standard/CustomerLogin.dtl file, adding the new field around line 80 (see Script below).

```
[...]
<div class="NewLine">
  <label for="Room">$Text{"Room{CustomerUser}"}</label>
  <input title="$Text{"Room Number"}" name="Room" type="text" id="UserRoom"
  maxlength="50" />
</div>
[...]
```

Script: Displaying a new field in the web interface.

4.2. Отображения клиентов

In the next step, the customer mapping has to be expanded with the new entry for the room number. To ensure that the changes are not lost after an update, put the "CustomerUser" settings from the Kernel/Config/Defaults.pm into the Kernel/Config.pm. Now change the MAP array and add the new room number field, as shown in the script below.

```
# 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.',
    },
},
};

```

Script: Changing the map array.

4.3. Customizing the customer_user table in the OTRS DB

The last step is to add the new room number column to the customer_user table in the OTRS database (see Script below). In this column, the entries for the room numbers will be stored.

```
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:~#
```

Script: Adding a new column to the customer_user table.

Теперь новое поле для номера комнаты должно отображаться в customer.pl панели. Когда новые клиенты будут регистрировать аккаунт, им нужно будет вписать номер комнаты. Если для работы OTRS используется Apache и модуль mod_perl, то чтобы новые изменения вступили в силу нужно перезапустить веб-сервер.

Глава 12. Состояния

1. Предопределенные состояния

OTRS позволяет изменять предопределенные состояния заявок и их типы, а также добавлять новые. Для состояния важны два атрибута: имя (state-name) и тип (state-type).

Предустановленные состояния в OTRS: "закрыто успешно", "закрыто неудачно", "объединено", "новая", "открытая", "в ожидании с автозакрытием+", "в ожидании с автозакрытием-", "в ожидании с напоминанием", "удаленная".

1.1. Новая

Заявки находятся в этом состоянии, когда они создаются на основе входящих сообщений электронной почты.

1.2. Открытая

Это состояние по умолчанию для заявок, которые присвоены очередям или агентам.

1.3. Ожидание с напоминанием

После того как время ожидания истекло, владелец заявки будет получать напоминание на электронную почту. Если заявка не закрыта, то напоминание о заявке будет отправлено всем агентам в очереди. Напоминание о заявках будет отправлено только в рабочее время и будет повторяться каждые 24-часа, пока агент не изменит состояние заявки. Время, которое заявка проведет с таким статусом будет добавлено к времени эскалации.

1.4. Ожидание авто-закрытие -

Если время ожидания вышло, заявки с этим статусом будут установлены в "Закрытые неуспешно". Время, проведенное заявкой в этом статусе будет добавлено к времени эскалации.

1.5. Ожидание авто-закрытие+

Если вышло время ожидания, заявки с этим статусом будут установлены в "Закрыто Успешно". Время, проведенное заявкой в этом статусе будет добавлено к времени эскалации.

1.6. Объединенные

Это состояние для заявок, которые были объединены с другими заявками.

1.7. Закрыта Успешно

Это конечное состояние для заявок, которые были решены успешно. В зависимости от конфигурации, у вас будет или не будет возможности заново открыть ранее закрытые заявки.

1.8. Закрыта Неудачно

Это конечное состояние для заявок которые НЕ были успешно решены. В зависимости от конфигурации, у вас будет или не будет возможности заново открыть ранее закрытые заявки.

2. Настраиваемы состояния

Каждое состояние имеет название (state-name) и тип (state-type). Чтобы создать новое состояние перейдите по ссылке Состояния на Панели Администрирования и нажмите кнопку "Добавить состояние". Можно свободно выбирать имя нового со-

стояния. Типы состояний не могут изменяться посредством веб-интерфейса. Если нужно добавить новые типы или изменить существующие, - все изменения придется делать напрямую в базе данных. Предусмотренные типы состояний не могут быть изменены, поскольку это может привести к непредсказуемым результатам. Например, расчет эскалации и функция разблокирования основаны на конкретных типах состояний.

The name of an already existing state can be changed, or new states added through this screen. If the state "new" has been changed via the web interface, this change also has to be configured via the config file `Kernel/Config.pm` or via the SysConfig interface. The settings specified in the script below have to be modified to ensure that OTRS works with the changed state for "new".

```
[...]  
# PostmasterDefaultState  
# (The default state of new tickets.) [default: new]  
$Self->{PostmasterDefaultState} = 'new';  
  
# CustomerDefaultState  
# (default state of new customer tickets)  
$Self->{CustomerDefaultState} = 'new';  
[...]
```

Script: Modifying the Kernel/Config.pm settings.

If a new state type should be added, the `ticket_state_type` table in the OTRS database needs to be modified with a database client program, as shown in the script below.

```
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:~#
```

Script: Modifying the OTRS database.

На данный момент можно использовать новый тип состояния, который вы только что создали. Как только состояние будет связано с этим новым типом состояния, то чтобы убедиться что новое состояние используется и работает корректно нужно также изменить настройки OTRS. Используя SysConfig внесите изменения в следующие опции:

Ticket -> Frontend::Agent::Ticket::ViewPhoneNew > AgentTicketPhone###StateDefault - определить следующее состояние по умолчанию для новых заявок созданных на основе телефонного звонка.

Ticket -> Frontend::Agent::Ticket::ViewPhoneNew > AgentTicketPhone###StateType - для определения последующих доступных состояний для новых заявок, созданных на основе телефонного звонка.

Ticket -> Frontend::Agent::Ticket::ViewEmailNew > AgentTicketEmail###StateDefault - установка последующих следующих состояний для заявок созданных на базе сообщений электронной почты.

Ticket -> Frontend::Agent::Ticket::ViewEmailNew > AgentTicketEmail###StateType - для определения последующих доступных состояний для новых заявок созданных на базе сообщений электронной почты.

Ticket -> Frontend::Agent::Ticket::ViewPhoneOutbound > AgentTicketPhoneOutbound###State - для определения последующих доступных состояний для новых заявок созданных на базе новых телефонных статей.

Ticket -> Frontend::Agent::Ticket::ViewPhoneOutbound > AgentTicketPhoneOutbound###StateType - для определения последующих доступных состояний для новых заявок созданных на базе новых телефонных статей.

Ticket:Frontend::Agent::Ticket::ViewMove:Ticket::DefaultNextMoveStateType - для определения последующих доступных состояний для перемещенных заявок.

Ticket -> Frontend::Agent::Ticket::ViewBounce > StateDefault - для определения последующих доступных состояний для "подпрыгивающих" заявок.

Ticket -> Frontend::Agent::Ticket::ViewBounce > StateType - для определения последующих доступных состояний для экранов с отказами.

Ticket -> Frontend::Agent::Ticket::ViewBulk > StateDefault - для определения прдеопределенных последующих состояний для групповых действий.

Ticket -> Frontend::Agent::Ticket::ViewBulk > StateType - для определения прдеопределенных последующих состояний для экрана групповых действий

Ticket -> Frontend::Agent::Ticket::ViewClose > StateDefault - для определения прдеопределенных последующих состояний после закрытия заявки

Ticket -> Frontend::Agent::Ticket::ViewClose > StateType - для определения прдеопределенных последующих состояний для страницы закрытия.

Ticket -> Frontend::Agent::Ticket::ViewCompose > StateDefault - для определения прдеопределенных последующих состояний для Compose-страницы

Ticket -> Frontend::Agent::Ticket::ViewCompose > StateType - для определения прдеопределенных последующих состояний для Compose-страницы.

Ticket -> Frontend::Agent::Ticket::ViewForward > StateDefault - для определения прдеопределенных последующих состояний после перенаправления заявки.

Ticket -> Frontend::Agent::Ticket::ViewForward > StateType - для определения прдеопределенных последующих состояний для страницы перенаправления.

Ticket -> Frontend::Agent::Ticket::ViewForward > StateDefault - для определения прдеопределенных последующих состояний для страницы free-text.

Ticket -> Frontend::Agent::Ticket::ViewForward > StateType - для определения прдеопределенных последующих состояний для free text-страницы.

Ticket -> Core::PostMaster > PostmasterDefaultState - для определения прдеопределенных последующих состояний для заявок, созданных с сообщений электронной почты.

Ticket -> Core::PostMaster > PostmasterFollowUpState - для определения прдеопределенных последующих состояний для заявок после последующих, которые должны быть сохранены.

Ticket -> Core::PostMaster > PostmasterFollowUpStateClosed - для определения состояния заявок, после того как придет новый ответ на уже закрытую заявку.

Ticket -> Core::Ticket > ViewableStateType - для определения состояний, которые будут отображаться в различных местах системы, например в Queueview.

Ticket -> Core::Ticket > UnlockStateType - для определения типов состояний для разблокированных заявок.

Ticket -> Core::Ticket > PendingReminderStateType - для определения типов состояний для заявок с напоминанием.

Ticket -> Core::Ticket > PendingAutoStateType - для определения типов состояний для ожидающих заявок.

Ticket -> Core::Ticket > StateAfterPending - для определения состояния заявки установленной в Таймере Авто-Ожидания, если время для настроенного состояния истекло.



Глава 13. Изменение приоритетов заявки

OTRS поставляется с пятью предустановленными уровнями приоритетов, которые можно изменить перейдя по ссылке "Приоритеты" на Панели Администрирования. При создании настраиваемого списка приоритетов, пожалуйста помните, что они сортируются в алфавитном порядке. Также OTRS сортирует заявки в QueueView по их внутреннему номеру (ID).

Примечание

Как и в случае с другими сущностями OTRS, приоритеты не могут быть удалены, а только деактивированы путем установки параметра Действительный в значение *не действительный* или *не действительный-временно*.

Важно

Если был создан новый приоритет, или был изменен уже существующий, то можно также произвести изменения некоторых параметров в SysConfig:

- Ticket:Core::Postmaster::PostmasterDefaultPriority - определяет предустановленный приоритет для всех входящих сообщений электронной почты.
- Ticket:Frontend::Agent:Ticket::ViewPhoneNew:Priority - определяет предустановленный приоритет для новых Заявок созданных на основе Телефонных Звонков.
- Ticket:Frontend::Agent:Ticket::ViewEmailNew:Priority - определяет предустановленный приоритет на странице с новыми Email-Заявками для агентов.
- Ticket:Frontend::Customer:Ticket::ViewNew:PriorityDefault - определяет предустановленные приоритеты для страицы Новые Заявки в пользовательском веб-интерфейсе.

Глава 14. Создание своих собственных тем (шаблонов)

Для OTRS можно создавать свои собственные темы и использовать в качестве веб-интерфейса тот шаблон, который вам нравится больше всего. Для создания собственной темы нужно настроить шаблоны вывода под свои нужды.

More information on the syntax and structure of output templates can be found in the Developer Manual at <http://doc.otrs.org>, especially in the chapter on *templates*.

В качестве примера, выполните следующие шаги для создания новой темы, которая называется "Компания":

1. Создайте директорию и назовите ее `Kernel/Output/HTML/Company` и скопируйте все файлы, которые хотите изменить, из `Kernel/Output/HTML/Standard` в новую директорию.

Важно

Копируйте только те файлы, которые вы действительно планируете изменять. OTRS автоматически получит недостающие файлы из Стандартной темы. На более поздних этапах такой подход позволит сделать модернизацию более простой.

2. Настройте файлы в директории `Kernel/Output/HTML/Company` и измените шаблоны по своему вкусу.
3. Чтобы активировать новую тему, перейдите по ссылке `Frontend::Themes` и добавьте ее в `SysConfig`.

Теперь новая тема должна быть доступна. Вы можете выбрать эту тему перейдя по ссылке предпочтения.

Предупреждение

Не изменяйте файлы темы, поставляемой с OTRS, поскольку эти изменения будут утеряны после обновления. Создавайте свои собственные темы только путем выполнения шагов, описанных выше.

Глава 15. Локализация интерфейса OTRS

OTRS предлагает многоязычную поддержку для своего веб-интерфейса.

Procedures for localization for the OTRS framework, steps to be followed to create a new language translation, as well as procedures for translation customizations, can be found in the "[Language Translations](#)" chapter from the developer manual on <http://doc.otrs.org>.



Глава 16. PGP

OTRS имеет возможность подписывать или шифровать исходящие сообщения с помощью PGP. Также, могут быть расшифрованы входящие зашифрованные сообщения. Шифрование и дешифрование осуществляется с помощью GPL-инструмента GnuPG. Чтобы установить GnuPG для OTRS, нужно проделать следующие действия:

1. Установка GnuPG с помощью менеджера пакетов вашей операционной системы.
2. Configure GnuPG for use with OTRS. The necessary directories for GnuPG and a private key have to be created. The command shown in the script below has to be executed as user 'otrs' from a shell.

```
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 addre
ss support@example.com) <support@example.com>
sub 2048g/52B97069 2006-02-03

linux:~$

```

Script: Configuring GnuPG.

As shown in the script below, the default settings can be applied for most of the required parameters. Only the values for the key owner have to be entered correctly, with a proper password specified for the key.

3. Теперь OTRS готов для использования PGP. Через Панель Администрирования откройте интерфейс SysConfig и найдите ссылку "PGP". Из результатов поиска выберите подгруппу Crypt::PGP.

На странице с PGP-настройками, PGP должен быть активирован для OTRS (первая опция). Также нужно установить и проверить путь к gpg-программе.

Следующим нужно изменить конфигурационный параметр (PGP::Options). Используя эти конфигурационные настройки можно определить параметры, которые будут использоваться для каждого вызова gpg пользователем 'otrs'. Каталог с конфигурационными файлами для GnuPG является очень важным. В примере используется /opt/otrs/.gnupg. Эта директория была создана на первом шаге конфигурации PGP.

Используя следующий конфигурационный параметр (PGP::Key::Password), можно указать пары для ключей ID и их пароли для собственных закрытых ключей. Поскольку партнеры извне пишут зашифрованные сообщения в систему заявок используя ваш публичный ключ, OTRS может расшифровать эти сообщения с помощью ID/паролей указанных здесь.

How to get the id of your own private key? The ID of your own private key is already shown during the key generation (see step 1 from above). It is also possible to get the ID if the command specified in the following script is executed as user 'otrs':

```

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 pgp key for ticket system with  
address support@example.com) <support@example.com>  
sub 2048g/52B97069 2006-02-03  
  
linux:~$
```

Script: Getting the ID of your own private key.

Идентификатор (ID) закрытого ключа можно найти в строке, которая начинается с "sub". Это шестнадцатеричное представление строки, длиной в восемь символов, например "52B97069". Пароль, который нужно указать для этого ключа в системе обработки заявок такой же как данные при генерации ключей.

После того, как эти данные были введены, нажмите кнопку "Обновить" чтобы сохранить настройки. Теперь OTRS готов принимать и расшифровывать зашифрованные сообщения.

4. И наконец, нужно импортировать открытый ключ клиента. Это гарантирует, что зашифрованные сообщения могут быть отправлены этому клиенту. Есть два способа импортировать открытый ключ клиента.

Первый способ - указать открытый ключ клиента в веб-интерфейсе управления клиентами.

Вторая возможность заключается в определении ключей посредством настройки PGP, которая доступна через Панель Администрирования. В правой стороне этой страницы отображаются все импортированные открытые ключи клиентов. После того как PGP активирован и сконфигурирован, здесь также должен отображаться ваш собственный открытый ключ. В левой части страницы настроек PGP можно производить поиск по ключам. Также, новый открытый ключ может быть загружен в систему из файла.

Файлы с открытым ключем, которые нужно импортировать в OTRS должны соответствовать файлам ключей GnuPGP. В большинстве случаев, ключ сохраняется в файле как "ASCII armored key". OTRS позволяет работать с этим форматом.

Глава 17. S/MIME

На первый взгляд, S/MIME-шифрование кажется более сложным чем PGP. Прежде всего для OTRS нужно создать Центр Сертификации - Certification Authority (CA). Последующие процедуры очень похожи на те, которые вы выполняли для PGP: настроить OTRS, установить собственные сертификаты, импортировать другие открытые сертификаты (если необходимо), и т.д.

The S/MIME configuration is conducted outside the OTRS web interface for the most part, and should be carried out in a shell by the 'otrs' user. The MIME configuration under Linux is based on SSL (OpenSSL). Therefore, check first of all whether the OpenSSL package is installed on your system. The OpenSSL package includes a script called CA.pl, with which the most important steps of certificate creation can be performed. To simplify the procedure, find out where in the filesystem the CA.pl script is stored and enter the location temporarily into the PATH variable of the shell (see Script below).

```
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>
```

Script: Configuring S/MIME.

The script above shows that a new temporary directory ~/tmp has been created, in which the certificate is to be generated.

Для создания сертификата, проделайте следующие операции в командной строке (предположим, что OTRS-администратор должен создать SSL-сертификат с целью тестирования и обучения. В случае, если у вас уже есть SSL-сертификат для шифрования, используйте его и пропустите следующие шаги):

1. Establish your own Certification Authority for SSL. You need it to certify the request for your own SSL certificate (see Script below).

```
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>
```

Script: Establishing a Certification Authority for SSL.

2. Generate a certificate request (see Script below).

```
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>
```

Script: Creating a certificate request.

3. Signing of the certificate request. The certificate request can either be signed and thereby certified by your own CA, or made more credible by being signed by another external certified CA (see Script below).

```
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>

```

Script: Signing of the certificate request.

4. Generate your own certificate, and all data going with it, using the signed certificate request (see Script below).

```

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>

```

Script: Generating a new certificate.

Теперь эти операции выполнены, установка S/MIME в OTRS должна быть завершена.

Эта часть установки осуществляется в Панели Администрирования, выбрав ссылку "SMIME". В случае, если в OTRS нету поддержки S/MIME, маска указывает администратору удобные ссылки чтобы активировать поддержку S/MIME.

Используя группу "Crypt::SMIME" в SysConfig, также можно включить и настроить поддержку S/MIME.

Здесь можно активировать поддержку S/MIME, определить пути для команд OpenSSL, и директории для сертификатов. Файл ключей должен храниться в директории, указанной здесь. В противном случае OpenSSL не будет их использовать.

Следующим шагом, который следует выполнить - перейти по ссылке Настройка S/MIME в Панели Администрирования. Здесь можно импортировать закрытые ключи OTRS-системы и открытые ключи других партнеров. Введите публичный ключ, который был создан и добавлен в OTRS в начале этого раздела.

Перейдя по ссылке Инструменты администрирования клиентов можно импортировать все открытые S/MIME ключи партнеров по общению.



Глава 18. Списки Контроля Доступа (ACLs)

1. Introduction

From OTRS 2.0 on, Access Control Lists (ACLs) can be used to control access to tickets, modules, queues, etc., or to influence actions on tickets (closing, moving, etc.) in certain situations. ACLs can be used to supplement the existing permission system of roles and groups. Using ACLs, rudimental workflows within the system can be mapped, based on ticket attributes.

As yet, ACLs cannot be created using the SysConfig interface. They must be directly entered into the Kernel/Config.pm file. This chapter has some ACL examples which will walk you through the process of defining ACL definitions, and a reference of all possible important ACL settings.

2. Examples

Пример 18.1. Списки прав доступа (ACL) позволяют перемещать в очереди только заявки с приоритетом 5.

This example shows you the basic structure of an ACL. First, it needs to have a name. In this case, it is "ACL-Name-2". Note that the ACLs will be numerically sorted before execution, so you should use the names carefully.

Secondly, you have a "Properties" section which is a filter for your tickets. All the criteria defined here will be applied to a ticket to determine if the ACL must be applied or not. In our example, a ticket will match if it is in the queue "Raw" and has priority "5 very high". This is also affected by changes in the form (e.g. if the ticket is in the queue "raw" and had a priority "3 normal", but then priority drop-down is selected and the priority is changed now to "5 very high" will also match).

Lastly, a section "Possible" defines modifications to the screens. In this case, from the available queues, only the queue "Alert" can be selected in a ticket screen.

```
# 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'],
    },
  },
};
```

Пример 18.2. ACL allowing movement into a queue of only those tickets with ticket priority 5 stored in the database.

This example is very similar to the last one, but in this case only tickets in the queue "Raw" and with a priority "5 very high", both stored in the database will match. This kind of ACLs does not consider changes in the form before the ticket is really updated in the database.

```
# ticket acl
```

```
$Self->{TicketAcl}->{'100-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'],
    },
  },
};
```

Please note that PropertiesDatabase is available since OTRS 3.1.5.

Пример 18.3. Списки прав доступа (ACL) делают невозможным закрытие заявок в очереди raw и скрывают кнопку "закрыть".

Here you can see how a ticket field (state) can be filtered with more than one possible value to select from. It is also possible to limit the actions that can be executed for a certain ticket. In this case, the ticket cannot be closed.

```
$Self->{TicketAcl}->{'101-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'],
    },
  },
  # possible action options
  Action => {
    AgentTicketBounce => 1,
    AgentTicketClose => 0,
    AgentTicketCompose => 1,
    AgentTicketCustomer => 1,
    AgentTicketForward => 1,
    AgentTicketFreeText => 1,
    AgentTicketHistory => 1,
    AgentTicketLink => 1,
    AgentTicketLock => 1,
    AgentTicketMerge => 1,
    AgentTicketMove => 1,
    AgentTicketNote => 1,
    AgentTicketOwner => 1,
    AgentTicketPending => 1,
    AgentTicketPhone => 1, # only used to hide the Split action
    AgentTicketPhoneInbound => 1,
    AgentTicketPhoneOutbound => 1,
    AgentTicketPrint => 1,
    AgentTicketPriority => 1,
    AgentTicketResponsible => 1,
    AgentTicketWatcher => 1,
    AgentTicketZoom => 1,
    AgentLinkObject => 1, # only used to hide the Link action
  },
};
```

Пример 18.4. ACL removing always state closed successful.

This example shows how it is possible to define negative filters (the state "closed successful" will be removed). You can also see that not defining match properties for a ticket will match any ticket, i. e. the ACL will always be applied. This may be useful if you want to hide certain values by default, and only enable them in special circumstances (e. g. if the agent is in a specific group).

```
$Self->{TicketAcl}->{'102-Third-ACL-Example'} = {
  # match properties
  Properties => {
    # current ticket match properties (match always)
  },
  # return possible options
  PossibleNot => {
    # possible ticket options
    Ticket => {
      State => ['closed successful'],
    },
  },
};
```

Пример 18.5. ACL only showing Hardware services for tickets that are created in queues that start with "HW".

This example also shows you how you can use regular expressions for matching tickets and for filtering the available options.

```
$Self->{TicketAcl}->{'Only-Hardware-Services-for-HW-Queues'} = {
  # 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)'],
    },
  },
};
```

3. Reference

In the example below there is a list of all parameters which can be used for ACLs.

Пример 18.6. Reference showing all possible important ACL settings.

```
# ticket acl
$Self->{TicketAcl}->{'200-ACL-Reference'} = {
  # match properties (current values from the form)
  Properties => {

    # the used frontend module
    Frontend => {
      Action => ['AgentTicketPhone', 'AgentTicketEmail'],
    },
    # the logged in agent
    User => {
      UserLogin => ['some login'],
      # ...
      Group_rw => [
```

```

        'hotline',
    ],
    # ...
    Role => [
        'admin',
    ],
    # ...
},
# the logged in customer
CustomerUser => {
    UserLogin => ['some login'],
    # ...
    Group_rw => [
        'hotline',
    ],
    # ...
    Role => [
        'admin',
    ],
    # ...
},

#
# ticket properties
#

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 => {
    # keys must be in DynamicField_<field_name> format
    DynamicField_Field1      => ['some value'],
    DynamicField_OtherField  => ['some value'],
    DynamicField_TicketFreeText2 => ['some value'],
    # ...
  },
  # alternatively, ticket properties can be specified in the ticket hash
  Ticket => {
    Queue           => ['Raw'],
    State           => ['new', 'open'],
    Priority         => ['some priority'],
    Lock            => ['lock'],
    CustomerID      => ['some id'],
    CustomerUserID  => ['some id'],
    Owner           => ['some owner'],
    DynamicField_Field1 => ['some value'], # Must be the untranslated values
                                           # specified in the dynamic field
                                           # definition and not the IDs

    DynamicField_MyField => ['some value'],
    # ...
  },
},

# match properties (existing values from the database)
# This section is available since OTRS 3.1.5.
PropertiesDatabase => {
  # See section "Properties", the same config can be used here.
  # ...
}

# return possible options (white list)
Possible => {
  # possible ticket options (white list)
  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'],
    # ...
  },

  # possible action options (white list)
  Action => {
    AgentTicketBounce      => 1,
    AgentTicketClose       => 1,
    AgentTicketCompose     => 0,
    AgentTicketCustomer    => 0,
    AgentTicketForward     => 0,
    AgentTicketFreeText    => 1,
    AgentTicketHistory     => 1,
    AgentTicketLink        => 0,
    AgentTicketLock        => 1,

```

```

AgentTicketMerge      => 0,
AgentTicketMove       => 1,
AgentTicketNote       => 1,
AgentTicketOwner      => 1,
AgentTicketPending    => 1,
AgentTicketPhone      => 1, # only used to hide the Split action
AgentTicketPhoneInbound => 0,
AgentTicketPhoneOutbound => 1,
AgentTicketPrint      => 1,
AgentTicketPriority    => 0,
AgentTicketResponsible => 1,
AgentTicketWatcher    => 1,
AgentTicketZoom       => 1,
AgentLinkObject       => 1, # only used to hide the Link action
    },
},
# remove options (black list)
PossibleNot => {
    # See section "Possible"
    # ...
},
};

```

Примечание

While matching ACLs if CustomerUserID parameter sent, the ACL mechanism will compare the defined ACLs using the supplied CustomerUserID to gather the CustomerUser details to fill the CustomerUser hash and it also overrides the Customer information in the Ticket hash for the Properties match. On the other hand this calculations are also made for the PropertiesDatabase part, but using the Ticket Customer as the basis to gather the data.

Notice that in Customer Interface, the CustomerUserID is always sent with the current logged Customer User.

Be aware that in ticket search screens (AgentTicketSearch and CustomerTicketSearch) the only affected attributes by ACLs are the Dynamic Fields. This means that this screens you can not restrict any other attribute like ticket type, state, queue, etc.

Глава 19. Stats-модуль

Модуль статистики OTRS дает возможность оперативно отслеживать статистику и создавать пользовательские отчеты, связанные с использованием OTRS. Система OTRS использует общеупотребляемый термин "stat" для обозначения отчетов с различными показателями.

Правильная настройка модуля статистики OTRS связана с множеством различных условий. К ним относятся различные модули системы OTRS, подлежащие оценке, настройки прав доступа пользователей, признаки, которые должны быть рассчитаны и уровень их сложности, легкость настройки модуля статистики, скорость и эффективность вычислений, поддержка множества вариантов вывода информации.

Статистические элементы, такие как например файлы, которые дополняют функциональные возможности модуля статистики для конкретных требований, могут быть интегрированы для подсчета сложной статистики.

1. Обработка модуля агентом

When signed on as an agent, the navigation bar displays the link "Statistics". This has various submenu options, as shown in Figure.

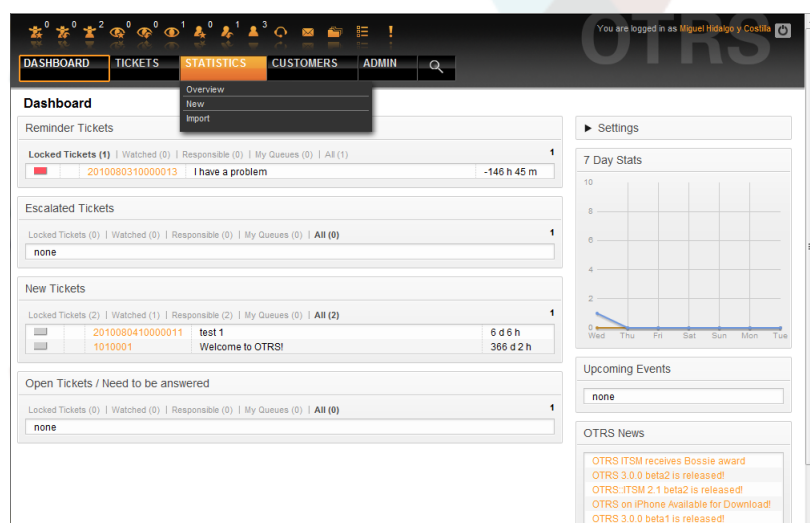


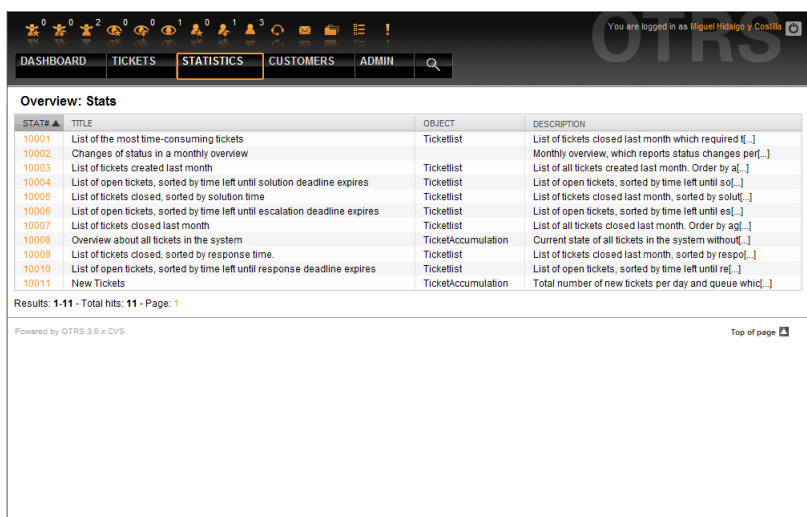
Figure: Statistics menu options.

Различные варианты представленные в меню статистики:

- **Overview.** Предоставляет список различных предварительно сконфигурированных отчетов.
- **Новая.** Требуется права rw.
- **Импорт.** Требуется rw-права.

1.1. Обзор

Selecting the "Statistics" link in the navigation bar, and then the submenu link "Overview", calls up the Overview screen. The Overview screen presents a list of all pre-configured reports the agent can use (see Figure below).



The screenshot shows the OTRS web interface. At the top, there is a navigation bar with 'DASHBOARD', 'TICKETS', 'STATISTICS' (highlighted), 'CUSTOMERS', and 'ADMIN'. Below this is a search bar and a user profile indicator. The main content area is titled 'Overview: Stats' and contains a table of reports. The table has four columns: 'STAT#', 'TITLE', 'OBJECT', and 'DESCRIPTION'. Below the table, it shows 'Results: 1-11 - Total hits: 11 - Page: 1'. At the bottom, it says 'Powered by OTRS 3.0.x CVS' and 'Top of page'.

STAT#	TITLE	OBJECT	DESCRIPTION
10001	List of the most time-consuming tickets	Ticketlist	List of tickets closed last month which required [...]
10002	Changes of status in a monthly overview	Ticketlist	Monthly overview, which reports status changes per[...]
10003	List of tickets created last month	Ticketlist	List of all tickets created last month. Order by a[...]
10004	List of open tickets, sorted by time left until solution deadline expires	Ticketlist	List of open tickets, sorted by time left until so[...]
10005	List of tickets closed, sorted by solution time	Ticketlist	List of tickets closed last month, sorted by solu[...]
10006	List of open tickets, sorted by time left until escalation deadline expires	Ticketlist	List of open tickets, sorted by time left until es[...]
10007	List of tickets closed last month	Ticketlist	List of all tickets closed last month. Order by ag[...]
10008	Overview about all tickets in the system	TicketAccumulation	Current state of all tickets in the system without[...]
10009	List of tickets closed, sorted by response time.	Ticketlist	List of tickets closed last month, sorted by respo[...]
10010	List of open tickets, sorted by time left until response deadline expires	Ticketlist	List of open tickets, sorted by time left until re[...]
10011	New Tickets	TicketAccumulation	Total number of new tickets per day and queue whic[...]

Figure: Overview of the standard reports.

Для каждого из отчетов, перечисленных в Обзоре предоставлена следующая информация:

- *Stat#*. Уникальный номер отчета.
- *Заголовок*. Заголовок отчета.
- *Объект*. Объект, который используется для генерации статистики. В случае статической статистики не отображается ни один объект, поскольку в ее генерации динамические объекты не участвуют.
- *Описание*. Краткое описание отчета.

Когда модуль статистики установлен, он поставляется с несколькими предустановленными отчетами, выборочно встроенными в систему. Они приведены в виде списка на странице "Обзор". Если список слишком большой и не помещается на одну страницу, у агента есть возможность просматривать другие страницы. Список отчетов может быть отсортирован по своему вкусу, путем нажатия на желаемый заголовок столбца в списке. Чтобы создать конкретный отчет, нажмите на соответствующие stat-номер, связанный с отчетом. В результате откроется интерфейс просмотра отчета.

1.2. Создание и просмотр отчетов

The view user interface provides the stat's configuration settings (see Figure below).

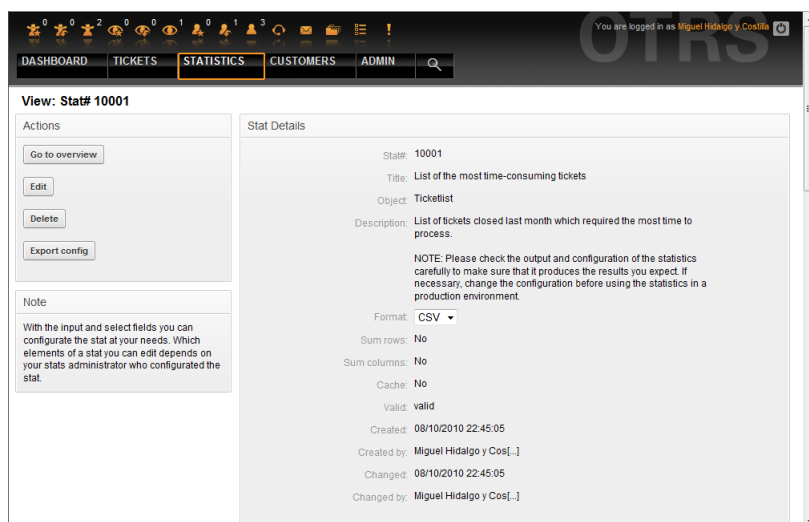


Figure: Viewing a specific report.

Параметры конфигурации для конкретного отчета могут быть установлены путем выбора ряда возможных опций. Как автор отчета так и другие пользователи с соответствующими правами доступа могут создавать настройки.

На странице отображается следующее:

- Возможные действия:
 - *Перейти к обзору.* Ссылка, ведущая обратно к Просмотру списка отчетов.
 - *Редактировать.* Редактировать имеющуюся структуру отчета (требуются права для чтения и записи, rw).
 - *Удалить.* Удалить текущий отчет (требуются права для чтения и записи, rw).
 - *Экспортировать настройки.* Экспортировать настройки отчета с помощью загрузки файла (требуются права для чтения и записи, rw).

Традиционно: функции Экспорт и Импорт позволяют создавать и тестировать отчеты та тестовых системах с их легкой дальнейшей интеграцией в производственную систему.

- Подробности отчета:
 - *Отчет#.* Номер отчета.
 - *Заголовок.* Заголовок отчета.
 - *Объект.* Объект, который используется для создания отчета.
 - *Описание.* Описание цели отчета.
 - *Формат.* Выходной формат отчета. В зависимости от конфигурации, можно выбрать один из следующих форматов вывода:
 - CSV.
 - Печать.
 - Графические линии.
 - Столбцовая диаграмма.

- Диаграмма.
- Точечная диаграмма.
- Точечно-линейная диаграмма.
- Диаграмма
- Круговая "пирожок"-диаграмма.
- *Размер в пикселях.* Размер графиков/диаграмм в пикселях. Эта опция доступна только в том случае, если в настройках отчета разрешены графики. Все часто используемые графические размеры настраиваются OTRS-администратором в SysConfig. Затем, во время настройки отчета, агент может выбрать необходимый формат из всех доступных.
- *Количество строк.* Указывает когда отчет заменяется колонкой, чьи ячейки содержат сумму соответствующих строк.
- *Количество колонок.* Указывает когда отчет заменяется строкой, чьи ячейки содержат сумму соответствующих колонок.
- *Кэш.* Указывает когда сгенерированный отчет кэшируется в файловой системе.
- *Действительный.* Может быть установлен в значение "недействительный" если по какой то причине отчет временно не должен запускаться. Кнопка "Старт" в нижней части правой панели больше не отображается. Таким образом отчет не может быть создан.
- *Создан.* Время создания отчета.
- *Создан (кем).* Имя агента, который создал отчет.
- *Изменен.* Время последней модификации отчета.
- *Изменен кем.* Имя агента, который последним внес изменения в отчет.
- *X-axis.* Используя эту функцию, агент может переключать x и y axes (только в том случае, если эта опция активирована OTRS-администратором).
- Общая информация сопровождается информацией о самом отчете. Есть два различных шаблона просмотра отчета (или статистики):
 - *Static stat view.* Static report generators can be integrated into the stats module (see Figure below).

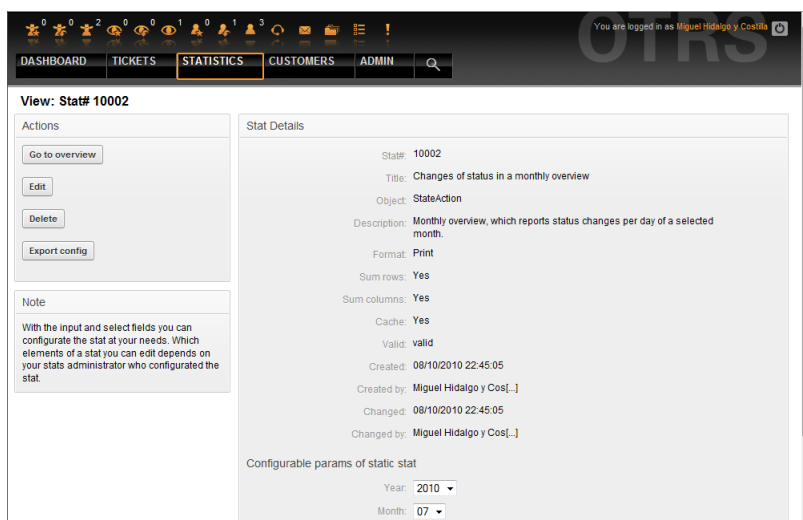


Figure: Viewing a static report.

- *Dynamic stat view* (see Figure above). They can be displayed in two different ways:
 - *Неизменяемые настройки*. Создатель отчета не имеет прав для изменения этих полей.
 - *Изменяемые настройки*. Параметры конфигурации таких отчетов могут быть изменены агентом.

Нажатие кнопки "Пуск" (в нижней части экрана) является последним шагом при создании отчета. Есть две возможные причины, когда эта кнопка не будет отображаться:

1. Отчет был установлен недействительным и, таким образом отключен.
2. Отчет был настроен не аккуратно и, следовательно, не может выполняться. В этом случае необходимую информацию можно найти в разделе уведомлений OTRS (внизу, под навигационной панелью).

Если настройки на странице Просмотр неверны, эта страница отображается опять, после нажатия кнопки "Пуск" и информация о введенных неверных данных будет выведена в разделе уведомлений.

1.3. Редактировать / Создать новый

Агенты, которые имеют права на запись, могут редактировать настройки существующего отчета, запустив пользовательский веб-интерфейс модуля статистики. В качестве альтернативы они могут создать новый отчет. Соответствующие страницы могут быть запущены следующим образом:

1. Редактировать: Используя кнопку "Редактировать" в шаблоне просмотра статистики.
2. Новый: Используя ссылку "Новый" в меню Статистика навигационной панели, или кнопку "Добавить" на странице Просмотр.

Статистика редактируется с помощью мастера в четыре этапа:

1. Общие технические условия.
2. Определение элементов для X-axis.

3. Спецификация значений ряда.

4. Выбор ограничений для отчета.

Шаги с 2-4 необходимы только для создания отчетов с динамической статистикой. Для статической статистики требуется только общая информация (пункт 1).

Информация о том, как обрабатывать страницу приводится на каждом из этих экранов, ниже панели "Действия" панели Подсказок.

Если введены неправильные данные, предыдущий пользовательский интерфейс отображается заново, на этот раз уже с информацией о неправильных входных данных. Эта информация отображается в OTRS в разделе уведомлений. Следующий пользовательский веб-интерфейс для ввода отображается только в том случае, если все поля формы заполнены правильно.

1. *General specifications*. It is the first page of the Edit wizard (see Figure below).

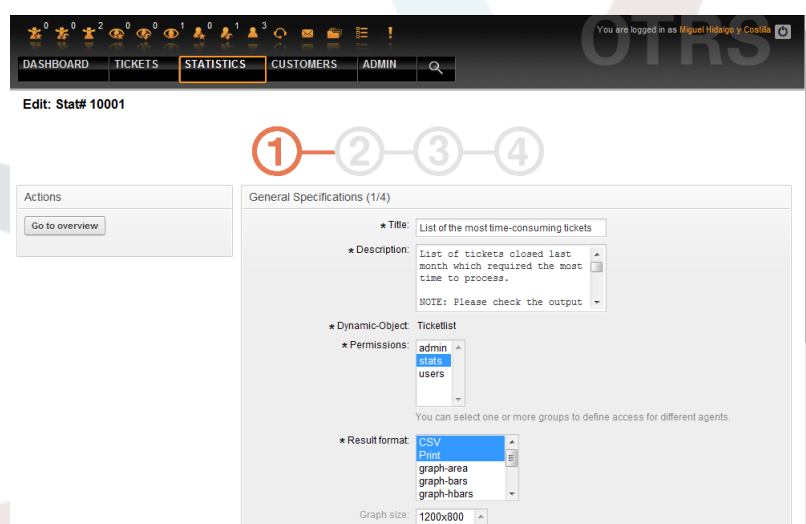


Figure: Editing the general specifications of a report.

In the screen showed in Figure, there are a great number of common specifications and settings that can be edited:

- **Заголовок.** Должны в сжатой форме отражать назначение (цель).
- **Описание.** Более подробная информация об отчете, типах конфигурационных параметров и т.д.
- **Динамический объект.** Если установленная система OTRS предоставляет различные динамические объекты, один из них может быть выбран. Объекты отвечают требованиям отдельных модулей.
- **Static file.** Обычно эта возможность не отображается до тех пор, пока отображаются статические файлы, назначенные отчету. Если "Статический файл" отображается, позже, очень важно поставить галочку в поле и выбрать режим генерации (динамический с динамический объектом или статический с файлом). Если выбран статический файл, пользовательский веб-интерфейс 2-4 не отображается как статический файл содержащий все необходимые настройки.
- **Настройки прав доступа.** Содействие ограничению групп (и, следовательно, агентов), которые позже могут просматривать и генерировать предварительные отчеты. Таким образом, различные отчеты могут быть выделена для раз-

личных отделов и рабочих групп, которые в них нуждаются. Можно выделить одну статистику для различных групп.

Пример 1: Выбрана "stats" группа. Отчет доступен для просмотра для всех пользователей имеющих по крайней мере права на группу "stats". Этот доступ доступен по умолчанию.

Пример 2: Была выбрана группа под названием "продажи". Все пользователи, имеющие права ro (read only) к этой группе могут видеть статистику в режиме просмотра и генерировать ее. Однако отчет не будет доступен для просмотра другим пользователям.

- **Формат.** Формат отображения статистики. В зависимости от конфигурации, могут быть выбраны следующие один или несколько форматов:
 - CSV.
 - Печать.
 - graph-lines.
 - graph-bars.
 - graph-hbars.
 - graph-points.
 - graph-lines-points.
 - graph-area.
 - graph-pie.
- **Размер графика.** Выберите размер диаграммы в пикселях. Этот выбор необходим только в том случае, если графический формат вывода был выбран в "Формат". Все графические размеры, которые можно использовать определяются в OTRS в SysConfig. При настройке отчета, агент может предварительно выбрать все соответствующие форматы.
- **Количество строк.** Указывает когда отчет заменяется колонкой, чьи ячейки содержат сумму соответствующих строк.
- **Сумма столбцов.** Указывает когда отчет заменяется строкой, чьи ячейки содержат сумму соответствующих строк.
- **Кэш.** Указывает на то, что созданный отчет должен кэшироваться в файловой системе. Это позволяет существенно экономить вычислительные мощности и время, но должно использоваться только в том случае если содержание отчета больше не будет изменяться.

Кэширование автоматически прекращается, если отчет не содержит временных обозначений, или когда временные обозначения указывают на будущее.

В случае редактирования закэшированного отчета удаляются все закэшированные данные.

- **Действительный.** Устанавливается в значение "недействительный" если ранее сконфигурированный отчет по какой то причине не должен выполняться. Кнопка "Статистика" также не отображается в правом нижнем углу блока. Отчет больше не генерируется.

2. *Definition of the element for the X-axis.* It is the configuration of the element used for the depiction of the X-axis or, if tables are used, of the column name applied to the X-axis (see Figure).

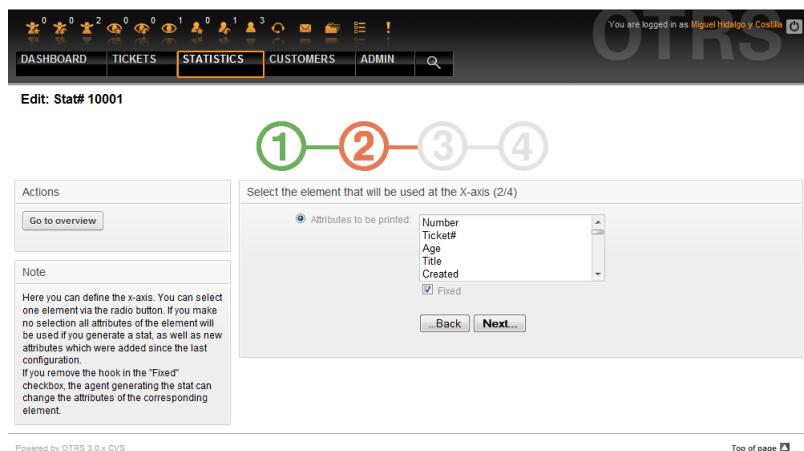


Figure: Definition of the element for the X-axis.

Во-первых, элемент выбирается с помощью опционального поля. Потом нужно выбрать два или более атрибутов элемента. Если выбранных атрибутов нету, используются все атрибуты, в том числе и те, которые были добавлены после конфигурации отчета.

Если отключена настройка "Фиксированный", агент, генерирующий отчет, может изменить атрибуты соответствующего элемента в меню "Вид" пользовательского веб-интерфейса.

Элементы времени отличаются от периодов времени. Тип и число элементов зависит от использующихся динамических объектов и сильно зависит от этого.

Если все входные данные введены правильно, нажатие кнопки "Следующий шаг" открывает форму "Серия значений". Кроме этого, есть возможность заново вернуться к редактированию предыдущих разделов.

3. Спецификация серии значений.

In the third step of the report configuration, the value series are defined (see Figure below). They will later form the individual graphs or the various series within a tabular view.

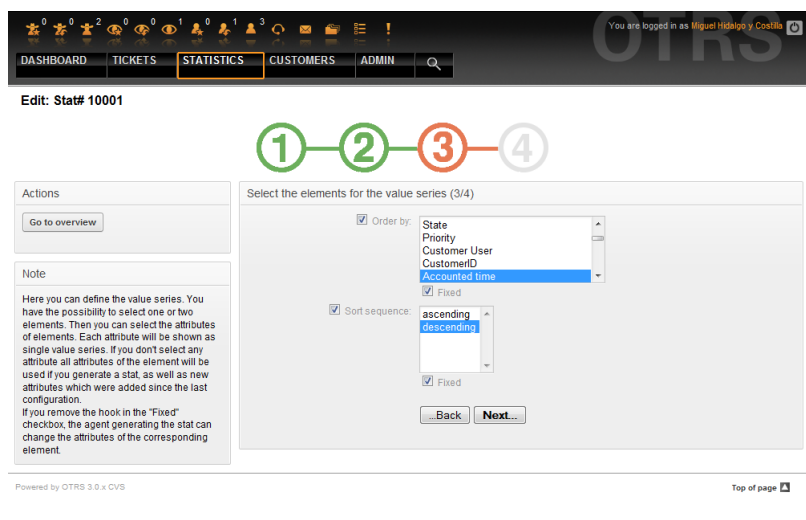


Figure: Definition of the value series.

Если выбран определенный элемент, каждый выбранный атрибут будет соответствовать значению серии (см. ниже Пример 19-1).

Пример 19.1. Определение значения серии - один из элементов

Элемент Очередь:

- Серия значений 1 = Raw
- Значение серии 2 = Junk
-

Если для формирования значения ряда, выбрано два элемента, каждый выбранный атрибут первого элемента комбинируется с атрибутом второго элемента (см. ниже Рисунок 19-2).

Пример 19.2. Определение значения серии - один из элементов

Элемент 1 очередь, Элемент 2 статус:

- Значение создания 1 = Raw - открыть
- Серия значений 2 = Raw - успешно закрыта
- Серия значений 3 = Junk - открыто
- Серия значений 4 = Junk - успешно закрыто

Выбор из трех или более элементов не допускается.

Кроме этого эти условия распространяются и на выбор атрибутов и флажок "Основные" как для "X-axis":

- Если для элемента не выбрано никаких атрибутов, то используются все атрибуты, включая и те, которые были добавлены после настройки отчета.

- Если параметр "Фиксированный" отключен, агент, который создает отчет может изменять атрибуты соответствующих элементов.

4. *Setting restrictions to the report.* This is the fourth and final step of the configuration (see Figure below). The restrictions serve to limit the results to the selected criteria. In many cases, no restrictions at all may be set up.

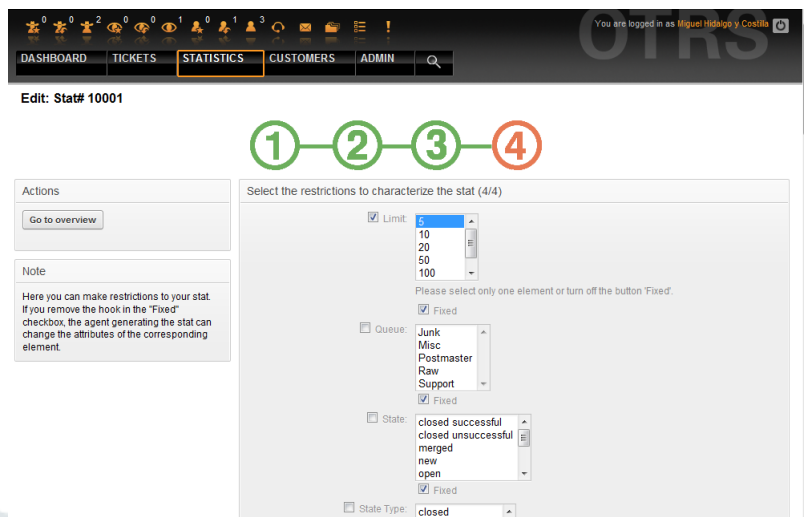


Figure: Definition of restrictions.

После установки всех ограничений, конфигурация отчета завершается нажатием кнопки "Готово".

1.4. Импорт

The Import user interface (see Figure below) can be accessed by choosing from the navigation bar, the link "Statistics", then "Import". Alternately, pressing the Import button on the Overview screen achieves the same result. "rw" rights to the report are required.

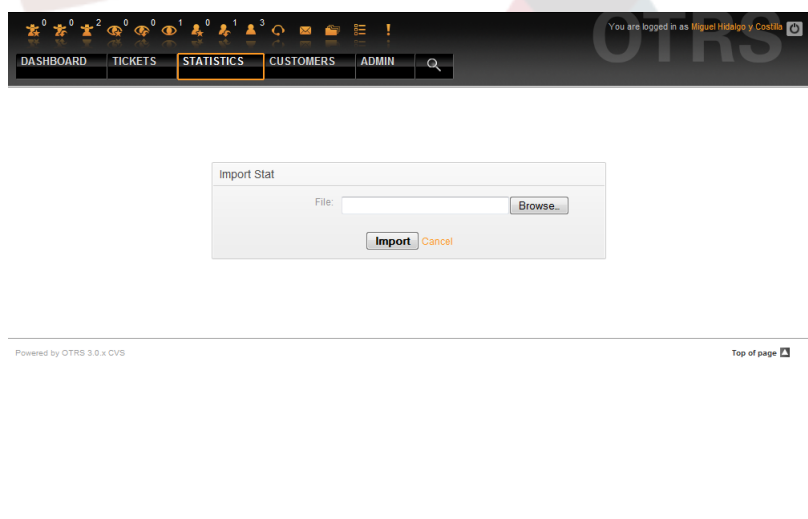


Figure: The Import user interface.

Очень удобная функция, значительно облегчает импорт статистики, особенно в сочетании с возможностями модуля экспорта. Статистику можно создать и протестировать на тестовых системах, а затем импортировать в реальную систему.

Импорт осуществляется путем простой загрузки файла. После этого, для импорта статистики, автоматически открывается пользовательский интерфейс "Вид".

2. Администрирование модуля статистики OTRS администратором.

Этот раздел предоставляет информацию о задачах и функциях администратора OTRS, который работает с модулем статистики.

2.1. Настройки полномочий, Групп и Ролей

Нет новых очередей и/или групп, созданных при установке модуля статистики.

В конфигурации по умолчанию модуль регистрации дает всем агентам группы "статистика" права доступа к модулю статистики.

Доступ в соответствии с разрешенными настройками:

- *rw*. Разрешает настройку статистики и отчетов.
- *ro*. Разрешает генерацию предварительно сконфигурированной статистики и отчетов.

OTRS-администратор самостоятельно определяет когда агенты с правами для генерации предварительно настроенных отчетов выделяются в *ro*-права для группы "статистика", или если их соответствующие группы добавляются в модуль регистрации в SysConfig.

2.2. SysConfig

Группы SysConfig Framework:Core::Stats, Framework:Core::Stats::Graph and Framework:Frontend::Agent::Stats содержат все параметры конфигурации базового набора из модуля статистики. Больше того, конфигурационный параметр `$Self->{'Frontend::Module'}->{'AgentStats'}` управляют организацией и регистрацией модулей и иконкой в модуле статистики.

3. Администрирование модуля статистики системным администратором

Как правило, системного администратора не требуется для эксплуатации, настройки и обслуживания модуля статистики. Однако, немного справочной информации дано для системного администратора в этой точке.

Примечание

Пути к файлам см. в подкаталогах домашнего каталога OTRS (в большинстве случаев `/opt/otrs`).

3.1. Таблица базы данных

Включены все конфигурационные параметры отчета и управляются в XML и, следовательно, хранятся в таблице базы данных "xml_storage". Другие модули, контент которых хранится в xml-формате также используют эту таблицу.

3.2. Список файлов

Следующие файлы необходимы для корректной работы модуля статистики:

- Kernel/System/Stats.pm
- Kernel/Modules/AgentStats.pm
- Kernel/System/CSV.pm

- Kernel/Output/HTML/Standard/AgentStatsOverview.dtl
- Kernel/Output/HTML/Standard/AgentStatsDelete.dtl
- Kernel/Output/HTML/Standard/AgentStatsEditSpecification.dtl
- Kernel/Output/HTML/Standard/AgentStatsEditRestrictions.dtl
- Kernel/Output/HTML/Standard/AgentStatsEditXaxis.dtl
- Kernel/Output/HTML/Standard/AgentStatsEditValueSeries.dtl
- Kernel/Output/HTML/Standard/AgentStatsImport.dtl
- Kernel/Output/HTML/Standard/AgentStatsPrint.dtl
- Kernel/Output/HTML/Standard/AgentStatsView.dtl
- Kernel/System/Stats/Dynamic/Ticket.pm
- bin/otrs.GenerateStats.pl

3.3. Кэширование

Результаты статистики должны быть сохранены в кэше или не могут быть установлены в конфигурации. Кэшированные результаты отчетов хранятся в файлах в каталоге var/tmp установленной системы OTRS (в большинстве случаев /opt/otrs/var/tmp).

Кэшированная статистика может быть распознана с помощью префикса "Stats".

В случае потери данных системе не будет принесено серьезного ущерба. При вызове модуля статистики в следующий раз, в случае если он не обнаружит файл, отчет сгенерируется заново. Конечно, это может занять немного больше времени.

3.4. otrs.GenerateStats.pl

Этот файл хранится в директории bin. Он позволяет генерировать статистику прямо из командной строки.

As an example, see the command line call in the following script.

```
bin> perl otrs.GenerateStats.pl -n 10004 -o /output/dir
```

Script: Generating a report from the command line.

Отчет из конфигурации статистики "Stat# 10004" генерируется и хранится в csv-формате в директории /output/dir.

The generated report can also be sent as an e-mail. More information can be called up with the command in the script below.

```
bin> perl otrs.GenerateStats.pl --help
```

Script: Getting information about the otrs.GenerateStats.pl file.

3.5. Автоматическое создание статистики с помощью планировщика заданий - Cronjob

Очевидным является тот факт, что в действительности, нету необходимости генерировать отчеты вручную, из командной строки, так как модуль статистики име-

ет удобный графический веб-интерфейс пользователя. Однако, создание отчетов вручную имеет смысл в сочетании с Cronjob.

Представьте себе следующую ситуацию: Каждый первый день месяца, руководители отделов хотят получать отчет за последний месяц. Объединив cronjob с командой запуска сценария в командной строке, статистика может отправляться им по электронной почте автоматически.

3.6. Статическая статистика

Модуль статистики облегчает генерацию статической статистики. Для каждой статической статистики создается отдельный файл, в котором точно определено его содержание.

Таким образом может быть сгенерирована очень сложная статистика. Главным недостатком является то, что этот способ есть не очень гибким.

Файлы хранятся в директории Kernel/System/Stats/Static/.

3.7. Использование устаревших данных статической статистики

Прежние версии OTRS 1.3 и 2.0 уже позволяли генерировать отчеты. Различные отчеты для OTRS версии 1.3 и 2.0 которые были специально разработаны для удовлетворения потребностей клиентов могут быть использованы в более поздних версиях OTRS.

Файлы должны быть просто перемещены из Kernel/System/Stats/ в Kernel/System/Stats/Static/. Кроме этого имя пакета соответствующего сценария должен быть заменен на "::Static".

В следующем примере показано как изменяется первый путь.

```
package Kernel::System::Stats::AccountedTime;
```

```
package Kernel::System::Stats::Static::AccountedTime;
```

3.8. Статистика по умолчанию

"Не всегда нужно изобретать велосипед..."

Модуль статистики предоставляет различные предустановленные отчеты. Предустановленные отчеты, которые могут заинтересовать всех OTRS пользователей можно в будущем добавить в модуль статистики. Предустановленные отчеты хранятся в модуле статистики в xml-формате в директории scripts/test/sample/.

Глава 20. Generic Interface

The OTRS Generic Interface consists of a multiple layer framework that lets OTRS communicate with other systems via a web service. This communication could be in two different directions:

- *OTRS as Provider:* OTRS acts as a server listening to requests from the External System, processing the information, performing the requested action, and answering the request.
- *OTRS as Requester:* OTRS acts as a client collecting information, sending the request to the Remote System, and waiting for the response.

1. Generic Interface Layers

Generic Interface is build based on a layer model, to be flexible and easy to customize.

A layer is a set of files, which control how the Generic Interface performs different parts of a web service. Using the right configuration one can build different web services for different External Systems without creating new modules.

Примечание

If the Remote System does not support the current bundled modules of the Generic Interface, special modules need to be developed for that specific web service.

The list of provided Generic Interface modules shipped with OTRS will be updated and increased over time.

Figure: The graphical interface layers.

1.1. Network Transport

This layer is responsible for the correct communication with the Remote System. It receives requests and generates responses when acting as provider, and generates requests and receives responses when acting as requester.

Provider communication is handled by a new web server handle called "nph-genericinterface.pl".

Requester communication could be initiated during an event triggered by a Generic Interface module or any other OTRS module. This event is caught by the event handler and depending on the configuration the event will be processed directly by the requester object or delegated to the Scheduler (a separated daemon designed to process tasks asynchronously).

1.2. Data Mapping

This layer is responsible for translating data structures between OTRS and the Remote System (data internal and data external layers). Usually Remote Systems have different data structures than OTRS (including different values and names for those values), and here resides the importance of the layer to change the received information into something that OTRS can understand and on the opposite way send the information to each Remote System using their data dictionaries.

Example: "Priority" (OTRS) might be called "Prio" in a remote system and it could be that value "1 Low" (OTRS) should be mapped to "Information" on the remote system.

1.3. Controller

Controllers are collections of similar Operations or Invokers. For example, a Ticket controller might contain several standard ticket operations. Custom controllers can be implemented, for example a "TicketExternalCompany" controller which may contain similar functions as the standard Ticket controller, but with a different data interface, or function names (to adapt to the Remote System function names) or complete different code.

One application for Generic Interface could be to synchronize information with one Remote System that only can talk with another Remote System of the same kind. In this case new controllers needs to be developed and the Operations and Invokers has to emulate the Remote System behavior in such way that the interface that OTRS exposes is similar to the Remote System's interface.

1.4. Operation (OTRS as a provider)

An Operation is a single action that can be performed within OTRS. All operations have the same programming interface, they receive the data into one specific parameter, and return a data structure with a success status, potential error message and returning data.

Normally operations uses the already mapped data (internal) to call core modules and perform actions in OTRS like: Create a Ticket, Update a User, Invalidate a Queue, Send a Notification, etc. An operation has full access to the OTRS API to perform the action.

1.5. Invoker (OTRS as a requester)

An Invoker is an action that OTRS performs against a Remote System. Invokers use the OTRS Core modules to process and collect the needed information to create the request. When the information is ready it has to be mapped to the Remote System format in order to be sent to the Remote System, that will process the information execute the action and send the response back, to either process the success or handle errors.

2. Generic Interface Communication Flow

Generic Interface has a defined flow to perform actions as a provider and as a requester.

This flows are described below:

2.1. OTRS as Provider

2.1.1. Remote Request:

1. HTTP request
 - OTRS receives HTTP request and pass it through the layers.
 - The provider module in in charge to execute and control this actions.
2. Network Transport
 - The network transport module decodes the data payload and separates the operation name from the rest of the data.
 - The operation name and the operation data are returned to the provider.
3. *Data External*
 - Data as sent from the remote system (This is not a module-based layer).
4. Mapping
 - The data is transformed from the External System format to the OTRS internal format as specified in the mapping configuration for this operation (Mapping for incoming request data).

- The already transformed data is returned to the provider.

5. *Data Internal*

- Data as transformed and prepared to be passed to the operation (This is not a module based layer).

6. Operation

- Receives and validates data.
- Performs user access control.
- Executes the action.

2.1.2. OTRS Response:

1. Operation

- Returns result data to the provider.

2. *Data Internal*

- Data as returned from operation.

3. Mapping

- The data is transformed back to the Remote system format as specified in the mapping configuration (Mapping for outgoing response data).
- The already transformed data is returned to the provider.

4. *Data external*

- Data as transformed and prepared to be passed to Network Transport as response.

5. Network Transport

- Receives the data already in the Remote System format.
- Constructs a valid response for this network transport type.

6. HTTP response

- The response is sent back to the web service client.
- In the case of an error, an error response is sent to the remote system (e.g. SOAP fault, HTTP error, etc).

2.2. OTRS as Requester

2.2.1. OTRS Request:

1. Event Trigger Handler

- Based on the web service configuration determines if the request will be synchronous or asynchronous.
 - Synchronous
 - A direct call to the Requester is made in order to create a new request and pass it through the layers.
 - Asynchronous

- Create a new Generic Interface (Requester) task for the OTRS Scheduler (by delegating the request execution to the Scheduler, the user experience could be highly improved, otherwise all time needed to prepare the request and the remote execution will be added to the OTRS Events that trigger those requests).
- In its next cycle the Scheduler process reads the new task and creates a call to the Requester that will create a new request and pass it through the layers.

2. Invoker

- Receives data from the event.
- Validates received data (if needed).
- Call core modules to complement the data (if needed).
- Return the request data structure or send a Stop Communication signal to the requester, to gracefully cancel the request.

3. *Data Internal*

- Data as passed from the invoker (This is not a module based layer).

4. Mapping

- The data is transformed to the Remote system format as specified in the mapping configuration (Mapping for outgoing response data).
- The already transformed data is returned to the requester.

5. *Data External*

- Data as transformed and prepared for sending to the remote system.

6. Network Transport

- Receives the remote operation name and the data already transformed to the Remote System format from the requester.
- Constructs a valid request for the network transport.
- Sends the request to the remote system and waits for the response

2.2.2. Remote Response:

1. Network transport

- Receives the response and decodes the data payload.
- Returns the data to the requester.

2. *Data External*

- Data as received from the Remote System

3. Mapping

- The data is transformed form the External System format to the OTRS internal format as specified in the mapping configuration for this operation (Mapping for incoming response data).
- The already transformed data is returned to the requester.

4. Data Internal

- Data as transformed and ready to be passed back to the requester.

5. Invoker

- Receives return data.
- Handles the data as needed by specifically by each Invoker (included error handling if any).
- Return the Invoker result and data to the Requester.

6. Event Handler or Scheduler

- Receives the data from the Requester, in the case of the Scheduler this data might contain information to Re-Schedule the task immediately or in the future.

3. Web Services

A Web Service is a communication method between two systems, in our case OTRS and a Remote System.

The heart of the Web Service is its configuration, where is defined what actions the web service can perform internally (Operation), what can actions the OTRS request can perform Remote System (Invokers), how data is converted from one system to the other (Mapping), and over which protocol the communication will take place (Transport)

The Generic Interface is the framework that makes it possible to create Web Services for OTRS in a pre-defined way, using already made building blocks that are independent from each other and interchangeable.

4. Web Service Graphical Interface

The web service graphical user interface (GUI) is a tool that allows to construct complex web service configurations in a friendly and nice interface. It allows to:

- Create and Delete web services.
- Import and Export configurations (in YAML file format) for existing web services.
- View, Revert and Export old configurations for existing web services in the Web Service History screen.
- Track all communication logs for each web service in the Debugger screen.

4.1. Web Service Overview

The "Web Services" link in the main screen of Admin Interface (in the System Administration box) leads to the web services overview screen, where you are able to manage your web service configurations. You can add new web services or change the configuration of the existing ones from this screen.

Every web service configuration screen has in the upper part of the screen a navigation path in a "bread crumbs" style. This navigation path is useful to know exactly in which part of the web service configuration we are, and also we can jump back to any level of the configuration at any time (this action will not save any changes).

Примечание

To create a new web service, press the button "Add web service", and provide the needed information.

Figure: Web services overview.

4.2. Web Service Add

The only required field in this part is the web service "Name" that needs to be unique in the system and non empty. Other fields are also necessary for the configuration like the "Debug Threshold" and "Validity" but these fields are already filled with the default value for each list.

The default value for "Debug Threshold" is "debug", under this configuration all communication logs are registered in the database, each Debug Threshold value is more restrictive and discard communication logs set for lower values.

Debug Threshold levels (from lower to upper)

- Debug
- Info
- Notice
- Error

It is also possible to define the network transport protocol for "OTRS as Provider" and "OTRS as requester".

Click on the "Save" button to register the new web service in the database or click "Cancel" to discard this operation. You will now be returned to the web service overview screen.

If you already have a web service configuration file in YAML format you can click on the "Import web service" button on the left side of the screen. For more information on importing web services please check the next section "Web Service Change".

Примечание

To change or add more details to a web service, click on the web service name in the web service overview screen.

Figure: Web services add.

4.3. Web Service Change

On this screen you have a complete set of functions to handle every part of a web service. On the left side in the action column you can find some buttons that allows you to perform all possible actions on a web service:

- Clone web service.
- Export web service.
- Import web service.
- Configuration History.
- Delete web service.
- Debugger.

Примечание

"Configuration history" and "Debugger" will lead you to different screens.

4.3.1. Web Service Clone

To clone a web service, you need to click on the "Clone web service" button, a dialog will be shown where you can use the default name or set a new name for the (cloned) web service.

Примечание

Remember the name of the web service must be unique within the system.

Click on "Clone" button to create the web service clone or "Cancel" to close the dialog.

Figure: Web service clone.

4.3.2. Web Service Export

The "Export web service" button gives you the opportunity to dump the configuration of the current web service into a YAML file, download it and store it on your file system. This can be specially useful if you want to migrate the web service from one server to another, for example from a testing environment to a production system.

Предупреждение

All stored passwords in the web service configuration will be exported as plain text.

Right after clicking the "Export web service" button a save dialog of your browser will appear, just like when you click on a file download link on a web page.

Примечание

Each browser on each operating system has its own save dialog screen and style, depending on the browser and its configuration it is possible that no dialog is shown and the file is saved to a default directory on your file system. Please check your browser documentation for more specific instructions if needed.

Figure: Web services export.

4.3.3. Web Service Import

A valid web service configuration YAML file is required to use the import web service feature. Click on the "Import web service" button, browse for the configuration file or provide the complete path in the input box.

Click "Import" button to create a new web service from a file or "Cancel" to close the dialog.

Примечание

The web service name will be taken from the configuration file name (e.g. if the file name is MyWebservice.yml the resulting web service will be named MyWebservice). If a web service is registered in the system with the same name as the web service that you want to import, the system will lead you to the web service change screen to let you change the name of the imported web service.

Figure: Web services import.

4.3.4. Web Service History

Every change to the web service configuration creates a new entry in the web service history (as a journal). The web service history screen displays a list of all configuration versions for a web service. Each row (version) in the "Configuration History List" represents a single revision in the web service history.

Click on one of the rows to show the whole configuration as it was on that particular date / time. The configuration will be shown in the "History details" section of this screen. Here you are also able to export the selected web service configuration version or to restore that version into the current web service configuration.

The "Export web service configuration" behaves exactly as the "Export web service" feature in the web service change screen. For more information refer to that section.

If changes to the current web service configuration does not work as expected and it is not easy to revert the changes manually, you can click on the "Revert web service configuration" button. This will open a dialog to ask you if you are sure to revert the web service configuration. Click "Revert web service configuration" in this dialog to replace the current configuration with the selected version, or click "Cancel" to close the dialog.

Предупреждение

Remember that any passwords stored in the web service configuration will be exported as plain text.

Please be careful when you restore a configuration because this can't be undone.

Figure: Web service history.

4.3.5. Web Service Delete

Sometimes it is necessary to delete a web service completely. To do this you can press on the "Delete web service" button and a new dialog will appear asking for confirmation.

Click on "Delete" to confirm the removal of the web service or on "Cancel" to close the dialog.

Предупреждение

Delete a web service can't be undone, please be careful when deleting a web service.

Figure: Web service delete.

4.3.6. Web Service Debugger

The Debugger stores the log of a web service. In the debugger screen you can track all the web service communications for either provider or requester types.

When this screen is shown the request list starts to load. After the list is fully filled you can choose one of the rows (that means a communication sequence) to check its details. This details will appear in a box below.

You can narrow the communication list using the filter on the right part of the screen. You can filter by:

- Communication type (provider or requester)

- Date: before and / or after a particular date
- The remote IP Address
- A combination of all.

After filter settings are set, push the "Refresh" button and a new list will be displayed meeting your search criteria.

Примечание

Depending on the search criteria for the filters the new list could return no results.

On the left part of the screen under the action column you can select "Go back to the web service" or clear the debugger log by pushing the "Clear" button. This will open a dialog that ask you to confirm erasing of the log. Click "Clear" in the dialog button to perform the action or click on "Cancel" to close this dialog.

In the "Request details" section you can see all the details for the selected communication. Here you can track the complete flow and check for possible errors or confirm success responses.

Figure: Web service debugger.

4.3.7. Web Service Configuration Change

Returning to the web service change screen, now we are going to review the right side of it. Here we have the possibility to modify all the general data for a web service such as name, description, debug threshold, etc. Also there are two more sections below that allows us to modify specific parameters for communication types "OTRS as Provider" and "OTRS as Requester".

The web service configuration needs to be saved on each level. This means that if a setting is changed, links to other, deeper parts of the configuration will be disabled forcing to save the current configuration level. After saving the disabled links will be re-enabled again allowing you to continue with the configuration.

On the "OTRS as provider" section it is possible to set or configure the network transport protocol. Only network transport backends that are registered are shown on the list. To configure the network transport click on the "Configure" button. It is also possible to add new operations in this box. To do this select one of the available operations from the "Add Operation" list. This will lead you to the operation configuration screen. After saving the new operation it will be listed in the table above.

"OTRS as requester" is very similar to the previous one, but instead of "operations" you can add invokers here.

Click the "Save" button to save and continue configuring the web service, "Save and finish" to save and return to the web service overview screen, or "Cancel" to discard current configuration level changes and return to web service overview screen.

Figure: Web services change.

Примечание

Like the other Generic Interface configuration screens such as Network Transport, operation, Invoker and Mapping, the initial configuration (add) screen will only present two options: "Save" and "Cancel", when the configuration is re-visited then

a new option "Save and Finish" will appear. The behavior of this feature is defined below.

"Save" will store the current configuration level in the database and it will return to the same screen to review your changes or to configure deeper settings.

"Save and Finish" will store the current configuration level in the database and it will return to the previous screen in the configuration hierarchy (to the immediate upper configuration level).

"Cancel" will discard any configuration change to the current configuration level and will return to the previous screen in the configuration hierarchy.

4.3.7.1. Web Service Provider Network Transport

In future the list of available network transports will be increased. Currently only the "HTTP::SOAP" transport is available. Each transport has different configuration options to setup and they might use different frontend modules to configure it, but mostly they should look similar to the "HTTP::SOAP" transport configuration module.

For "HTTP::SOAP" protocol as provider the configuration is quite simple. There are only two settings: "Namespace" and "Maximum message length". These fields are required. The first one is a URI to give SOAP methods a context, reducing ambiguities, and the second one it's a field where you can specify the maximum size (in bytes) for SOAP messages that OTRS will process.

Figure: Web service provider network transport.

4.3.7.2. Web Service Operation

The actions that can be performed when you are using OTRS as a provider are called "Operations". Each operation belongs to a controller. Controllers are collections of operations or invokers, normally operations from the same controller need similar settings and shares the same configuration dialog. But each operation can have independent configuration dialogs if needed.

Name, Description, Backend, and Mappings are fields that normally appear on every operation, other special fields can appear in non default configuration dialogs to fulfill specific needs of the operation.

Normally there are two mapping configuration sections on each operation, one for the incoming data and another one for the outgoing data. You can choose different mapping types (backends) for each mapping direction, since their configuration is independent from each other and also independent from the operation backend. The normal and most common practice is that the operation uses same mapping type in both cases (with inverted configuration). The complete mapping configuration is done in a separate screen which depends on the mapping type.

The operation backend is pre-filled and is not editable. You will see this parameter when you choose the operation on the web service edit screen. The field is only informative.

In the left part of the screen on the action column you have the options: "Go back to web service" (discarding all changes since the last save) and "Delete". If you click on the last one, a dialog will open and ask you if you like to remove the operation. Click on "Delete" button to confirm the removal of the operation and its configuration or "Cancel" to close the delete dialog.

Figure: Web service operation.

4.3.7.3. Web Service Provider Transport

The network transport configuration for the requester is similar to the configuration for the provider. For the Requester "HTTP::SOAP" network transport there are more fields to be set.

Apart from the "Endpoint" (URI of the Remote System web service interface to accept requests) and "Namespace" which are required fields, you can also specify:

- Encoding (such as utf-8, latin1, iso-8859-1, cp1250, etc) for the SOAP message.
- SOAPAction Header: you can use this to send an empty or filled SOAPAction header. Set to "No" and the SOAPAction header on the SOAP message will be an empty string, or set to "Yes" to send the soap action in Namespace#Action format and define the separator (typically "/" for .Net web services and "#" for the rest).
- Authentication: to set the authentication mechanism, set to "-" to not use any authentication or select one from the list and the detail fields will appear.

Примечание

Currently only the "BasicAuth" (HTTP) authentication mechanism is implemented. You can decide whether or not to use it depending on the Remote System configuration. If used, you must provide the User Name and the Password to access the remote system.

Предупреждение

If you supply a password for authentication and after you export the web service to a YAML file this password will be revealed and will be written into a plain text string inside the YAML file. Be aware of it and take precautions if needed.

Figure: Web service requester network transport.

4.3.7.4. Web Service Invoker

The actions that can be performed when you are using OTRS as a requester are called "Invokers". Each invoker belongs to a controller (controllers are collections of operations or invokers), normally invokers from the same controller need similar settings and share the same configuration dialogs. Each invoker can have independent configuration dialogs if needed.

Name, Description, Backend, and Mappings are fields that normally appear on every invoker, as well as the list of event triggers other special fields can appear on non default configuration dialogs to fulfill special needs of the invoker.

Normally there are two mapping configuration sections for each invoker, one for the incoming data and another one for the outgoing data. You can choose different mapping types (backends) for each mapping direction, since their configuration is independent from each other and also independent from the invoker backend. The normal and most common practice is that the invoker uses the same mapping type in both cases, with inverted configuration. The complete mapping configuration is done in a separate screen, which depends on the mapping type.

The invoker backend is pre-filled and is not editable. You will see this parameter when you choose the invoker on the web service edit screen. The field is only informative.

Event triggers are events within OTRS such as "TicketCreate", "ArticleSend", etc. These can act as triggers to execute the invoker. Each invoker needs to have at least one event

trigger registered, or the invoker will be useless, because it will never be called. The asynchronous property of the event triggers define if the OTRS process will handle the invoker or if it will be delegated to the Scheduler.

Примечание

The OTRS Scheduler is a separated process that executes tasks in the background. Using this the OTRS process itself will not be affected if the Remote System takes a long time to respond, if it is not available or if there are network problems. If you don't use the scheduler using web services can make OTRS slow or non-responsive. Therefore it is highly recommend to use asynchronous event triggers as often as possible.

To add an Event trigger first select the event family from the first list, then the event name from the second list, then set the asynchronous property (if unchecked means that the event trigger will not be asynchronous) and then click on the plus button. A new event trigger will be created and it will be listed on the invoker "Event Triggers" list.

To delete an Event trigger, simply locate the event trigger to be deleted in the "Event Triggers" list and click on the trash icon at the end of the row. This will open a dialog that ask you if you are sure to delete the event trigger. Click "Delete" to remove the event trigger from the list, or "Cancel" to close the dialog.

In the left part of the screen on the action column you have the options: "Go back to web service" (discarding all changes since the last save) and "Delete". If you click on the last one, a dialog will emerge and ask you if you like to remove the invoker. Click on the "Delete" button to confirm the removal of the invoker and its configuration or "Cancel" to close the delete dialog.

Figure: Web service invoker.

4.3.7.5. Web Service Mapping

There are cases where you need to transform the data from one format to another (map or change data structure), because normally a web service is used to interact with a Remote System, that is highly probable that is not another OTRS system and / or could not understand the OTRS data structures and values. In these cases some or all values has to be changed, and sometimes even the names of the values (keys) or sometimes the complete structure, in order to match with the expected data on the other end. To accomplish this task the the Generic Interface Mapping Layer exists.

Each Remote System has it own data structures and it is possible to create new mapping modules for each case (e.g. there is a customized mapping module for SAP Solution Manager shipped with OTRS), but it is not always necessary. The module Mapping::Simple should cover most of the mapping needs.

Примечание

When Mapping::Simple does not cover all mapping needs for a web service a new mapping module should be created. To learn more about how to create new mapping modules please consult the OTRS Development Manual.

This module gives you the opportunity to set default values to map for each key or value for the whole communication data.

At the beginning of the screen you will see a general section where you can set the default rules that will apply for all the unmapped keys and values. there are three options available, these options are listed below:

- Keep (leave unchanged): doesn't touch the keys or values in any way.
- Ignore (drop key/value pair): when this is applied to the key it deletes the key and value, because when a key is deleted then in consequence its associated value is deleted too. When this is applied to the value, only the value is deleted, keeping the key, that now will be associated to an empty value.
- MapTo (use provided key or value as default): all keys and / or values without a defined map rule, will use this as default, when you select this option a new text field will appear to set this default.

Clicking on the "+" button for new key map, will display a new box for a single mapping configuration. You can add as many key mappings as needed. Just click on the "+" button again and a new mapping box will appear below the existing one. From this mapping boxes you can define a map for a single key, with the next options:

- Exact value(s): the old key string will be changed to a new one if the old key matches exactly.
- Regular expression: The key string will be replaced following a regular expression rule.

Pressing the new value map "+" button will display a new row for a value map. Here also is possible to define rules for each value to be mapped with the same options as for the key map (Exact value and Regular expression). You can add as many values to map as needed, and if you want to delete one of them, just click on the "-" button for each mapping value row.

Deleting the complete key mapping section (box) is possible, just push on the "-" button located on the up right corner of each box that you want to delete.

If you need to delete a complete mapping configuration: go back to the corresponding operation or invoker screen, look for the mapping direction that you select before and set its value to "-", and save the configuration to apply changes.

Figure: Web service mapping.

5. Web Service Command Line Interface

The Command Line Interface (CLI) is fast way to work with the web services. It consists of a set of tools can be use to perform basic operations like:

- Create, Update, Read, List and Delete web services based on YAML files.
- Read the Debugger log, with filter options.

Примечание

You don't need to use the CLI to work with web services. Integrated into the Admin interface there is a complete set of screens to interact with every part of the web services. Please read the web service GUI section included in this manual.

5.1. Web Service Configuration

The "WebserviceConfig.pl" was developed in order to create basic, but fast and powerful tool to work with web service configurations. It gives you the ability to perform the following actions:

- Add: to create web services using a YAML file as the configuration source.
- Update: to change an existing web service, the configuration can be changed using a different or modified YAML file.

- **Read:** to get the current web service configuration displayed on the screen.
- **List:** to get a complete list of all the web services registered in system.
- **Delete:** to delete a web service from the system. Be careful when you use it, because this action can't be undone.

Предупреждение

A web service READ operation will display all the configuration as plain text on the screen, including any stored passwords. Please be aware of this and take the needed precautions!

Example: Creating a new web service configuration:

```
shell> OTRS_HOME/bin/otrs.WebserviceConfig.pl -a write -n <webservice_name>
-f /path/to/yaml/file
```

Also you can use 'otrs.WebserviceConfig.pl' with following options:

- **-a read -i <webservice_id>** - To read a stored configuration.
- **-a write -n <webservice_name> -f /path/to/yaml/file** - To create a new web service.
- **-a write -i <webservice_id> -f /path/to/yaml/file** - To update a web service.
- **-a list** - To list available web services.
- **-a delete -i <webservice_id>** - To delete a web service.

5.2. Web Service Debugger

Another available tool on the command line is the "otrs.GenericInterfaceDebugRead.pl" script, which is an interface to search for web service debugger log entries.

Example: Searching for debugger log entries:

```
shell> bin/otrs.GenericInterfaceDebugRead.pl
```

Optional parameters can be used for the "otrs.GenericInterfaceDebugRead.pl" script:

- **-c** - to filter by Communication ID (md5sum format).
- **-t** - to filter by CommunicationType ('Provider' or 'Requester').
- **-a** - to filter by date (At or After a date).
- **-b** - to filter by date (At or Before a date).
- **-i** - to filter by IP Address (must be valid IPv4 or IPv6 address).
- **-w** - to filter by Web Service ID.
- **-d** - to include detailed communication data.

Example: Searching for debugger log entries with all parameters:

```
shell> ./otrs.GenericInterfaceDebugRead.pl -c
a7cc4d9f5c70387a9bfbe1351bc88966 -t Provider -a '2011-07-22 00:00:00' -b '2011-07-26
00:00:00' -i 127.0.0.1 -w 123 -d 1
```

Примечание

It is highly recommended to include at least one of the filter options listed above, and even more if the "-d" option is selected, because *a lot of* information can be retrieved from the data base and displayed on the screen, this could result in slow response times and much more information than what you really needed.

6. Web Service Configuration

From its design the web services were conceived to be portable from one OTRS system to another, e.g. from a test or development environment to a production system. Therefore it was needed to have an easy way to extract the web service configuration from the database, and import it to another. To accomplish this task Generic Interface uses YAML files as the web services configuration basis.

Why YAML? YAML is a markup language designed to be human friendly to read and write (it is easier to understand than JSON), it does not have some of the limitations of XML like numeric tags, it is open, standardized, and is complete enough to store the whole web service configuration.

Примечание

To learn more about YAML please visit <http://www.yaml.org/>.

The following is a web service configuration file example in YAML format:

```
---
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

```

6.1. Configuration Details

6.1.1. General

- Description: a short text that describes the web service.
- RemoteSystem: a short description of the Remote System.
- Debugger: a container for the debugger settings.
- Provider: a container for the provider settings.
- Requester: a container for the requester settings.

6.1.2. Debugger

- DebugThreshold: the debugger level

Possible Values

- debug: all logs are stored in the database.
- info: info, notice and error level logs are stored in the database.
- notice: notice and error level logs are stored in the database.

- error: only error level logs are stored in the database.

6.1.3. Provider

- Operation: a container for each operation settings.
- Transport: a container for provider network transport settings.

6.1.3.1. Operation

- <OperationName>: Unique name for the operation, container for its own operation settings (cardinality 0..n, but not duplicate).

6.1.3.1.1. <OperationName>

This section is based on operations from type "Test::Test" other operations might contain more or different settings.

- Description: a short text that describes the operation.
- MappingInbound: a container for the mapping settings for the incoming request data.
- MappingOutbound: a container for the mapping settings for the outgoing response data.
- Type: the operation backend, in Controller::Operation format.

6.1.3.1.1.1. MappingInbound

This section is based on mappings from type "Simple". Other mappings might contain more or different settings.

- Config: a container for this mapping settings.
- Type: the mapping backend.

6.1.3.1.1.1.1. Config

- KeyMapDefault: a container for all non mapped keys settings.
- ValueMapDefault: a container for all non mapped values settings.
- KeyMapExact: a container for all exact key mappings (cardinality 0 .. 1).
- KeyMapRegex: a container for all regular expression key mappings (cardinality 0 .. 1).
- ValueMap: a container for all value mappings (cardinality 0 .. 1).

6.1.3.1.1.1.1.1. KeyMapDefault

- MapTo: the new value to be used (only applicable if MapType is set to MapTo).
- MapType: the rule for the mapping.

Possible Values

- Keep: leave unchanged.
- Ignore: drop.
- MapTo: change to the MapTo value.

6.1.3.1.1.1.1.2. ValueMapDefault

Similar to KeyMapDefault.

6.1.3.1.1.1.1.3. KeyMapExact

- <oldkey>: <newkey> (cardinality 0 .. n but not duplicate).

6.1.3.1.1.1.1.4. KeyMapRegEx

- <oldkey(RegEx)>: <newkey> (cardinality 0 .. n but no duplicates).

6.1.3.1.1.1.1.5. ValueMap

- <newkey>: a container for value mappings for this new key (cardinality depends on the new keys from KeyMapExact and KeyMapRegEx).

6.1.3.1.1.1.1.5.1. <newkey>

- ValueMapExact: a container for all exact value mappings (cardinality 0 .. 1).
- ValueMapRegEx: a container for all regular expression value mappings (cardinality 0 .. 1).

6.1.3.1.1.1.1.5.1.1. valueMapExact

- <oldvalue>: <newvalue> (cardinality 0 .. n but not duplicate).

6.1.3.1.1.1.1.5.1.2. ValueMapRegEx

- <oldvalue(RegEx)>: <newvalue> (cardinality 0 .. n but not duplicate).

6.1.3.1.1.2. MappingOutbound

Same as MappingInbound.

6.1.3.1.1.3. Transport

This section is based on the provider network transport HTTP::SOAP, other transports might contain more or different settings.

- Config: a container for the specific network transport configuration settings.
- Type: the provider network transport backend.

6.1.3.1.1.3.1. Config

- MaxLength: the maximum length in bytes to be read in a SOAP message by OTRS.
- NameSpace: an URI that gives a context to all operations that belongs to this web service.

6.1.4. Requester

- Invoker: a container for each invokers' settings.
- Transport: a container for requester network transport settings.

6.1.4.1. Invoker

- <InvokerName>: Unique name for the invoker, container for its own invoker settings (cardinality 0..n, but not duplicate).

6.1.4.1.1. <InvokerName>

This section is based on invokers from type "Test::Test" other invokers might contain more or different settings.

- Description: a short text that describes the invoker
- Events: a container for a unnamed list of event trigger settings.
- MappingInbound: a container for the mapping settings for the incoming response data.
- MappingOutbound: a container for the mapping settings for the outgoing request data.
- Type: the invoker backend, in Controller::Invoker format.

6.1.4.1.1.1. Events

- *List Element*: (cardinality 0 .. n)
 - Asynchronous: to set if the invoker execution will be delegated to the Scheduler

Possible Values

- 0: not handled by the Scheduler.
- 1: handled by the Scheduler.
- Event: the name of the event trigger.

Possible Values (for ticket events)

- TicketCreate
- TicketDelete
- TicketTitleUpdate
- TicketUnlockTimeoutUpdate
- TicketQueueUpdate
- TicketTypeUpdate
- TicketServiceUpdate
- TicketSLAUpdate
- TicketCustomerUpdate
- TicketFreeTextUpdate
- TicketFreeTimeUpdate
- TicketPendingTimeUpdate
- TicketLockUpdate
- TicketArchiveFlagUpdate
- TicketStateUpdate
- TicketOwnerUpdate
- TicketResponsibleUpdate
- TicketPriorityUpdate
- HistoryAdd

- HistoryDelete
- TicketAccountTime
- TicketMerge
- TicketSubscribe
- TicketUnsubscribe
- TicketFlagSet
- TicketFlagDelete
- TicketSlaveLinkAdd
- TicketSlaveLinkDelete
- TicketMasterLinkDelete

Possible Values (for article events)

- Article Events
- ArticleCreate
- ArticleFreeTextUpdate
- ArticleUpdate
- ArticleSend
- ArticleBounce
- ArticleAgentNotification
- ArticleCustomerNotification
- ArticleAutoResponse
- ArticleFlagSet
- ArticleFlagDelete
- ArticleAgentNotification
- ArticleCustomerNotification

6.1.4.1.1.2. MappingInbound

Same as Operation MappingInbound

6.1.4.1.1.3. MappingOutbound

Same as Operation MappingInbound.

6.1.4.1.1.4. Transport

This section is based on the requester network transport HTTP::SOAP, other transports might contain more or different settings.

- Config: a container for the specific network transport configuration settings.

- Type: the requester network transport backend.

6.1.4.1.1.4.1. Config

- Authentication: a container for authentication settings.
- Encoding: the SOAP Message request encoding
- Endpoint: the URI of the Remote Server web service to accept OTRS requests
- NameSpace: an URI that gives a context to all invokers that belongs to this web service.
- SOAPAction: to send an empty or filled SOAPAction header in the SOAP Message (in "<NameSpace> <Separator> <Action>" format).

Possible Values

- YES: to send a filled SOAPAction header.
- No: to send an empty SOAPAction header.
- SOAPActionSeparator: to set the <Separator> of a filled SOAPAction header.

Possible Values

- '/': used for .net web services.
- '#': used for all the rest web services.

6.1.4.1.1.4.1.1. Authentication

- User: the privileged user name that has access to the remote web service.
- Password: the password for privileged user in plain text.
- Type: the type of authentication.

7. Connectors

A Connector is in essence set of actions called Operations if OTRS acts as a web service provider or Invokers if OTRS acts as a web service requester. But it can also include special Mappings or Transports

One Connector can have only Operations, Only Invokers or both. A connector can even use parts of other connectors like the Mappings or Transports if they are not so specific for the Connector that implements them.

In another words a Connector is not limited to just the Controller layer but it can be extended to Data Mapping or Network Transport layers if needed.

Due to the modular design of the Generic Interface a Connector can be seen as a plug-in; this means that by adding Connectors the capabilities of the generic interface can be extended using: OTRS Feature add ons, OTRS Custom modules, 3rd Party modules, and so on.

7.1. Bundled Connectors

Included with this version of OTRS the following connectors are ready to be used.

- Session
- Ticket

7.1.1. Session Connector

This connector is capable to create a valid SessionID that can be used in any other operation.

Provides:

- Operations:
 - SessionCreate

7.1.1.1. Operations

7.1.1.1.1. SessionCreate

Creates a new new valid SessionID to be used in other operations from other connectors like TicketCreate.

Примечание

To use the SessionID in other operations from other connectors is necessary that the operation implements authentication by SessionID. all the rest of the bundled operations are capable to accept a valid SessionID as an authentication method.

Possible Attributes:

```
<SessionCreate>
  <!--You have a MANDATORY CHOICE of the next 2 items at this level-->
  <!--Optional:-->
  <UserLogin?></UserLogin>
  <!--Optional:-->
  <CustomerUserLogin?></CustomerUserLogin>
  <!--Optional:-->
  <Password?></Password>
</SessionCreate>
```

7.1.2. Ticket Connector

This connector supplies the basic functionality to interact with tickets

Provides:

- Operations:
 - TicketCreate
 - TicketUpdate
 - TicketGet
 - TicketSearch

7.1.2.1. Operations

7.1.2.1.1. TicketCreate

Provides an interface to create ticket in OTRS, a ticket must contain an Article and can contain several attachments, all defined Dynamic Fields can be also set on TicketCreate operation.

Possible Attributes:

```

<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:-->
    <PendingTime>
      <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>

```

7.1.2.1.2. TicketUpdate

TicketUpdate operation add the capability to modify attributes from a ticket or add a new article, including attachments and all defined dynamic fields for the ticket and the new article.

Примечание

It is not necessary to create a new article to modify a ticket attribute.

Possible Attributes:

```

<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:-->
  <PendingTime>
    <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>

```

7.1.2.1.3. TicketGet

This operation is used to get all the attributes of a ticket including the dynamic fields, all the articles and all the attachments that belongs to the ticket.

Possible Attributes:

```

<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>
<!--Optional:-->
<TicketID>?</TicketID>
<!--Optional:-->
<DynamicFields>?</DynamicFields>
<!--Optional:-->
<Extended>?</Extended>
<!--Optional:-->
<AllArticles>?</AllArticles>
<!--Optional:-->
<ArticleSenderType>?</ArticleSenderType>
<!--Optional:-->
<ArticleOrder>?</ArticleOrder>
<!--Optional:-->

```

```

<ArticleLimit>?</ArticleLimit>
<!--Optional:-->
<Attachments>?</Attachments>
</TicketGet>

```

7.1.2.1.4. TicketSearch

TicketSearch operation returns a list of Ticket IDs that matches a predefined criteria.

Possible Attributes:

```

<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:-->
  <TypeID>?</TypeID>
  <!--Zero or more repetitions:-->
  <States>?</States>
  <!--Zero or more repetitions:-->
  <StateIDs>?</StateIDs>
  <!--Zero or more repetitions:-->
  <StateType>?</StateType>
  <!--Zero or more repetitions:-->
  <StateTypeID>?</StateTypeID>
  <!--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:-->
<CreatedTypeID>?</CreatedTypeID>
<!--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:-->
<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:-->
<TicketEscalationTimeNewerMinutes?></TicketEscalationTimeNewerMinutes>
<!--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>

```

7.2. Examples:

7.2.1. Web Service Configuration

The following is a basic but complete web service configuration file in YAML format to use all the Ticket Connector operations, in order to use it in OTRS you need to copy the content, save it into a file called GenericTicketConnector.yml, and import it into OTRS in the Web Services screen in the Admin panel by clicking in the "Add web service" action from the overview screen and then clicking in the "Import web service" action in the add screen.

```

---
Debugger:
  DebugThreshold: debug
  TestMode: 0
Description: Ticket Connector Sample
FrameworkVersion: 3.1.x CVS
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: Retrieve 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: ''

```

7.2.2. Perl SOAP Requester

The following code is a Perl script that can connect to OTRS via the generic interface, to perform the operations provided by the Ticket Connector, it uses two Perl CPAN modules SOAP::Lite and Data::Dumper, be sure that your environment is capable to use that modules before you try to run the script.

```

#!/usr/bin/perl -w
# --
# otrs.SOAPRequest.pl - sample to send a SOAP request to OTRS Generic Interface Ticket
# Connector
# Copyright (C) 2001-2013 OTRS AG, http://otrs.org/
# --
# $Id: genericinterface-connectors.xml,v 1.8.2.1 2013-01-10 15:33:57 ub Exp $
# --
# This program is free software; you can redistribute it and/or modify
# it under the terms of the GNU AFFERO General Public License as published by
# the Free Software Foundation; either version 3 of the License, or
# 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 Affero General Public License
# along with this program; if not, write to the Free Software
# Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
# or see http://www.gnu.org/licenses/agpl.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 OTRS Admin Manual in order to get the complete list.
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} );  
}  
}
```



Глава 21. OTRS Scheduler

The OTRS Scheduler is an independent system process that executes tasks in background. These kind of processes are known as *daemons* in Unix / Linux systems or as *services* on Windows environments. It is independent but that doesn't mean that the Scheduler does everything alone, it is fully integrated into OTRS and can use any OTRS module as needed to complete each task.

Currently the OTRS Scheduler is only able to handle Generic Interface tasks. These kind of tasks executes invokers that send requests to remote systems. Other handlers for different tasks will be added in future OTRS versions.

For sanity reasons the Scheduler process needs to be restarted from time to time. This is done automatically by the scheduler process itself once a day, but it can be adjusted as needed using the SysConfig by editing the "Scheduler::RestartAfterSeconds" setting.

The OTRS Scheduler is a set it and forget it process, the only needed human interaction is to check its status periodically and start or stop it as needed.

Примечание

If the Scheduler is stopped for any reason all pending tasks and new tasks registered when the Scheduler is stopped will be executed as soon as the Scheduler starts again (unless the tasks are set to be executed in the future).

1. Scheduler Graphical Interface

The Scheduler is not visible in the OTRS Graphical User Interface unless it stops running.

1.1. Scheduler Not Running Notification

There are two different types of notifications if the system detects that scheduler is not running. This detection is based on the Scheduler process update frequency, if the difference between current time and the last process update time is 2 times the process update frequency a warning message will be displayed in the OTRS notification area. If it is over 4 times the process frequency then an alert will be displayed instead.

The Scheduler process update time can be configured via the SysConfig in the "Scheduler::PIDUpdateTime" setting.

If you would see a warning message it is not always necessary to take an action, but is highly recommended to check if the scheduler process is running. If you see an alert, then is highly probable that the scheduler is in fact not running and should be started.

By default the Scheduler not running notification is enabled, if there is a valid web service registered in the database, and is only displayed to the users in the "admin" group.

To disable the notification (not recommended) or to change or add the notification groups, please edit the "Frontend::NotifyModule###800-Scheduler-Check" setting in the SysConfig.

Figure: Scheduler notification.

1.2. Start Scheduler

By clicking on the Scheduler not running notification link (either warning or alert) a dialog box will open to let you start the Scheduler process again. The Scheduler can be started normally or forced to start, by clicking on the appropriate check box in the dialog.

Примечание

A forced Scheduler start is only necessary if previous Scheduler process was terminated abnormally and the Process ID is still registered in the database.

To have full control of the Scheduler process and to check it real status please use the command line tools described below.

Figure: Start Scheduler.

2. Scheduler Command Line Interface

The Scheduler command line tools let you control the Scheduler process (Start / Stop) or query it status. There are also tools to register the process to be controlled by the operating system.

Included with OTRS there are two set of CLI tools, one for Unix / Linux OS and another for MS Windows OS.

2.1. Unix / Linux

2.1.1. Scheduler Init.d Files

Init.d files are special scripts that are called by the operating system at startup and shutdown (or restart) times.

OTRS provide init.d scripts to start / stop the OTRS Scheduler process automatically by the operating system, this scripts are located under OTRS_HOME/scripts.

Init.d scripts needs to be copied to the correct location for your operating system, they need to had the proper permissions and some internal variables needs to be set to work properly.

Init.d Script Internal Variables

- **OTRS_HOME** - the path of your OTRS installation.
- **User** - the apache process user name.
- **Group** - the apache process user's group name.

Примечание

Currently there are only init.d scripts for Linux platforms.

Таблица 21.1. List of Init Scripts And Supported Operating Systems

Init Script	Supported OS
otrs-scheduler-linux	Red Hat, Fedora, CentOS, SUSE, openSUSE, Debian, Ubuntu
otrs-scheduler-gentoo-init.d, scheduler-gentoo-conf.d	otrs- Gentoo

Пример 21.1. Example To Start The OTRS Scheduler Form An Init.d Script

```
shell> /etc/init.d/otrs-scheduler-linux start
```

Available Actions

- **start** to start the OTRS Scheduler process.
- **stop** to stop the OTRS Scheduler process.
- **restart** to restart the OTRS Scheduler process.
- **status** to query the OTRS Scheduler process status.

The Scheduler needs the database to be available to register its Process ID, for this reason is necessary to:

- Execute the Scheduler init.d script to *start* the Scheduler process after the database process is up and running.
- Execute the Scheduler init.d script to *stop* the Scheduler before the database process shuts down.

Примечание

If you want the Scheduler to run at system startup, please read the documentation of the operating system for the right location to place the init.d scripts, how to configure them to run automatically and how to set the run order.

2.1.2. Scheduler Daemon File

This is the part of the Scheduler that stays running in the background checking for tasks to execute. It also provides the main functions to control the process.

All Unix / Linux uses the file **OTRS_HOME/bin/otrs.Scheduler.pl**.

Пример 21.2. Example To Start The OTRS Scheduler

```
shell> OTRS_HOME/bin/otrs.Scheduler.pl -a start
```

Available Options

- **-a** action.

Possible Values

- **start-** to start the Scheduler process.
- **stop-** to stop the Scheduler process.
- **status-** to query Scheduler process status.
- **-f** to force the start or stop of the Scheduler process.

Пример 21.3. Example To Force Stop The OTRS Scheduler

```
shell> OTRS_HOME/bin/otrs.Scheduler.pl -a stop -f 1
```

Примечание

Force stop the Scheduler is used to remove the process ID from the database when scheduler is not running and the process is still registered.

Force start the Scheduler is used to start the Scheduler process if the scheduler is not running and the process is registered.

Force start or stop are only necessary if the start of the process is needed to be done before the process update time expires. Otherwise an expired entry in the database is discarded by normal start.

2.2. Windows

2.2.1. Scheduler Service Installer

The integration of the services into the MS Windows Operating System is done via the Windows Service Control Manager (SCM). In order to make the OTRS Scheduler process to be controlled by the SCM is necessary to register this service

OTRS provides the script **OTRS_HOME/bin/otrs.Scheduler4WinInstaller.pl** to register or unregister the OTRS Scheduler into the SCM.

Пример 21.4. Example To Register The OTRS Scheduler Into the Windows SCM

```
shell> OTRS_HOME/bin/otrs.Scheduler4WinInstaller.pl -a install
```

Available Options

- **-a** action.

Possible Values

- **install**- to install the Scheduler process into the Windows SCM.
- **remove**- to remove the Scheduler process from the Windows SCM.

After installing into the Windows SCM the OTRS Scheduler process can be used as any other service in Windows. It can be started, stopped and restarted and can be configured to be started manually or automatic.

Примечание

To learn more about Windows Services and the Windows SCM please read the Windows documentation, and Microsoft online help.

2.2.2. Scheduler Service File

This is the part of the Scheduler that stays running in the background checking for tasks to execute. It also provides the main functions to control the process.

Windows Operating System uses the file **OTRS_HOME/bin/otrs.Scheduler4Win.pl**.

Пример 21.5. Example To Start The OTRS Scheduler

```
shell> OTRS_HOME/bin/otrs.Scheduler4Win.pl -a start
```

Available Options

- **-a** action.

Possible Values

- **start**- to start the Scheduler process.
- **stop**- to stop the Scheduler process.
- **status**- to query Scheduler process status.
- **-f** to force the start or stop of the Scheduler process.

Пример 21.6. Example To Force Stop The OTRS Scheduler

```
shell> OTRS_HOME/bin/otrs.Scheduler4Win.pl -a stop -f 1
```

Примечание

Force stopping the Scheduler is used to remove the process ID from the database when scheduler is not running and the process is still registered.

Force starting the Scheduler is used to start the Scheduler process if the scheduler is not running and the process is still registered.

Force start or stop are only necessary if starting the process is needed to be done before the process update time expires. Otherwise an expired entry in the database would be discarded by a normal start.

Глава 22. Dynamic Fields

1. Introduction

A dynamic field is a special kind of field in OTRS, created to extend the information stored on a ticket or article. These fields are not fixed in the system and they can appear only in specific screens, they can be mandatory or not, and their representation in the screens depends on the field type defined at their creation time according to the data to be held by the field. For example, there are fields to hold a text, a date, a selection of items, etc.

Dynamic fields are the evolution of TicketFreeText TicketFreeKey TicketFreeTime, ArticleFreeText and ArticleFreeKey fields that were commonly used in OTRS 3.0 and before. The limitation of these "Free Fields" was that they can be defined up to 16 (text or dropdown) fields and 6 time fields for a ticket and 3 (text or dropdown) fields for each article only, not more.

Now with dynamic fields the limitation in the number of fields per ticket or article is removed, you can create as many dynamic fields you like either for ticket or articles. And beyond that, the framework behind the dynamic fields is prepared to handle custom fields for other objects rather than just ticket and articles.

This new framework that handles the dynamic fields is built using a modular approach, where each kind of dynamic field can be seen as a plug-in module for the framework. This means that the variety of dynamic fields can be easily extended by public OTRS modules, OTRS Feature Add-ons, OTRS custom developments, and other custom developments.

The following dynamic field types are included with this release:

- Text (one line of text)
- Textarea (multiple lines of text)
- Checkbox
- Dropdown (single choice, multiple values)
- Multiselect (multiple choice, multiple values)
- Date
- Date / Time

2. Configuration

By default, a clean installation of OTRS 3.1.x does not include any dynamic fields. If you plan to use such fields in tickets or articles you need to create dynamic fields.

An updated installation from OTRS 3.0.x will have all the old "free fields" created as dynamic fields for compatibility and data preservation. The following is the list of dynamic fields that are created during the migration from OTRS 3.0.x to 3.1.x.

- TicketFreeKey[1-16] (TicketFreeKey1, TicketFreeKey2 ... TicketFreeKey16)
- TicketFreeText[1-16]
- TicketFreeTime[1-6]
- ArticleFreeKey[1-16]
- ArticleFreeText[1-16]

Примечание

During the migration procedure from OTRS 3.0.x to OTRS 3.1.x all the old "free fields" data and configuration are migrated to the new dynamic fields architecture. Any custom development around the old "free fields" has to be updated to use the new dynamic field framework.

The migration of the configuration include the field itself and the screen configurations to hide, show or show field as mandatory for each screen.

The configuration of a dynamic field is split in two parts, to add a new dynamic field or manage an existing one you need to navigate into the "Admin" panel in the "Dynamic Fields" link. To show, show as mandatory or hide a dynamic field in one screen you need to change the OTRS settings in the "SysConfig" screen.

2.1. Adding a Dynamic Field

Click in the "Admin" button located in the navigation bar, then click on the "Dynamic Field" link inside "Ticket Settings" box located in the lower center of the screen. The dynamic fields overview will display as follows:

Figure: Dynamic fields overview screen, empty.

Notice that this screen will change as you add more dynamic fields to list all created dynamic fields. This screen might already have some fields if the installation was updated from an older version of OTRS.

The Actions in the side bar at the left of the screen describes two possibilities: Article and Ticket, each one has it's own dropdown selection of dynamic fields.

Примечание

The installation of an OTRS package could add more objects to the Action side bar.

The general procedure to create a dynamic field is:

- Click on the desired dynamic field object dropdown in the Action side bar.
- Click on the dynamic field type that you want to add from the list.
- Fill the configuration.
- Save.

The configuration dialogs for the dynamic fields are split in two parts, the upper section is common among all the fields and the lower part might be different from one type of dynamic field to another.

General dynamic field settings:

- Name: Mandatory, unique, only letters and numbers are allowed.

This is the internal name of the field, used for example to show or hide a field in one screen. Any modification on field name (not recommended) will need manual a update on the "SysConfig" settings where the field is referenciated.

- Label: Mandatory.

This is field name to be displayed in the screens, it supports translations.

Примечание

Label translations have to be added manually to language translations files.

- Field order: Mandatory.

Defines the relative order in which the field will be displayed in the screen, by default each new field has the last position, a change in this setting will affect the other of the other created dynamic fields.

- Validity: Mandatory.

An invalid dynamic field will not be displayed in any screen, no matter if is configured to displayed.

- Field type: Mandatory, Read only.

Shows the current selected field type.

- Object type: Mandatory, Read only.

Shows the scope of field.

Примечание

To illustrate each specific field type settings a few fields will be added in our example. These new fields will be referenced in later sections.

For the following examples all the dynamic fields will be created for the Ticket object if you need to create a dynamic field for Article object, just chose the field from the Article dropdown list.

Таблица 22.1. The following fields will be added into the system:

Название	Label	Type
Field1	My Field 1	Text
Field2	My Field 2	Textarea
Field3	My Field 3	Checkbox
Field4	My Field 4	Dropdown
Field5	My Field 5	Multiselect
Field6	My Field 5	Date
Field7	My Field 6	Date / Time

2.2. Text Dynamic Field Configuration

Text dynamic field is used to store a single line string.

Text dynamic field settings:

- Default value: Optional.

This is the value to be shown by default in the edit screens (like New Phone Ticket or Ticket Compose).

- Show link: Optional.

If set, the field value will be converted into a clickable link for display screens (like ticket zoom or overviews).

For example, if "Show link" is set to "http://www.otrs.com", clicking on the filled value will make your browser to open the OTRS web page.

Примечание

The use of `$LQData{"NameX"}` in the Set link value, where NameX is the name of the field will add the field value as part of the link reference.

Figure: Dynamic field Text configuration dialog.

2.3. Textarea Dynamic Field Configuration

Textarea dynamic field is used to store a multiple line string.

Textarea dynamic field settings:

- Number of rows: Optional, integer.

Used to define the height of the field in the edit screens (like New Phone Ticket or Ticket Compose).

- Number of cols: Optional, Integer.

This is value is used to define the width of the field in the edit screens.

- Default value: Optional.

This is the value to be shown by default in the edit screens (it can be a multiple line text).

Figure: Dynamic field Textarea configuration dialog.

2.4. Checkbox Dynamic Field Configuration

Checkbox dynamic field is used to store true or false value, represented by a checked or unchecked check box.

Checkbox dynamic field settings:

- Default value: Mandatory.

This is the value to be shown by default in the edit screens (like New Phone Ticket or Ticket Compose), the default value for this field is closed selection that can be Checked or Unchecked.

Figure: Dynamic field Checkbox configuration dialog.

2.5. Dropdown Dynamic Field Configuration

Dropdown dynamic field is used to store a single value, from a closed list.

Dropdown dynamic field settings:

- Possible values: Mandatory.

List of values to choose. when add a new value is necessary to specify the Key (internal value) and the Value (display value).

- Default value: Optional.

This is the value to be show by default in the edit screens (like New Phone Ticket or Ticket Compose), the default value for this field is closed selection defined by the Possible values.

- Add empty value: Mandatory, (Yes / No).

If this option is activated an extra value is defined to show a "-" in the list of possible values, this special value is empty internally.

- Translatable values: Mandatory, (Yes / No).

This setting is used mark the possible values of this field to be translated. Only the display values are translated, internal values are not affected, the translation of the values needs to be manually added to the language files.

- Show link: Optional.

If set, the field value will be converted into a clickable HTP link for display screens (like Zoom or overviews).

For example, if Show link is set to "<http://www.otrs.com>", clicking on the filed value will make your browser to open the OTRS web page.

Примечание

The use of `$LQData{"NameX"}` in the Set link value, where NameX is the name of the field will add the field value as part of the link reference.

Figure: Dynamic field Dropdown configuration dialog.

2.6. Multiselect Dynamic Field Configuration

Multiselect dynamic field is used to store a multiple values, from a closed list.

Multiselect dynamic field settings:

- Possible values: Mandatory.

List of values to choose. when add a new value is necessary to specify the Key (internal value) and the Value (display value).

- Default value: Optional.

This is the value to be show by default in the edit screens (like New Phone Ticket or Ticket Compose), the default value for this field is closed selection defined by the Possible values.

- Add empty value: Mandatory, (Yes / No).

If this option is activated an extra value is defined to show a "-" in the list of possible values, this special value is empty internally.

- Translatable values: Mandatory, (Yes / No).

This setting is used mark the possible values of this field to be translated. Only the display values are translated, internal values are not affected, the translation of the values needs to be manually added to the language files.

Figure: Dynamic field Multiselect configuration dialog.

2.7. Date Dynamic Field Configuration

Date dynamic field is used to store a date value (Day, Month and Year).

Date dynamic field settings:

- Default date difference: Optional, Integer.

Number of seconds (positive or negative) between the current date and the selected date to be show by default in the edit screens (like New Phone Ticket or Ticket Compose).

- Define years period: Mandatory (Yes / No).

Used to set a defined number of years in past and future from current date in the year select of this field, If set to Yes the following options are available:

- Years in the past: Optional, Positive integer.

Define the number of years in past from current day to display in the year selection for this dined in edit screens.

- Years in the future: Optional, Positive integer.

Define the number of years in future from current day to display in the year selection for this dined in edit screens.

- Show link: Optional.

If set, the field value will be converted into a clickable HTP link for display screens (like Zoom or overviews).

For example, if Show link is set to "<http://www.otrs.com>", clicking on the filed value will make your browser to open the OTRS web page.

Примечание

The use of `$LQData{"NameX"}` in the Set link value, where NameX is the name of the field will add the field value as part of the link reference.

Figure: Dynamic field Date configuration dialog.

2.8. Date / Time Dynamic Field Configuration

Date / Time dynamic field is used to store a date time value (Minute, Hour, Day, Month and Year).

Date / Time dynamic field settings:

- Default date difference: Optional, Integer.

Number of seconds (positive or negative) between the current date and the selected date to be show by default in the edit screens (like New Phone Ticket or Ticket Compose).

- Define years period: Mandatory (Yes / No).

Used to set a defined number of years in past and future from current date in the year select of this field, If set to Yes the following options are available:

- Years in the past: Optional, Positive integer.

Define the number of years in past from current day to display in the year selection for this field in edit screens.

- Years in the future: Optional, Positive integer.

Define the number of years in future from current day to display in the year selection for this field in edit screens.

- Show link: Optional.

If set, the field value will be converted into a clickable HTTP link for display screens (like Zoom or overviews).

For example, if Show link is set to "http://www.otrs.com", clicking on the field value will make your browser to open the OTRS web page.

Примечание

The use of `$LQData{"NameX"}` in the Set link value, where NameX is the name of the field will add the field value as part of the link reference.

Figure: Dynamic field Date / Time configuration dialog.

2.9. Editing a Dynamic Field

A filled dynamic field overview screen (with the previous examples) should look like:

Figure: Dynamic field overview screen filled with sample data.

To change or edit a dynamic field you must have at least one field defined, select an already added field from the dynamic fields overview screen and update its settings.

Примечание

Not all the dynamic field settings can be changed, the Field type and Object type are fixed from the selection of the field and they can't be changed.

It is not recommended to change the field internal name, but the label can be changed at any time. If internal name is changed all "SysConfig" settings that has a reference to that particular field needs to be updated as well as user preferences (if defined).

2.10. Showing a Dynamic Field on a Screen

To display a dynamic field on a particular screen there are two mandatory conditions:

1. The dynamic field must be valid.
2. The dynamic field must be set to 1 or 2 in the configuration of the screen.

Follow these steps to show a dynamic field in a screen

- Be sure that the dynamic field is set to valid, you can see the validity of the field from the dynamic field overview screen. Set to valid by editing the field if necessary.
- Open the "sysconfig" and select "Ticket" from the dropdown list in the Actions side bar located in the left part of the screen.

Примечание

You can also search for "DynamicField" in the search box above or the "sysconfig" key directly if you already know it.

- Locate the setting sub-group for the screen that you are looking for and click on it. For example "Frontend::Agent::Ticket::ViewPhoneNew".
- Search for the setting that ends with "###DynamicField". For example "Ticket::Frontend::AgentTicketPhone###DynamicField".
- If the setting is empty or does not have the required dynamic field name, click on the "+" button to add a new entry. For example Key: Field1, Content: 1.

If the setting already has the dynamic field name listed be sure that is set to "1" to display the field or to "2" to display it as mandatory.

- Save the configuration by clicking in the "Update" button and the bottom of the screen and navigate to the screen where you want the field to be displayed.

2.10.1. Show Examples

The following are "sysconfig" configurations examples to show or hide dynamic fields on different screens.

Пример 22.1. Activate Field1 in New Phone Ticket Screen.

- *Group:* Ticket
- *Sub-group:* Frontend::Agent::Ticket::ViewPhoneNew
- *Setting:* Ticket::Frontend::AgentTicketPhone###DynamicField
- *Value:*

Key	Content
Field1	1

Figure: Field1 in New Phone Ticket Screen.

Пример 22.2. Activate Field1 in New Phone Ticket Screen as mandatory.

- *Group:* Ticket
- *Sub-group:* Frontend::Agent::Ticket::ViewPhoneNew
- *Setting:* Ticket::Frontend::AgentTicketPhone###DynamicField
- *Value:*

Key	Content
Field1	2

Figure: Field1 in New Phone Ticket Screen as mandatory.

Пример 22.3. Activate several fields in New Phone Ticket Screen.

- *Group:* Ticket
- *Sub-group:* Frontend::Agent::Ticket::ViewPhoneNew
- *Setting:* Ticket::Frontend::AgentTicketPhone###DynamicField
- *Value:*

Key	Content
Field1	1
Field2	1
Field3	1
Field4	1
Field5	1
Field6	1
Field7	1

Figure: Several fields in New Phone Ticket Screen as mandatory.

Пример 22.4. Deactivate some fields in New Phone Ticket Screen.

- *Group:* Ticket
- *Sub-group:* Frontend::Agent::Ticket::ViewPhoneNew
- *Setting:* Ticket::Frontend::AgentTicketPhone###DynamicField
- *Value:*

Key	Content
Field1	1
Field2	0
Field3	1
Field4	0
Field5	1
Field6	0
Field7	1

Figure: Some deactivated fields in New Phone Ticket Screen as mandatory.

Пример 22.5. Activate Field1 in Ticket Zoom Screen.

- *Group:* Ticket
- *Sub-group:* Frontend::Agent::Ticket::ViewZoom
- *Setting:* Ticket::Frontend::AgentTicketZoom###DynamicField
- *Value:*

Key	Content
Field1	1

Figure: Field1 in Ticket Zoom Screen.

Пример 22.6. Activate Field1 in Ticket Overview Small Screens.

- *Group:* Ticket
- *Sub-group:* Frontend::Agent::TicketOverview
- *Setting:* Ticket::Frontend::OverviewSmall###DynamicField
- *Value:*

Key	Content
Field1	1

Figure: Field1 in Ticket Overview Small Screen.

This setting affects: Escalation View, Locked View, Queue View, Responsible View, Status View and Watch View screens.

2.11. Setting a Default Value by a Ticket Event Module

A ticket event (e.g. TicketCreate) can trigger a value set for a certain field, if the field does not have a value yet.

Примечание

By using this method this default value, is not seen in the edit screen (e.g. New Phone Ticket) since the value is set after the creation of the ticket.

To activate this feature is necessary to enable the following setting: "Ticket::EventModulePost###TicketDynamicFieldDefault".

Пример 22.7. Activate Field1 in TicketCreate event.

- *Group:* Ticket
- *Sub-group:* Core::TicketDynamicFieldDefault
- *Setting:* Ticket::TicketDynamicFieldDefault###Element1

Примечание

This configuration can be set in any of the 16 Ticket::TicketDynamicFieldDefault###Element settings.

If more that 16 fields needs to be set up a custom XML file must be places in \$OTRS_HOME/Kernel/Config/files directory to extend this feature.

- *Value:*

Key	Content
Event	TicketCreate
Название	Field1
Value	a new value

2.12. Set a Default Value by User Preferences

The dynamic field default value can be overwritten with a user defined value stored in the user preferences.

Using this method, the default value of the field will be shown on any screen where the field is activated (if the field does not have already a different value).

The "sysconfig" setting "PreferencesGroups###DynamicField" located in the "Frontend::Agent::Preferences" Sub-group. This setting is an example of how to create an entry in the User Preferences screen to set an exclusive dynamic field default value for the selected user. The limitation of this setting is that it only admits one dynamic field. if two or more fields will use this feature is necessary to create a custom XML configuration file to add more settings similar to this one.

Примечание

Remember, if more settings are added in a new XML each setting name needs to be unique in the system and different than "PreferencesGroups###DynamicField". for example:
 PreferencesGroups###101-DynamicField-Field1, PreferencesGroups###102-DynamicField-Field2,
 PreferencesGroups###My-Field1, PreferencesGroups###My-Field2, etc.

Пример 22.8. Activate Field1 in the User preferences.

- *Group:* Ticket
- *Sub-group:* Frontend::Agent::Preferences
- *Setting:* PreferencesGroups###101-DynamicField-Field1
- *Value:*

Key	Content
Event	TicketCreate
Active	1
Block	Input
Column	Другие настройки
Data:	<code>\$Env{"UserDynamicField_Field1"}</code>
Key:	My Field 1
Label:	Default value for: My Field 1
Module:	Kernel::Output::HTML::PreferencesGeneric
PrefKey:	UserDynamicField_Field1
Prio:	7000

Figure: Field1 in User preferences screen.

2.13. Updating from OTRS 3.0

During the update from OTRS 3.0 there is a special step that is very important and must not be skipped, there is a script file named "DBUpdate-to-3.1.pl" under the "\$OTRS_HOME/scripts" that must be called at the precise time in the update procedure (please refer to UPGRADING file for more details).

The script mentioned above is in charge to get all the "Free fields" configurations and creates their dynamic field substitute, it also gathers each "Free field" data from every ticket and article and move it to the dynamic fields format. It also reads the screen configuration for "Free fields" and port it to the dynamic fields format.

Примечание

All the "Free fields" configurations are kept in the system but they are not used anymore, they are placed under the group "Z_Deprecated", and they might be removed in further versions of OTRS.

While the updating script tries to be very precise migrating the screen configurations it is always good to do a manual fine tuning. Please note that "Free fields" TicketFreeKey[1-16] and TicketFreeText[1-16] came always in pair in OTRS 3.0 systems (and before). This is no longer necessary it could be that the TicketFreeKey field is not necessary anymore for a particular TicketFreeText field.

The dynamic fields overview screen from clean OTRS 3.0.x updated to 3.1.x should look like:

Figure: Updated dynamic field overview screen page2.

Figure: Updated dynamic field overview screen page2.

Please note that the new dynamic field named "TicketFreeText2" has the label "Product", this is because the default configuration from this "Free field" in OTRS 3.0.x specifies that "TicketFreeKey2" has a fixed value of "Product". In this case the dynamic field "TicketFreeKey2" is not needed anymore and it can be manually set to invalid.

It might be also a very good idea to set the label value of each valid migrated field to a text that describes better the use of the field in the system.

Глава 23. Дополнительные приложения

Для расширения функциональности OTRS-фреймворка, можно также установить дополнительные пакеты приложений. Это можно сделать с помощью менеджера пакетов из Панели Администрирования, который загружает приложения из онлайн репозитория и управляет зависимостями между этими пакетами. Также приложения можно установить из локальных файлов.

1. FAQ (Часто Задаваемые Вопросы)

FAQ - это компонент Базы Знаний. Он позволяет редактировать и просматривать статьи. Статьи можно просматривать и ограничивать для агентов, клиентов и пользователей или анонимных пользователей. Также они могут быть структурированы по группам и быть написаны на различных языках.



Глава 24. Настройка Производительности

Ниже представлен перечень различных техник, которые можно использовать для получения максимально возможной производительности системы OTRS: настройка, кодирование, использование памяти и многое другое.

1. OTRS

Есть несколько возможных вариантов улучшения производительности OTRS.

1.1. TicketIndexModule

Есть два типа хранилища информации для хранения индексов заявок:

- `Kernel::System::Ticket::IndexAccelerator::RuntimeDB` (по умолчанию), "на лету" генерирует из таблицы заявок шаблон просмотра для каждой очереди. Пока в системе не больше 60, 000 открытых заявок, проблем с производительностью не будет.
- `Kernel::System::Ticket::IndexAccelerator::StaticDB` - один из самых мощных модулей, его следует использовать, когда у вас есть больше 80.000 открытых заявок. Он использует дополнительную таблицу `ticket_index`, которая работает как обычный шаблон. Используйте `bin/otrs.RebuildTicketIndex.pl` для получения начального индекса после создания хранилища информации.

Вы можете изменить `IndexAccelerator` через `SysConfig`.

1.2. TicketStorageModule

Существует два различных хранилища для хранения заявок/статей:

- Настройте `Kernel::System::Ticket::ArticleStorageDB` (по умолчанию) для хранения вложений, и т.д. в базе данных. Примечание: Не используйте его для больших наборов данных.

Pro: Если пользователь вашего веб-сервера не является пользователем 'otrs', используйте этот модуль, чтобы избежать проблем с правами доступа.

Противопоказания: Не совсем целесообразно хранить вложения в базе данных. Будьте осторожны, при хранении больших объектов. Настройте конфигурационный параметр MySQL `"set-variable=max_allowed_packet=8M"` для хранения объектов размером 8 MB (по умолчанию это 2M).

- Configure `Kernel::System::Ticket::ArticleStorageFS` для хранения вложений и т.д. в локальной файловой системе. Примечание: Рекомендовано при больших установках.

Pro: Это быстро!

Противопоказания: Пользователь веб-сервера должен быть пользователем 'otrs'. Кроме того, если у вас есть несколько серверов, нужно убедиться в том, что файловая система разделяется между этими серверами. Так что разместите его на общедоступный NFS-ресурс или SAN, или же используйте аналогичные решения.

Примечание: можно переключаться с одного бэк-енда на другой "на лету". Бэк-энды можно переключить в `SysConfig` и затем запустить в командной строке утилиту `otrs.ArticleStorageSwitch.pl`, чтобы внести статьи из базы данных в файловую систему и наоборот. Можно использовать опции `-s` и `-d` для указания источника и конечный бэк-энд. Обратите внимание, что весь процесс может занять много времени, в зависимости от количества статей, которые у вас есть и доступной мощности процессора и/или пропускной способности сети.

```
shell> bin/otrs.ArticleStorageSwitch.pl -s ArticleStorageDB -d ArticleStorageFS
```

Script: Switching storage back-ends from database to filesystem.

1.3. Архивирование Заявок

Поскольку OTRS может использоваться в качестве системы аудита доказательств, то удаление закрытых заявок не очень хорошая идея. Именно по этому мы реализовали функцию архивирования заявок.

Заявки, которые удовлетворяют определенным критериям, могут быть отмечены как "архивированные". При обычном поиске заявок или при поиске заявок с помощью задания Generic Agent, они будут недоступны. Сама собой система не имеет дела с огромным количеством заявок, до тех пор, пока не принимается во внимание "последняя" заявка. Это может означать огромный прирост производительности на больших системах.

Для использования функции архивации выполните следующие действия:

1. Включение архивирования системы в SysConfig

В Панели Администрирования перейдите в SysConfig и выберите группу Заявка. В Core::Ticket найдите опцию Ticket::ArchiveSystem, по умолчанию установленную в значение "нет". Измените значение этой настройки на "да" и сохраните изменения.

2. Определение работы GenericAgent

В Панели Администратора выберите GenericAgent и добавьте новое задание (работу).

a. Настройки Задания

Введите имя для работы архивирования, и выберите надлежащие опции для планирования этой работы.

b. Ticket Filter

Фильтр заявок производит поиск заявок, которые отвечают выбранным критериям. Хорошей идеей будет заархивировать закрытые заявки, которые были закрыты за несколько месяцев до этого.

c. Действия над заявками

В этой части установите поле "Архивировать выбранные заявки" в "архив заявок".

d. Сохранить работу

В конце страницы у вас будет возможность сохранить работу (задание).

e. Обработанные заявки

Система отобразит все заявки, которые будут заархивированы при выполнении задания Generic Agent-ом.

3. Поиск Заявок

При поиске заявок, система по умолчанию производит поиск среди не архивированных заявок. Установите критерий поиска "поиск в архивах", если нужно чтобы поиск происходил также и в архивированных заявках.

2. База данных

Решения зависят от используемой базы данных. Изучайте документацию к используемой базе данных или же обратитесь за помощью к администратору.

2.1. MySQL

Если для MySQL-таблиц используется MyISAM (а по умолчанию так и есть) и из таблицы была удалена большая часть данных, или же производилось множество действий над таблицей, которая содержит строки переменной длины (таблицы, которые содержат поля типов VARCHAR, BLOB или TEXT), то нужно дефрагментировать таблицу, или другими словами выполнить команду "оптимизировать".

You should try this if the mysqld daemon needs a lot of your CPU time. Optimize the tables - ticket, ticket_history and article (see Script below).

```
shell$ mysql -u user -p database
mysql$ optimize table ticket;
mysql$ optimize table ticket_history;
mysql$ optimize table article;
```

Script: Optimizing data base tables.

2.2. PostgreSQL

PostgreSQL лучше настраивается путем изменений в файле postgresql.conf который находится в директории с установленным PostgreSQL. За советом, как это сделать следите в следующих статьях:

- <http://www.revsys.com/writings/postgresql-performance.html>
- <http://varlena.com/GeneralBits/Tidbits/perf.html>
- http://varlena.com/GeneralBits/Tidbits/annotated_conf_e.html

Если производительность остается на неудовлетворительном уровне, мы предлагаем вам присоединиться к спискам рассылки "PostgreSQL Performance" (<http://www.postgresql.org/community/lists/>) и задавать вопросы именно там. Люди в этом списке рассылке очень дружелюбны а всегда постараются вам помочь.

3. Веб-сервер

Конечно же нужно использовать mod_perl 2.0 (<http://perl.apache.org/>). Он быстрее (~ * 100) чем "чистый" cgi. Но он требует больше оперативной памяти (RAM).

3.1. Предустановленное соединение с базой данных

При запуске веб-сервера можно иметь заранее установленное соединение с базой данных. Такой подход существенно экономит время (см. README.webserver).

3.2. Предварительно загруженные модули - startup.pl

Чтобы все работало быстрее и занимало меньше памяти используйте сценарий запуска scripts/apache2-perl-startup.pl для предварительной загрузки/компиляции Perl модулей на сервере с mod_perl (см. README.webserver).

3.3. Перезагрузка Perl-модулей во время обновления с диска

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. Important: this would also mean you can't use the OTRS Package Manager via the web interface, you need to use the command line variant - bin/otrs.PackageManager.pl.

3.4. Выбор Правильной Стратегии

If you have a larger installation, say 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

Используйте mod_deflate для Apache2, если у вас "слабая" полоса пропускания . При загрузке html-страницы размером в 45к, mod_gzip/mod_deflate сожмет ее и страница будет занимать около 7к. Недостатком такого подхода является дополнительная нагрузка на сервер.



Глава 25. Резервное копирование системы

В этой главе описывается резервное копирование и восстановление данных OTRS.

1. Резервное копирование

Есть два типа данных для резервного копирования: файлы приложения (например, файлы в `/opt/otrs`) и данные, хранящиеся в базе данных.

To simplify backups, the script `scripts/backup.pl` is included with every OTRS installation. It can be run to backup all important data (see Script below).

```
linux:/opt/otrs# cd scripts/  
linux:/opt/otrs/scripts# ./backup.pl --help  
backup.pl <Revision 1.1> - backup script  
Copyright (c) 2001-2005 Martin Edenhofer <martin@otrs.org>  
usage: backup.pl -d /data_backup/ [-c bzip2|gzip] [-r 30] [-t nofullbackup]  
linux:/opt/otrs/scripts#
```

Script: Getting help about the OTRS backup mechanism.

Execute the command specified in the script below to create a backup:

```
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#
```

Script: Creating a backup.

All data was stored in the directory `/backup/2010-09-07_14-28/` (see Script below). Additionally, the data was saved into a `.tar.gz` file.

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

Script: Checking the backup files.

2. Восстановление

Для восстановления резервной копии, сохраненные данные приложения должны быть записаны обратно в каталог установки, например `/opt/otrs`. База данных также должна быть восстановлена.

A script `scripts/restore.pl` (see Script below), which simplifies the restore process, is shipped with every OTRS installation. It supports MySQL and PostgreSQL.

```
linux:/opt/otrs/scripts# ./restore.pl --help  
restore.pl <Revision 1.1> - restore script  
Copyright (c) 2001-2005 Martin Edenhofer <martin@otrs.org>  
usage: restore.pl -b /data_backup/<TIME>/ -d /opt/otrs/  
linux:/opt/otrs/scripts#
```

Script: Getting help about the restore mechanism.

Data that is stored, for example, in the directory `/backup/2010-09-07_14-28/`, can be restored with the command specified in the script below, assuming the OTRS installation is at `/opt/otrs`.

```
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#
```

Script: Restoring OTRS data.



Приложение А. Дополнительные ресурсы

Мы стараемся снабжать вас самой последней информацией об OTRS и предоставляем вам возможность оставить нам свой отзыв.

1. Домашняя страница OTRS.org

Домашняя страница нашего проекта находится по адресу <http://www.otrs.org/> (см. ниже Рисунок А-1) .

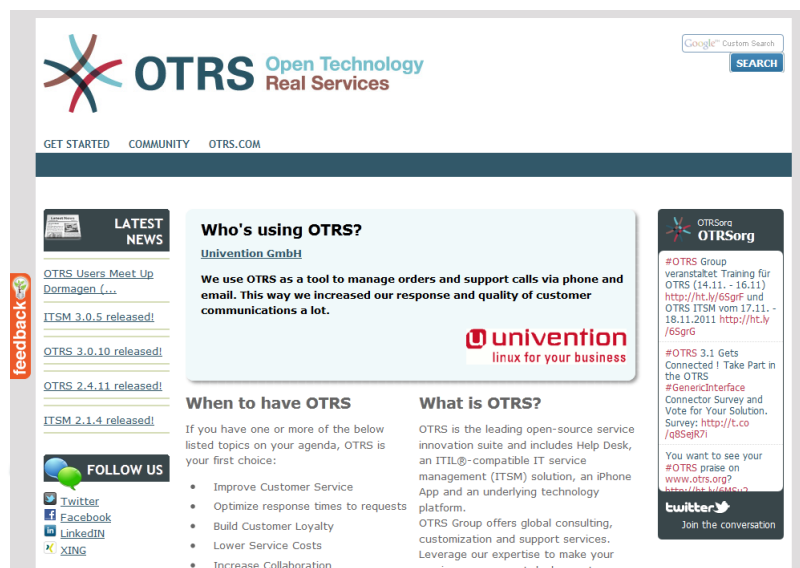


Figure: OTRS Homepage.

2. Списки рассылки

В Таблице А-1 приведен полный список рассылки для сообщества.

Таблица А.1. Списки рассылки

Название	Описание	Домашняя страница
announce@otrs.org	Низкий трафик, на английском языке, для объявлений о новых релизах OTRS и решениях о безопасности.	http://lists.otrs.org/cgi-bin/listinfo/announce
otrs@otrs.org	От среднего до высокого списка трафика, на английском языке, где вы можете найти все виды вопросов, касающихся поддержки и продукта.	http://lists.otrs.org/cgi-bin/listinfo/otrs
otrs-de@otrs.org	От среднего до большого списка трафика, на немецком языке, где вы можете найти все виды вопросов, касающихся поддержки и продукта.	http://lists.otrs.org/cgi-bin/listinfo/otrs-de
dev@otrs.org	Средний трафик на английском языке, где OTRS-	http://lists.otrs.org/cgi-bin/listinfo/dev

Название	Описание	Домашняя страница
	разработчики обсуждают различные планы и вопросы реализации.	
i18n@otrs.org	Небольшое количество пользователей на английском языке с вопросами интернационализации и локализации. Если вы есть или только хотите стать переводчиком проекта OTRS или имеете проблемы с одним из наших приложений, это именно то место, куда вам нужно.	http://lists.otrs.org/cgi-bin/listinfo/i18n

Чтобы подписаться на любой из этих списков рассылки перейдите по ссылке: <http://lists.otrs.org/>.

3. Трекинг ошибок

To submit bugs visit <http://bugs.otrs.org/> (see Figure below). Please take note of the difference between a bug and a configuration issue. Configuration issues are problems that you encounter when setting a system, or general questions on the use of OTRS. Bug reports should only be used for issues with the source code of OTRS itself, or to file enhancements for OTRS. All your bug reports and enhancement requests are very welcome in the bug tracker.

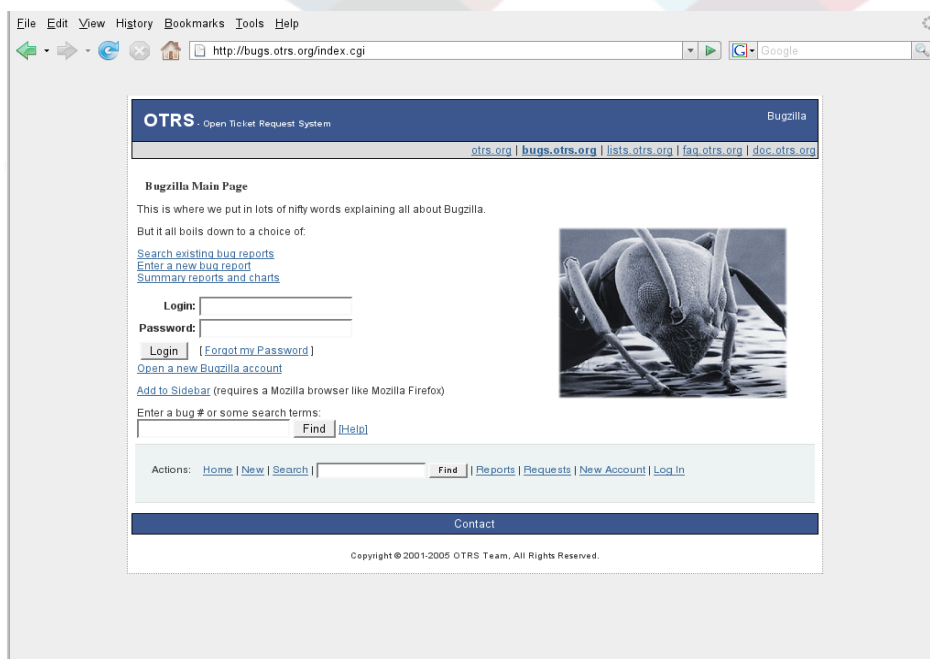


Figure: Bug tracker.

По вопросам конфигурации используйте ссылку [коммерческая поддержка, которая доступна на сайте OTRS.com](#), или же публичные списки рассылки.

Вы поможете нам улучшить продукт если сообщите об ошибке. Мы ценим ваш вклад!

4. Коммерческая поддержка

For services (support, consulting, development, and training) you can contact the company behind OTRS, OTRS AG. They have offices in Germany, USA, Mexico, the Netherlands and other countries. Look at their website for contact information: <http://www.otrs.com/en/corporate-navigation/contact/>



Приложение В. Configuration Options Reference

1. Framework

1.1. Core

1.1.1. SecureMode

Description	Value
Description:	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 (to avoid the use of destructive queries, such as DROP DATABASE, and also to steal user passwords).
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'SecureMode'} = '0';</pre>

1.1.2. Frontend::DebugMode

Description	Value
Description:	Enables or disable the debug mode over frontend interface.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::DebugMode'} = '0';</pre>

1.1.3. ConfigLevel

Description	Value
Description:	Sets the configuration level of the administrator. Depending on the config level, some sysconfig options will be not shown. The config levels are in 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.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<code>\$Self->{'ConfigLevel'} = '100';</code>

1.1.4. ProductName

Description	Value
Description:	Defines the name of the application, shown in the web interface, tabs and title bar of the web browser.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'ProductName'} = 'OTRS';</code>

1.1.5. SystemID

Description	Value
Description:	Defines the system identifier. Every ticket number and http session string contain 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).
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SystemID'} = '10';</code>

1.1.6. FQDN

Description	Value
Description:	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.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'FQDN'} = 'yourhost.example.com';</code>

1.1.7. HttpType

Description	Value
Description:	Defines the type of protocol, used by the web server, to serve the application. If https protocol will be used instead of plain http,

Description	Value
	it must be specified it 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 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.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'HttpType'} = 'http';</pre>

1.1.8. ScriptAlias

Description	Value
Description:	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.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'ScriptAlias'} = 'otrs/';</pre>

1.1.9. AdminEmail

Description	Value
Description:	Defines the system administrator's email address. It will be displayed in the error screens of the application.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'AdminEmail'} = 'admin@example.com';</pre>

1.1.10. Organization

Description	Value
Description:	Company name for the customer web interface. Will also be included in emails as an X-Header.
Group:	Framework
SubGroup:	Core

Description	Value
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Organization'} = 'Example Company';</code>

1.1.11. DefaultLanguage

Description	Value
Description:	Defines the default front-end language. All the possible values are determined by the available language files on the system (see the next setting).
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'DefaultLanguage'} = 'en';</code>

1.1.12. DefaultUsedLanguages

Description	Value
Description:	Defines all the languages that are available to the application. The Key/Content pair links the front-end display name to the appropriate language PM file. The "Key" value should be the base-name of the PM file (i.e. de.pm is the file, then de is the "Key" value). The "Content" value should be the display name for the front-end. Specify any own-defined language here (see the developer documentation http://doc.otrs.org/ for more information). Please remember to use the HTML equivalents for non-ASCII characters (i.e. for the German oe = o umlaut, it is necessary to use the ö symbol).
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'DefaultUsedLanguages'} = { 'ar_SA' => 'Arabic (Saudi Arabia)', 'bg' => 'Bulgarian (&#x0411;&#x044a;&#x043b;&#x0433;&#x0430;&#x0440;&#x0441;&#x043a;&#x0438;)', 'ca' => 'Catal&agrave;', 'cs' => 'Czech (&#x010c;esky)', 'da' => 'Dansk', 'de' => 'Deutsch', 'el' => 'Greek (&#x0395;&#x03bb;&#x03b9;&#x03b7;&#x03bd;&#x03b9;&#x03ba;&#x03ac;)', 'en' => 'English (United States)', 'en_CA' => 'English (Canada)', 'en_GB' => 'English (United Kingdom)', 'es' => 'Espa&ntilde;ol', 'es_CO' => 'Espa&ntilde;ol (Colombia)', 'es_MX' => 'Espa&ntilde;ol (M&eacute;xico)', 'et' => 'Eesti', 'fa' => 'Persian (&#x0641;&#x0627;&#x0631;&#x0633;&#x0649;)', </pre>

Description	Value
	<pre>'fi' => 'Suomi', 'fr' => 'Fran&ccedil;ais', 'hi' => 'Hindi', 'hr' => 'Hrvatski', 'hu' => 'Magyar', 'it' => 'Italiano', 'ja' => 'Japanese (&#x65e5;&#x672c;&#x8a9e)', 'lt' => "Lietuvi\x{173} kalba", 'lv' => 'Latvijas', 'ms' => 'Malay', 'nb_NO' => 'Norsk bokm&aring;l', 'nl' => 'Nederlands', 'pl' => 'Polski', 'pt' => 'Portugu&ecirc;s', 'pt_BR' => 'Portugu&ecirc;s Brasileiro', 'ru' => 'Russian (&#x0420;&#x0443;&#x0441;&#x0441;&#x043a;&#x0438;&#x0439;)', 'sk_SK' => 'Slovak (Sloven&#x010d;ina)', 'sl' => "Slovenian (Sloven\x{161}\x{10d;ina)", 'sr_Cyrl' => "Serbian Cyrillic (\x{441}\x{440}\x{43f}\x{441}\x{43a}\x{438})", 'sr_Latn' => 'Serbian Latin (Srpski)', 'sv' => 'Svenska', 'tr' => 'T&uuml;rk&ccedil;e', 'uk' => 'Ukrainian (&#x0423;&#x043a;&#x0440;&#x0430;&#x0457;&#x043d;&#x0441;&#x044c;&#x043a;&#x0430;)', 'vi_VN' => 'Vietnam (Vi&#x0246;t Nam)', 'zh_CN' => 'Chinese (Sim.) (&#x7b80;&#x4f53;&#x4e2d;&#x6587;)', 'zh_TW' => 'Chinese (Tradi.) (&#x6b63;&#x9ad4;&#x4e2d;&#x6587;)' };</pre>

1.1.13. DefaultTheme

Description	Value
Description:	Defines the default front-end (HTML) theme to be used by the agents and customers. The default themes are Standard and Lite. If you like, you can add your own theme. Please refer the administrator manual located at http://doc.otrs.org/ .
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'DefaultTheme'} = 'Standard';</code>

1.1.14. DefaultTheme::HostBased

Description	Value
Description:	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.
Group:	Framework
SubGroup:	Core
Valid:	0

Description	Value
Required:	0
Config-Setting:	<pre>\$Self->{'DefaultTheme::HostBased'} = { 'host1\\.example\\.com' => 'SomeTheme1', 'host2\\.example\\.com' => 'SomeTheme2' };</pre>

1.1.15. CheckMXRecord

Description	Value
Description:	Makes the application check the MX record of email addresses before sending an email or submitting a telephone or email ticket.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'CheckMXRecord'} = '1';</pre>

1.1.16. CheckMXRecord::Nameserver

Description	Value
Description:	Defines the address of a dedicated DNS server, if necessary, for the "CheckMXRecord" look-ups.
Group:	Framework
SubGroup:	Core
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'CheckMXRecord::Nameserver'} = 'ns.example.com';</pre>

1.1.17. CheckEmailAddresses

Description	Value
Description:	Makes the application check the syntax of email addresses.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'CheckEmailAddresses'} = '1';</pre>

1.1.18. CheckEmailValidAddress

Description	Value
Description:	Defines a regular expression that excludes some addresses from the syntax check (if "CheckEmailAddresses" is set to "Yes").

Description	Value
	Please enter a regex in this field for email addresses, that aren't syntactically valid, but are necessary for the system (i.e. "root@localhost").
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'CheckEmailValidAddress'} = '^(root@localhost admin@localhost)\$';</pre>

1.1.19. CheckEmailInvalidAddress

Description	Value
Description:	Defines a regular expression that filters all email addresses that should not be used in the application.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'CheckEmailInvalidAddress'} = '@(example)\.\.\. ...)\$';</pre>

1.1.20. CGILogPrefix

Description	Value
Description:	Specifies the text that should appear in the log file to denote a CGI script entry.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'CGILogPrefix'} = 'OTRS-CGI';</pre>

1.1.21. DemoSystem

Description	Value
Description:	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.
Group:	Framework
SubGroup:	Core
Valid:	1

Description	Value
Required:	1
Config-Setting:	<code>\$Self->{'DemoSystem'} = '0';</code>

1.1.22. SwitchToUser

Description	Value
Description:	Allows the administrators to login as other users, via the users administration panel.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SwitchToUser'} = '0';</code>

1.1.23. NotificationSenderName

Description	Value
Description:	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 Notification Master" otrs@your.example.com). Notifications are messages such as en::Customer::QueueUpdate or en::Agent::Move.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'NotificationSenderName'} = 'OTRS Notification Master';</code>

1.1.24. NotificationSenderEmail

Description	Value
Description:	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 Notification Master" otrs@your.example.com). You can use the OTRS_CONFIG_FQDN variable as set in your configuration, or choose another email address. Notifications are messages such as en::Customer::QueueUpdate or en::Agent::Move.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'NotificationSenderEmail'} = 'otrs@<OTRS_CONFIG_FQDN>';</code>

1.1.25. System::Customer::Permission

Description	Value
Description:	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.
Group:	Framework
SubGroup:	Core
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'System::Customer::Permission'} = ['ro', 'rw'];</pre>

1.1.26. LanguageDebug

Description	Value
Description:	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".
Group:	Framework
SubGroup:	Core
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'LanguageDebug'} = '0';</pre>

1.1.27. Secure::DisableBanner

Description	Value
Description:	If enabled, the OTRS version tag will be removed from the HTTP headers.
Group:	Framework
SubGroup:	Core
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Secure::DisableBanner'} = '0';</pre>

1.1.28. StandardResponse2QueueByCreating

Description	Value
Description:	List of default StandardResponses which are assigned automatically to new Queues upon creation.

Description	Value
Group:	Framework
SubGroup:	Core
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'StandardResponse2QueueByCreating'} = [''];</pre>

1.2. Core::LinkObject

1.2.1. LinkObject::ViewMode

Description	Value
Description:	Determines the way the linked objects are displayed in each zoom mask.
Group:	Framework
SubGroup:	Core::LinkObject
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'LinkObject::ViewMode'} = 'Simple';</pre>

1.2.2. LinkObject::Type###Normal

Description	Value
Description:	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.
Group:	Framework
SubGroup:	Core::LinkObject
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'LinkObject::Type'}->{'Normal'} = { 'SourceName' => 'Normal', 'TargetName' => 'Normal' };</pre>

1.2.3. LinkObject::Type###ParentChild

Description	Value
Description:	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.
Group:	Framework
SubGroup:	Core::LinkObject
Valid:	1

Description	Value
Required:	1
Config-Setting:	<pre>\$Self->{'LinkObject::Type'}->{'ParentChild'} = { 'SourceName' => 'Parent', 'TargetName' => 'Child' };</pre>

1.2.4. LinkObject::TypeGroup###0001

Description	Value
Description:	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.
Group:	Framework
SubGroup:	Core::LinkObject
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'LinkObject::TypeGroup'}->{'0001'} = ['Normal', 'ParentChild'];</pre>

1.3. Core::Log

1.3.1. LogModule

Description	Value
Description:	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.
Group:	Framework
SubGroup:	Core::Log
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'LogModule'} = 'Kernel::System::Log::SysLog';</pre>

1.3.2. LogModule::SysLog::Facility

Description	Value
Description:	If "SysLog" was selected for LogModule, a special log facility can be specified.
Group:	Framework
SubGroup:	Core::Log
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<code>\$Self->{'LogModule::SysLog::Facility'} = 'user';</code>

1.3.3. LogModule::SysLog::LogSock

Description	Value
Description:	If "SysLog" was selected for LogModule, a special log sock can be specified (on solaris you may need to use 'stream').
Group:	Framework
SubGroup:	Core::Log
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'LogModule::SysLog::LogSock'} = 'unix';</code>

1.3.4. LogModule::SysLog::Charset

Description	Value
Description:	If "SysLog" was selected for LogModule, the charset that should be used for logging can be specified.
Group:	Framework
SubGroup:	Core::Log
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'LogModule::SysLog::Charset'} = 'utf-8';</code>

1.3.5. LogModule::LogFile

Description	Value
Description:	If "file" was selected for LogModule, a logfile must be specified. If the file doesn't exist, it will be created by the system.
Group:	Framework
SubGroup:	Core::Log
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'LogModule::LogFile'} = '/tmp/otrs.log';</code>

1.3.6. LogModule::LogFile::Date

Description	Value
Description:	Adds a suffix with the actual year and month to the OTRS log file. A logfile for every month will be created.
Group:	Framework
SubGroup:	Core::Log
Valid:	1

Description	Value
Required:	1
Config-Setting:	<code>\$Self->{'LogModule::LogFile::Date'} = '0';</code>

1.4. Core::MIME-Viewer

1.4.1. MIME-Viewer###application/excel

Description	Value
Description:	Specifies the path to the converter that allows the view of Microsoft Excel files, in the web interface.
Group:	Framework
SubGroup:	Core::MIME-Viewer
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'MIME-Viewer'}->{'application/excel'} = 'xlhtml';</code>

1.4.2. MIME-Viewer###application/msword

Description	Value
Description:	Specifies the path to the converter that allows the view of Microsoft Word files, in the web interface.
Group:	Framework
SubGroup:	Core::MIME-Viewer
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'MIME-Viewer'}->{'application/msword'} = 'wvWare';</code>

1.4.3. MIME-Viewer###application/pdf

Description	Value
Description:	Specifies the path to the converter that allows the view of PDF documents, in the web interface.
Group:	Framework
SubGroup:	Core::MIME-Viewer
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'MIME-Viewer'}->{'application/pdf'} = 'pdftohtml -stdout -i';</code>

1.4.4. MIME-Viewer###text/xml

Description	Value
Description:	Specifies the path to the converter that allows the view of XML files, in the web interface.

Description	Value
Group:	Framework
SubGroup:	Core::MIME-Viewer
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'MIME-Viewer'}->{'text/xml'} = '<OTRS_CONFIG_Home>/scripts/tools/xml2html.pl';</pre>

1.5. Core::MirrorDB

1.5.1. Core::MirrorDB::DSN

Description	Value
Description:	If you want to use a mirror database for agent ticket fulltext search or to generate stats, specify the DSN to this database.
Group:	Framework
SubGroup:	Core::MirrorDB
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Core::MirrorDB::DSN'} = 'DBI:mysql:database=mirrordb;host=mirrordbhost';</pre>

1.5.2. Core::MirrorDB::User

Description	Value
Description:	If you want to use a mirror database for agent ticket fulltext search or to generate stats, the user to authenticate to this database can be specified.
Group:	Framework
SubGroup:	Core::MirrorDB
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Core::MirrorDB::User'} = 'some_user';</pre>

1.5.3. Core::MirrorDB::Password

Description	Value
Description:	If you want to use a mirror database for agent ticket fulltext search or to generate stats, the password to authenticate to this database can be specified.
Group:	Framework
SubGroup:	Core::MirrorDB
Valid:	0
Required:	0
Config-Setting:	

Description	Value
	<code>\$Self->{'Core::MirrorDB::Password'} = 'some_password';</code>

1.6. Core::PDF

1.6.1. PDF

Description	Value
Description:	Enables PDF output. The CPAN module PDF::API2 is required, if not installed, PDF output will be disabled.
Group:	Framework
SubGroup:	Core::PDF
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PDF'} = '1';</code>

1.6.2. PDF::LogoFile

Description	Value
Description:	Specifies the path of the file for the logo in the page header (gif jpg png, 700 x 100 pixel).
Group:	Framework
SubGroup:	Core::PDF
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PDF::LogoFile'} = '<OTRS_CONFIG_Home>/var/logo-otrs.png';</code>

1.6.3. PDF::PageSize

Description	Value
Description:	Defines the standard size of PDF pages.
Group:	Framework
SubGroup:	Core::PDF
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PDF::PageSize'} = 'a4';</code>

1.6.4. PDF::MaxPages

Description	Value
Description:	Defines the maximum number of pages per PDF file.
Group:	Framework
SubGroup:	Core::PDF
Valid:	1

Description	Value
Required:	1
Config-Setting:	<code>\$Self->{'PDF::MaxPages'} = '100';</code>

1.6.5. PDF::TTFontFile###Proportional

Description	Value
Description:	Defines the path and TTF-File to handle proportional font in PDF documents.
Group:	Framework
SubGroup:	Core::PDF
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PDF::TTFontFile'}->{'Proportional'} = 'DejaVuSans.ttf';</code>

1.6.6. PDF::TTFontFile###ProportionalBold

Description	Value
Description:	Defines the path and TTF-File to handle bold proportional font in PDF documents.
Group:	Framework
SubGroup:	Core::PDF
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PDF::TTFontFile'}->{'ProportionalBold'} = 'DejaVuSans-Bold.ttf';</code>

1.6.7. PDF::TTFontFile###ProportionalItalic

Description	Value
Description:	Defines the path and TTF-File to handle italic proportional font in PDF documents.
Group:	Framework
SubGroup:	Core::PDF
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'PDF::TTFontFile'}->{'ProportionalItalic'} = 'DejaVuSans-Oblique.ttf';</code>

1.6.8. PDF::TTFontFile###ProportionalBoldItalic

Description	Value
Description:	Defines the path and TTF-File to handle bold italic proportional font in PDF documents.

Description	Value
Group:	Framework
SubGroup:	Core::PDF
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'PDF::TTFontFile'}->{'ProportionalBoldItalic'} = 'DejaVuSans-BoldOblique.ttf';</pre>

1.6.9. PDF::TTFontFile###Monospaced

Description	Value
Description:	Defines the path and TTF-File to handle monospaced font in PDF documents.
Group:	Framework
SubGroup:	Core::PDF
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'PDF::TTFontFile'}->{'Monospaced'} = 'DejaVuSansMono.ttf';</pre>

1.6.10. PDF::TTFontFile###MonospacedBold

Description	Value
Description:	Defines the path and TTF-File to handle bold monospaced font in PDF documents.
Group:	Framework
SubGroup:	Core::PDF
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'PDF::TTFontFile'}->{'MonospacedBold'} = 'DejaVuSansMono-Bold.ttf';</pre>

1.6.11. PDF::TTFontFile###MonospacedItalic

Description	Value
Description:	Defines the path and TTF-File to handle italic monospaced font in PDF documents.
Group:	Framework
SubGroup:	Core::PDF
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'PDF::TTFontFile'}->{'MonospacedItalic'} = 'DejaVuSansMono-Oblique.ttf';</pre>

1.6.12. PDF::TTFontFile###MonospacedBoldItalic

Description	Value
Description:	Defines the path and TTF-File to handle bold italic monospaced font in PDF documents.
Group:	Framework
SubGroup:	Core::PDF
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'PDF::TTFontFile'}->{'MonospacedBoldItalic'} = 'DejaVuSansMono-BoldOblique.ttf';</pre>

1.7. Core::Package

1.7.1. Package::FileUpload

Description	Value
Description:	Enables file upload in the package manager frontend.
Group:	Framework
SubGroup:	Core::Package
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Package::FileUpload'} = '1';</pre>

1.7.2. Package::RepositoryRoot

Description	Value
Description:	Defines the location to get online repository list for additional packages. The first available result will be used.
Group:	Framework
SubGroup:	Core::Package
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Package::RepositoryRoot'} = ['http://ftp.otrs.org/pub/otrs/misc/packages/repository.xml', 'http://otrs.org/repository.xml'];</pre>

1.7.3. Package::RepositoryList

Description	Value
Description:	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".

Description	Value
Group:	Framework
SubGroup:	Core::Package
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Package::RepositoryList'} = { 'ftp://ftp.example.com/pub/otrs/misc/packages/' => '[Example] ftp://ftp.example.com/' };</pre>

1.7.4. Package::RepositoryAccessRegExp

Description	Value
Description:	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.
Group:	Framework
SubGroup:	Core::Package
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Package::RepositoryAccessRegExp'} = '127\\.0\\.0\\.1';</pre>

1.7.5. Package::Timeout

Description	Value
Description:	Sets the timeout (in seconds) for package downloads. Overwrites "WebUserAgent::Timeout".
Group:	Framework
SubGroup:	Core::Package
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Package::Timeout'} = '15';</pre>

1.7.6. Package::Proxy

Description	Value
Description:	Fetches packages via proxy. Overwrites "WebUserAgent::Proxy".
Group:	Framework
SubGroup:	Core::Package
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Package::Proxy'} = 'http://proxy.sn.no:8001/';</pre>

1.7.7. Package::ShowFeatureAddons

Description	Value
Description:	Toggles display of OTRS FeatureAddons list in PackageManager.
Group:	Framework
SubGroup:	Core::Package
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Package::ShowFeatureAddons'} = '1';</pre>

1.8. Core::PerformanceLog

1.8.1. PerformanceLog

Description	Value
Description:	Enables performance log (to log the page response time). It will affect the system performance. Frontend::Module###AdminPerformanceLog must be enabled.
Group:	Framework
SubGroup:	Core::PerformanceLog
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PerformanceLog'} = '0';</pre>

1.8.2. PerformanceLog::File

Description	Value
Description:	Specifies the path of the file for the performance log.
Group:	Framework
SubGroup:	Core::PerformanceLog
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'PerformanceLog::File'} = '<OTRS_CONFIG_Home>/var/log/Performance.log';</pre>

1.8.3. PerformanceLog::FileMax

Description	Value
Description:	Defines the maximum size (in MB) of the log file.
Group:	Framework
SubGroup:	Core::PerformanceLog
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<code>\$Self->{'PerformanceLog::FileMax'} = '25';</code>

1.9. Core::ReferenceData

1.9.1. ReferenceData::OwnCountryList

Description	Value
Description:	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.
Group:	Framework
SubGroup:	Core::ReferenceData
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'ReferenceData::OwnCountryList'} = { 'AT' => 'Austria', 'CH' => 'Switzerland', 'DE' => 'Germany' };</pre>

1.10. Core::SOAP

1.10.1. SOAP::User

Description	Value
Description:	Defines the username to access the SOAP handle (bin/cgi-bin/rpc.pl).
Group:	Framework
SubGroup:	Core::SOAP
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'SOAP::User'} = 'some_user';</code>

1.10.2. SOAP::Password

Description	Value
Description:	Defines the password to access the SOAP handle (bin/cgi-bin/rpc.pl).
Group:	Framework
SubGroup:	Core::SOAP
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'SOAP::Password'} = 'some_pass';</code>

1.10.3. SOAP::Keep-Alive

Description	Value
Description:	Enable keep-alive connection header for SOAP responses.

Description	Value
Group:	Framework
SubGroup:	Core::SOAP
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SOAP::Keep-Alive'} = '0';</code>

1.11. Core::Sendmail

1.11.1. SendmailModule

Description	Value
Description:	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.
Group:	Framework
SubGroup:	Core::Sendmail
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SendmailModule'} = 'Kernel::System::Email::Sendmail';</code>

1.11.2. SendmailModule::CMD

Description	Value
Description:	If "Sendmail" was selected as SendmailModule, the location of the sendmail binary and the needed options must be specified.
Group:	Framework
SubGroup:	Core::Sendmail
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SendmailModule::CMD'} = '/usr/sbin/sendmail -i -f';</code>

1.11.3. SendmailModule::Host

Description	Value
Description:	If any of the "SMTP" mechanisms was selected as SendmailModule, the mailhost that sends out the mails must be specified.
Group:	Framework
SubGroup:	Core::Sendmail
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SendmailModule::Host'} = 'mail.example.com';</code>

1.11.4. SendmailModule::Port

Description	Value
Description:	If any of the "SMTP" mechanisms was selected as SendmailModule, the port where your mailserver is listening for incoming connections must be specified.
Group:	Framework
SubGroup:	Core::Sendmail
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'SendmailModule::Port'} = '25';</code>

1.11.5. SendmailModule::AuthUser

Description	Value
Description:	If any of the "SMTP" mechanisms was selected as SendmailModule, and authentication to the mail server is needed, an username must be specified.
Group:	Framework
SubGroup:	Core::Sendmail
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'SendmailModule::AuthUser'} = 'MailserverLogin';</code>

1.11.6. SendmailModule::AuthPassword

Description	Value
Description:	If any of the "SMTP" mechanisms was selected as SendmailModule, and authentication to the mail server is needed, a password must be specified.
Group:	Framework
SubGroup:	Core::Sendmail
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'SendmailModule::AuthPassword'} = 'MailserverPassword';</code>

1.11.7. SendmailBcc

Description	Value
Description:	Sends all outgoing email via bcc to the specified address. Please use this only for backup reasons.
Group:	Framework
SubGroup:	Core::Sendmail
Valid:	1
Required:	0
Config-Setting:	

Description	Value
	<code>\$Self->{'SendmailBcc'} = '';</code>

1.11.8. SendmailNotificationEnvelopeFrom

Description	Value
Description:	If set, this address is used as envelope sender header in outgoing notifications. If no address is specified, the envelope sender header is empty.
Group:	Framework
SubGroup:	Core::Sendmail
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'SendmailNotificationEnvelopeFrom'} = '';</code>

1.11.9. SendmailEncodingForce

Description	Value
Description:	Forces encoding of outgoing emails (7bit 8bit quoted-printable base64).
Group:	Framework
SubGroup:	Core::Sendmail
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'SendmailEncodingForce'} = 'base64';</code>

1.12. Core::Session

1.12.1. SessionModule

Description	Value
Description:	Defines the module used to store the session data. With "DB" the frontend server can be splitted from the db server. "FS" is faster.
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionModule'} = 'Kernel::System::AuthSession::DB';</code>

1.12.2. SessionName

Description	Value
Description:	Defines the name of the session key. E.g. Session, SessionID or OTRS.
Group:	Framework
SubGroup:	Core::Session

Description	Value
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionName'} = 'Session';</code>

1.12.3. SessionCheckRemoteIP

Description	Value
Description:	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.
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionCheckRemoteIP'} = '1';</code>

1.12.4. SessionDeleteIfNotRemoteID

Description	Value
Description:	Deletes a session if the session id is used with an invalid remote IP address.
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionDeleteIfNotRemoteID'} = '1';</code>

1.12.5. SessionMaxTime

Description	Value
Description:	Defines the maximal valid time (in seconds) for a session id.
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionMaxTime'} = '57600';</code>

1.12.6. SessionMaxIdleTime

Description	Value
Description:	Sets the inactivity time (in seconds) to pass before a session is killed and a user is logged out.
Group:	Framework

Description	Value
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionMaxIdleTime'} = '21600';</code>

1.12.7. SessionDeleteIfTimeToOld

Description	Value
Description:	Deletes requested sessions if they have timed out.
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionDeleteIfTimeToOld'} = '1';</code>

1.12.8. SessionUseCookie

Description	Value
Description:	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.
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionUseCookie'} = '1';</code>

1.12.9. SessionUseCookieAfterBrowserClose

Description	Value
Description:	Stores cookies after the browser has been closed.
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionUseCookieAfterBrowserClose'} = '0';</code>

1.12.10. SessionCSRFProtection

Description	Value
Description:	Protection against CSRF (Cross Site Request Forgery) exploits (for more info see http://en.wikipedia.org/wiki/Cross-site_request_forgery).
Group:	Framework

Description	Value
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionCSRFProtection'} = '1';</code>

1.12.11. SessionDir

Description	Value
Description:	If "FS" was selected for SessionModule, a directory where the session data will be stored must be specified.
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionDir'} = '<OTRS_CONFIG_Home>/var/sessions';</code>

1.12.12. SessionTable

Description	Value
Description:	If "DB" was selected for SessionModule, a table in database where session data will be stored must be specified.
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionTable'} = 'sessions';</code>

1.12.13. SessionTableID

Description	Value
Description:	If "DB" was selected for SessionModule, a column for the identifiers in session table must be specified.
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionTableID'} = 'session_id';</code>

1.12.14. SessionTableValue

Description	Value
Description:	If "DB" was selected for SessionModule, a column for the values in session table must be specified.
Group:	Framework

Description	Value
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionTableValue'} = 'session_value';</code>

1.12.15. SessionSaveLoginPasswd

Description	Value
Description:	Saves the login and password on the session table in the database, if "DB" was selected for SessionModule.
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SessionSaveLoginPasswd'} = '0';</code>

1.12.16. CustomerPanelSessionName

Description	Value
Description:	Defines the name of the key for customer sessions.
Group:	Framework
SubGroup:	Core::Session
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CustomerPanelSessionName'} = 'CSID';</code>

1.13. Core::SpellChecker

1.13.1. SpellChecker

Description	Value
Description:	Enables spell checker support.
Group:	Framework
SubGroup:	Core::SpellChecker
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SpellChecker'} = '0';</code>

1.13.2. SpellCheckerBin

Description	Value
Description:	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.

Description	Value
Group:	Framework
SubGroup:	Core::SpellChecker
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'SpellCheckerBin'} = '/usr/bin/ispell';</pre>

1.13.3. SpellCheckerDictDefault

Description	Value
Description:	Defines the default spell checker dictionary.
Group:	Framework
SubGroup:	Core::SpellChecker
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'SpellCheckerDictDefault'} = 'english';</pre>

1.13.4. SpellCheckerIgnore

Description	Value
Description:	Defines a default list of words, that are ignored by the spell checker.
Group:	Framework
SubGroup:	Core::SpellChecker
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'SpellCheckerIgnore'} = ['www', 'webmail', 'https', 'http', 'html', 'rfc'];</pre>

1.14. Core::Stats

1.14.1. Stats::StatsHook

Description	Value
Description:	Sets the stats hook.
Group:	Framework
SubGroup:	Core::Stats
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<code>\$Self->{'Stats::StatsHook'} = 'Stat#';</code>

1.14.2. Stats::StatsStartNumber

Description	Value
Description:	Start number for statistics counting. Every new stat increments this number.
Group:	Framework
SubGroup:	Core::Stats
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::StatsStartNumber'} = '10000';</code>

1.14.3. Stats::MaxXaxisAttributes

Description	Value
Description:	Defines the default maximum number of X-axis attributes for the time scale.
Group:	Framework
SubGroup:	Core::Stats
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Stats::MaxXaxisAttributes'} = '1000';</code>

1.15. Core::Stats::Graph

1.15.1. Stats::Graph::t_margin

Description	Value
Description:	Specifies the top margin of the chart.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::t_margin'} = '10';</code>

1.15.2. Stats::Graph::l_margin

Description	Value
Description:	Specifies the left margin of the chart.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1

Description	Value
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::l_margin'} = '10';</code>

1.15.3. Stats::Graph::b_margin

Description	Value
Description:	Specifies the bottom margin of the chart.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::b_margin'} = '10';</code>

1.15.4. Stats::Graph::r_margin

Description	Value
Description:	Specifies the right margin of the chart.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::r_margin'} = '20';</code>

1.15.5. Stats::Graph::bgclr

Description	Value
Description:	Specifies the background color of the picture.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::bgclr'} = 'white';</code>

1.15.6. Stats::Graph::transparent

Description	Value
Description:	Makes the picture transparent.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1

Description	Value
Config-Setting:	<code>\$Self->{'Stats::Graph::transparent'} = '0';</code>

1.15.7. Stats::Graph::fgclr

Description	Value
Description:	Specifies the border color of the chart.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::fgclr'} = 'black';</code>

1.15.8. Stats::Graph::boxclr

Description	Value
Description:	Specifies the background color of the chart.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::boxclr'} = 'white';</code>

1.15.9. Stats::Graph::accentclr

Description	Value
Description:	Specifies the border color of the legend.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::accentclr'} = 'black';</code>

1.15.10. Stats::Graph::legendclr

Description	Value
Description:	Specifies the text color of the legend.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<code>\$Self->{'Stats::Graph::legendclr'} = 'black';</code>

1.15.11. Stats::Graph::textclr

Description	Value
Description:	Specifies the text color of the chart (e. g. caption).
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::textclr'} = 'black';</code>

1.15.12. Stats::Graph::dclrs

Description	Value
Description:	Defines the colors for the graphs.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'Stats::Graph::dclrs'} = ['red', 'green', 'blue', 'yellow', 'black', 'purple', 'orange', 'pink', 'marine', 'cyan', 'lgray', 'lblue', 'lyellow', 'lgreen', 'lred', 'lpurple', 'lorange', 'lbrown']; </pre>

1.15.13. Stats::Graph::line_width

Description	Value
Description:	Defines the boldness of the line drawn by the graph.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<code>\$Self->{'Stats::Graph::line_width'} = '1';</code>

1.15.14. Stats::Graph::legend_placement

Description	Value
Description:	Defines the placement of the legend. This should be a two letter key of the form: 'B[LCR] R[TCB]'. The first letter indicates the placement (Bottom or Right), and the second letter the alignment (Left, Right, Center, Top, or Bottom).
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::legend_placement'} = 'BC';</code>

1.15.15. Stats::Graph::legend_spacing

Description	Value
Description:	Defines the spacing of the legends.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::legend_spacing'} = '4';</code>

1.15.16. Stats::Graph::legend_marker_width

Description	Value
Description:	Defines the width of the legend.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::legend_marker_width'} = '12';</code>

1.15.17. Stats::Graph::legend_marker_height

Description	Value
Description:	Defines the height of the legend.
Group:	Framework
SubGroup:	Core::Stats::Graph
Valid:	1

Description	Value
Required:	1
Config-Setting:	<code>\$Self->{'Stats::Graph::legend_marker_height'} = '8';</code>

1.16. Core::Time

1.16.1. TimeInputFormat

Description	Value
Description:	Defines the date input format used in forms (option or input fields).
Group:	Framework
SubGroup:	Core::Time
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeInputFormat'} = 'Option';</code>

1.16.2. TimeShowAlwaysLong

Description	Value
Description:	Shows time in long format (days, hours, minutes), if set to "Yes"; or in short format (days, hours), if set to "No".
Group:	Framework
SubGroup:	Core::Time
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeShowAlwaysLong'} = '0';</code>

1.16.3. TimeZone

Description	Value
Description:	Sets the system time zone (required a system with UTC as system time). Otherwise this is a diff time to the local time.
Group:	Framework
SubGroup:	Core::Time
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'TimeZone'} = '+0';</code>

1.16.4. TimeZoneUser

Description	Value
Description:	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.

Description	Value
Group:	Framework
SubGroup:	Core::Time
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'TimeZoneUser'} = '0';</code>

1.16.5. TimeZoneUserBrowserAutoOffset

Description	Value
Description:	Sets the user time zone per user based on java script / browser time zone offset feature at login time.
Group:	Framework
SubGroup:	Core::Time
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'TimeZoneUserBrowserAutoOffset'} = '1';</code>

1.16.6. CalendarWeekDayStart

Description	Value
Description:	Define the start day of the week for the date picker.
Group:	Framework
SubGroup:	Core::Time
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CalendarWeekDayStart'} = '1';</code>

1.16.7. TimeVacationDays

Description	Value
Description:	Adds the permanent vacation days. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeVacationDays'} = { '1' => { '1' => 'New Year\'s Day' }, }, '12' => { '24' => 'Christmas Eve', '25' => 'First Christmas Day', '26' => 'Second Christmas Day',</code>

Description	Value
	<pre>'31' => 'New Year\'s Eve' }, '5' => { '1' => 'International Workers\' Day' } };</pre>

1.16.8. TimeVacationDaysOneTime

Description	Value
Description:	Adds the one time vacation days. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeVacationDaysOneTime'} = { '2004' => { '1' => { '1' => 'test' } } } };</pre>

1.16.9. TimeWorkingHours

Description	Value
Description:	Defines the hours and week days to count the working time.
Group:	Framework
SubGroup:	Core::Time
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeWorkingHours'} = { 'Fri' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Mon' => ['8', '9', '10', '11', '12', '13', '14',</pre>

Description	Value
	<pre> '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']]; </pre>

1.16.10. TimeShowCompleteDescription

Description	Value
Description:	Shows time use complete description (days, hours, minutes), if set to "Yes"; or just first letter (d, h, m), if set to "No".
Group:	Framework
SubGroup:	Core::Time
Valid:	1
Required:	1

Description	Value
Config-Setting:	<code>\$Self->{'TimeShowCompleteDescription'} = '0';</code>

1.17. Core::Time::Calendar1

1.17.1. TimeZone::Calendar1Name

Description	Value
Description:	Defines the name of the indicated calendar.
Group:	Framework
SubGroup:	Core::Time::Calendar1
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeZone::Calendar1Name'} = 'Calendar Name 1';</code>

1.17.2. TimeZone::Calendar1

Description	Value
Description:	Defines the time zone of the indicated calendar, which can be assigned later to a specific queue.
Group:	Framework
SubGroup:	Core::Time::Calendar1
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeZone::Calendar1'} = '0';</code>

1.17.3. TimeVacationDays::Calendar1

Description	Value
Description:	Adds the permanent vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar1
Valid:	1
Required:	1
Config-Setting:	<pre>\$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' } }</pre>

Description	Value
	} };

1.17.4. TimeVacationDaysOneTime::Calendar1

Description	Value
Description:	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).
Group:	Framework
SubGroup:	Core::Time::Calendar1
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeVacationDaysOneTime::Calendar1'} = { '2004' => { '1' => { '1' => 'test' } } };</pre>

1.17.5. TimeWorkingHours::Calendar1

Description	Value
Description:	Defines the hours and week days of the indicated calendar, to count the working time.
Group:	Framework
SubGroup:	Core::Time::Calendar1
Valid:	1
Required:	1
Config-Setting:	<pre>\$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',</pre>

Description	Value
	<pre> '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']] }; </pre>

1.18. Core::Time::Calendar2

1.18.1. TimeZone::Calendar2Name

Description	Value
Description:	Defines the name of the indicated calendar.
Group:	Framework
SubGroup:	Core::Time::Calendar2
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeZone::Calendar2Name'} = 'Calendar Name 2';</code>

1.18.2. TimeZone::Calendar2

Description	Value
Description:	Defines the time zone of the indicated calendar, which can be assigned later to a specific queue.
Group:	Framework
SubGroup:	Core::Time::Calendar2
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeZone::Calendar2'} = '0';</pre>

1.18.3. TimeVacationDays::Calendar2

Description	Value
Description:	Adds the permanent vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar2
Valid:	1
Required:	1
Config-Setting:	<pre>\$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' } };</pre>

1.18.4. TimeVacationDaysOneTime::Calendar2

Description	Value
Description:	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).
Group:	Framework
SubGroup:	Core::Time::Calendar2
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeVacationDaysOneTime::Calendar2'} = { '2004' => { '1' => { '1' => 'test' } } };</pre>

1.18.5. TimeWorkingHours::Calendar2

Description	Value
Description:	Defines the hours and week days of the indicated calendar, to count the working time.
Group:	Framework
SubGroup:	Core::Time::Calendar2
Valid:	1
Required:	1
Config-Setting:	<pre> \$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', </pre>

Description	Value
	<pre>'17', '18', '19', '20'], 'Wed' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20']];</pre>

1.19. Core::Time::Calendar3

1.19.1. TimeZone::Calendar3Name

Description	Value
Description:	Defines the name of the indicated calendar.
Group:	Framework
SubGroup:	Core::Time::Calendar3
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeZone::Calendar3Name'} = 'Calendar Name 3';</code>

1.19.2. TimeZone::Calendar3

Description	Value
Description:	Defines the time zone of the indicated calendar, which can be assigned later to a specific queue.
Group:	Framework
SubGroup:	Core::Time::Calendar3
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeZone::Calendar3'} = '0';</code>

1.19.3. TimeVacationDays::Calendar3

Description	Value
Description:	Adds the permanent vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar3
Valid:	1
Required:	1

Description	Value
Config-Setting:	<pre>\$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' } };</pre>

1.19.4. TimeVacationDaysOneTime::Calendar3

Description	Value
Description:	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).
Group:	Framework
SubGroup:	Core::Time::Calendar3
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeVacationDaysOneTime::Calendar3'} = { '2004' => { '1' => { '1' => 'test' } } };</pre>

1.19.5. TimeWorkingHours::Calendar3

Description	Value
Description:	Defines the hours and week days of the indicated calendar, to count the working time.
Group:	Framework
SubGroup:	Core::Time::Calendar3
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeWorkingHours::Calendar3'} = { 'Fri' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Mon' => [</pre>

Description	Value
	<pre> '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']] }; </pre>

1.20. Core::Time::Calendar4

1.20.1. TimeZone::Calendar4Name

Description	Value
Description:	Defines the name of the indicated calendar.

Description	Value
Group:	Framework
SubGroup:	Core::Time::Calendar4
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeZone::Calendar4Name'} = 'Calendar Name 4';</code>

1.20.2. TimeZone::Calendar4

Description	Value
Description:	Defines the time zone of the indicated calendar, which can be assigned later to a specific queue.
Group:	Framework
SubGroup:	Core::Time::Calendar4
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeZone::Calendar4'} = '0';</code>

1.20.3. TimeVacationDays::Calendar4

Description	Value
Description:	Adds the permanent vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar4
Valid:	1
Required:	1
Config-Setting:	<pre> \$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' } }; </pre>

1.20.4. TimeVacationDaysOneTime::Calendar4

Description	Value
Description:	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).
Group:	Framework
SubGroup:	Core::Time::Calendar4
Valid:	1

Description	Value
Required:	1
Config-Setting:	<pre>\$Self->{'TimeVacationDaysOneTime::Calendar4'} = { '2004' => { '1' => { '1' => 'test' } } };</pre>

1.20.5. TimeWorkingHours::Calendar4

Description	Value
Description:	Defines the hours and week days of the indicated calendar, to count the working time.
Group:	Framework
SubGroup:	Core::Time::Calendar4
Valid:	1
Required:	1
Config-Setting:	<pre>\$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',</pre>

Description	Value
	<pre>'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']];</pre>

1.21. Core::Time::Calendar5

1.21.1. TimeZone::Calendar5Name

Description	Value
Description:	Defines the name of the indicated calendar.
Group:	Framework
SubGroup:	Core::Time::Calendar5
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeZone::Calendar5Name'} = 'Calendar Name 5';</code>

1.21.2. TimeZone::Calendar5

Description	Value
Description:	Defines the time zone of the indicated calendar, which can be assigned later to a specific queue.
Group:	Framework
SubGroup:	Core::Time::Calendar5
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeZone::Calendar5'} = '0';</code>

1.21.3. TimeVacationDays::Calendar5

Description	Value
Description:	Adds the permanent vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar5
Valid:	1
Required:	1
Config-Setting:	<pre> \$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' } } </pre>

1.21.4. TimeVacationDaysOneTime::Calendar5

Description	Value
Description:	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).
Group:	Framework
SubGroup:	Core::Time::Calendar5
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'TimeVacationDaysOneTime::Calendar5'} = { '2004' => { '1' => { '1' => 'test' } } } </pre>

1.21.5. TimeWorkingHours::Calendar5

Description	Value
Description:	Defines the hours and week days of the indicated calendar, to count the working time.
Group:	Framework
SubGroup:	Core::Time::Calendar5
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'TimeWorkingHours::Calendar5'} = { 'Fri' => ['8', '9',] } </pre>

Description	Value
	<pre> '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', </pre>

Description	Value
	<pre>'18', '19', '20'] };</pre>

1.22. Core::Time::Calendar6

1.22.1. TimeZone::Calendar6Name

Description	Value
Description:	Defines the name of the indicated calendar.
Group:	Framework
SubGroup:	Core::Time::Calendar6
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeZone::Calendar6Name'} = 'Calendar Name 6';</pre>

1.22.2. TimeZone::Calendar6

Description	Value
Description:	Defines the time zone of the indicated calendar, which can be assigned later to a specific queue.
Group:	Framework
SubGroup:	Core::Time::Calendar6
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeZone::Calendar6'} = '0';</pre>

1.22.3. TimeVacationDays::Calendar6

Description	Value
Description:	Adds the permanent vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar6
Valid:	1
Required:	1
Config-Setting:	<pre>\$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' => {</pre>

Description	Value
	<pre>'1' => 'International Workers\' Day' } };</pre>

1.22.4. TimeVacationDaysOneTime::Calendar6

Description	Value
Description:	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).
Group:	Framework
SubGroup:	Core::Time::Calendar6
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeVacationDaysOneTime::Calendar6'} = { '2004' => { '1' => { '1' => 'test' } } };</pre>

1.22.5. TimeWorkingHours::Calendar6

Description	Value
Description:	Defines the hours and week days of the indicated calendar, to count the working time.
Group:	Framework
SubGroup:	Core::Time::Calendar6
Valid:	1
Required:	1
Config-Setting:	<pre>\$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',</pre>

Description	Value
	<pre> '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']] }; </pre>

1.23. Core::Time::Calendar7

1.23.1. TimeZone::Calendar7Name

Description	Value
Description:	Defines the name of the indicated calendar.
Group:	Framework
SubGroup:	Core::Time::Calendar7
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeZone::Calendar7Name'} = 'Calendar Name 7';</code>

1.23.2. TimeZone::Calendar7

Description	Value
Description:	Defines the time zone of the indicated calendar, which can be assigned later to a specific queue.
Group:	Framework
SubGroup:	Core::Time::Calendar7
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeZone::Calendar7'} = '0';</pre>

1.23.3. TimeVacationDays::Calendar7

Description	Value
Description:	Adds the permanent vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar7
Valid:	1
Required:	1
Config-Setting:	<pre>\$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' } };</pre>

1.23.4. TimeVacationDaysOneTime::Calendar7

Description	Value
Description:	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).
Group:	Framework
SubGroup:	Core::Time::Calendar7
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeVacationDaysOneTime::Calendar7'} = { '2004' => { '1' => { '1' => 'test' } } };</pre>

1.23.5. TimeWorkingHours::Calendar7

Description	Value
Description:	Defines the hours and week days of the indicated calendar, to count the working time.
Group:	Framework
SubGroup:	Core::Time::Calendar7
Valid:	1
Required:	1
Config-Setting:	<pre> \$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',] } </pre>

Description	Value
	<pre>'17', '18', '19', '20'], 'Wed' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20']];</pre>

1.24. Core::Time::Calendar8

1.24.1. TimeZone::Calendar8Name

Description	Value
Description:	Defines the name of the indicated calendar.
Group:	Framework
SubGroup:	Core::Time::Calendar8
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeZone::Calendar8Name'} = 'Calendar Name 8';</code>

1.24.2. TimeZone::Calendar8

Description	Value
Description:	Defines the time zone of the indicated calendar, which can be assigned later to a specific queue.
Group:	Framework
SubGroup:	Core::Time::Calendar8
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeZone::Calendar8'} = '0';</code>

1.24.3. TimeVacationDays::Calendar8

Description	Value
Description:	Adds the permanent vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar8
Valid:	1
Required:	1

Description	Value
Config-Setting:	<pre>\$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' } };</pre>

1.24.4. TimeVacationDaysOneTime::Calendar8

Description	Value
Description:	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).
Group:	Framework
SubGroup:	Core::Time::Calendar8
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeVacationDaysOneTime::Calendar8'} = { '2004' => { '1' => { '1' => 'test' } } };</pre>

1.24.5. TimeWorkingHours::Calendar8

Description	Value
Description:	Defines the hours and week days of the indicated calendar, to count the working time.
Group:	Framework
SubGroup:	Core::Time::Calendar8
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeWorkingHours::Calendar8'} = { 'Fri' => ['8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20'], 'Mon' => [</pre>

Description	Value
	<pre> '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']] }; </pre>

1.25. Core::Time::Calendar9

1.25.1. TimeZone::Calendar9Name

Description	Value
Description:	Defines the name of the indicated calendar.

Description	Value
Group:	Framework
SubGroup:	Core::Time::Calendar9
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeZone::Calendar9Name'} = 'Calendar Name 9';</code>

1.25.2. TimeZone::Calendar9

Description	Value
Description:	Defines the time zone of the indicated calendar, which can be assigned later to a specific queue.
Group:	Framework
SubGroup:	Core::Time::Calendar9
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TimeZone::Calendar9'} = '0';</code>

1.25.3. TimeVacationDays::Calendar9

Description	Value
Description:	Adds the permanent vacation days for the indicated calendar. Please use single digit pattern for numbers from 1 to 9 (instead of 01 - 09).
Group:	Framework
SubGroup:	Core::Time::Calendar9
Valid:	1
Required:	1
Config-Setting:	<pre> \$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' } }; </pre>

1.25.4. TimeVacationDaysOneTime::Calendar9

Description	Value
Description:	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).
Group:	Framework
SubGroup:	Core::Time::Calendar9

Description	Value
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'TimeVacationDaysOneTime::Calendar9'} = { '2004' => { '1' => { '1' => 'test' } } };</pre>

1.25.5. TimeWorkingHours::Calendar9

Description	Value
Description:	Defines the hours and week days of the indicated calendar, to count the working time.
Group:	Framework
SubGroup:	Core::Time::Calendar9
Valid:	1
Required:	1
Config-Setting:	<pre>\$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',</pre>

Description	Value
	<pre>'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']];</pre>

1.26. Core::Web

1.26.1. Frontend::WebPath

Description	Value
Description:	Defines the URL base path of icons, CSS and Java Script.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::WebPath'} = '/otrs-web/';</code>

1.26.2. Frontend::ImagePath

Description	Value
Description:	Defines the URL image path of icons for navigation.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<code>\$Self->{'Frontend::ImagePath'} = '<OTRS_CONFIG_Frontend::WebPath>skins/Agent/default/img/';</code>

1.26.3. Frontend::CSSPath

Description	Value
Description:	Defines the URL CSS path.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::CSSPath'} = '<OTRS_CONFIG_Frontend::WebPath>css/';</code>

1.26.4. Frontend::JavaScriptPath

Description	Value
Description:	Defines the URL java script path.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::JavaScriptPath'} = '<OTRS_CONFIG_Frontend::WebPath>js/';</code>

1.26.5. Frontend::RichText

Description	Value
Description:	Uses richtext for viewing and editing: articles, salutations, signatures, standard responses, auto responses and notifications.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::RichText'} = '1';</code>

1.26.6. Frontend::RichTextPath

Description	Value
Description:	Defines the URL rich text editor path.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<code>\$Self->{'Frontend::RichTextPath'} = '<OTRS_CONFIG_Frontend::WebPath>js/thirdparty/ckeditor-3.6.6/';</code>

1.26.7. Frontend::RichTextWidth

Description	Value
Description:	Defines the width for the rich text editor component. Enter number (pixels) or percent value (relative).
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::RichTextWidth'} = '620';</code>

1.26.8. Frontend::RichTextHeight

Description	Value
Description:	Defines the height for the rich text editor component. Enter number (pixels) or percent value (relative).
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::RichTextHeight'} = '320';</code>

1.26.9. Frontend::RichText::DefaultCSS

Description	Value
Description:	Defines the default CSS used in rich text editors.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::RichText::DefaultCSS'} = 'font-family:Geneva,Helvetica,Arial,sans-serif; font-size: 12px;';</code>

1.26.10. DefaultViewNewLine

Description	Value
Description:	Automated line break in text messages after x number of chars.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<code>\$Self->{'DefaultViewNewLine'} = '90';</code>

1.26.11. DefaultViewLines

Description	Value
Description:	Sets the number of lines that are displayed in text messages (e.g. ticket lines in the QueueZoom).
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'DefaultViewLines'} = '6000';</code>

1.26.12. Frontend::AnimationEnabled

Description	Value
Description:	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.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::AnimationEnabled'} = '1';</code>

1.26.13. AttachmentDownloadType

Description	Value
Description:	Allows choosing between showing the attachments of a ticket in the browser (inline) or just make them downloadable (attachment).
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'AttachmentDownloadType'} = 'attachment';</code>

1.26.14. WebMaxFileUpload

Description	Value
Description:	Defines the maximal size (in bytes) for file uploads via the browser.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<code>\$Self->{'WebMaxFileUpload'} = '16777216';</code>

1.26.15. WebUploadCacheModule

Description	Value
Description:	Selects the module to handle uploads via the web interface. "DB" stores all uploads in the database, "FS" uses the file system.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'WebUploadCacheModule'} = 'Kernel::System::Web::UploadCache::DB';</code>

1.26.16. Frontend::Output::FilterText###AAAURL

Description	Value
Description:	Defines the filter that processes the text in the articles, in order to highlight URLs.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Frontend::Output::FilterText'}->{'AAAURL'} = { 'Module' => 'Kernel::Output::HTML::OutputFilterTextURL', 'Templates' => { 'AgentTicketZoom' => '1' } } };</code>

1.26.17. Frontend::Themes

Description	Value
Description:	Activates the available themes on the system. Value 1 means active, 0 means inactive.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Frontend::Themes'} = { 'Lite' => '0', 'Standard' => '1' };</code>

1.26.18. Frontend::Output::FilterText###OutputFilterTextAutoLink

Description	Value
Description:	Defines a filter to process the text in the articles, in order to highlight predefined keywords.

Description	Value
Group:	Framework
SubGroup:	Core::Web
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Output::FilterText'}->{'OutputFilterTextAutoLink'} = { 'Module' => 'Kernel::Output::HTML::OutputFilterTextAutoLink', 'Templates' => { 'AgentTicketZoom' => '1' } }; </pre>

1.26.19. Frontend::Output::OutputFilterTextAutoLink###CVE

Description	Value
Description:	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.
Group:	Framework
SubGroup:	Core::Web
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Output::OutputFilterTextAutoLink'}->{'CVE'} = { 'RegExp' => ['(CVE CAN)\{3,4}\{2,6}'], '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>' } }; </pre>

1.26.20. Frontend::Output::OutputFilterTextAutoLink###Bugtraq

Description	Value
Description:	Defines a filter for html output to add links behind bugtraq numbers. The element Image allows two input kinds. At once the

Description	Value
	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.
Group:	Framework
SubGroup:	Core::Web
Valid:	0
Required:	0
Config-Setting:	<pre> \$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>' } } }; </pre>

1.26.21. Frontend::Output::OutputFilterTextAutoLink###MSBulletins

Description	Value
Description:	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.
Group:	Framework
SubGroup:	Core::Web
Valid:	0
Required:	0
Config-Setting:	<pre> \$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' }, </pre>

Description	Value
	<pre>'URL2' => { 'Description' => 'Google', 'Image' => 'http://www.google.de/favicon.ico', 'Target' => '_blank', 'URL' => 'http://google.com/search?q=MS<MATCH1>-<MATCH2>' } };</pre>

1.26.22. Frontend::Output::OutputFilterTextAutoLink###Setting1

Description	Value
Description:	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.
Group:	Framework
SubGroup:	Core::Web
Valid:	0
Required:	0
Config-Setting:	<pre>\$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' } } };</pre>

1.26.23. Frontend::Output::OutputFilterTextAutoLink###Setting2

Description	Value
Description:	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.
Group:	Framework
SubGroup:	Core::Web
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Output::OutputFilterTextAutoLink'}->{'Setting2'} = { 'RegExp' => ['RegExp'],],</pre>

Description	Value
	<pre>'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' } };</pre>

1.26.24. Loader::Enabled::CSS

Description	Value
Description:	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.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Loader::Enabled::CSS'} = '1';</code>

1.26.25. Loader::Enabled::JS

Description	Value
Description:	If enabled, OTRS will deliver all JavaScript files in minified form.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Loader::Enabled::JS'} = '1';</code>

1.26.26. Loader::Agent::CommonCSS###000-Framework

Description	Value
Description:	List of CSS files to always be loaded for the agent interface.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1

Description	Value
Config-Setting:	<pre>\$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.Print.css'];</pre>

1.26.27. Loader::Agent::CommonCSS::IE8###000-Framework

Description	Value
Description:	List of IE8-specific CSS files to always be loaded for the agent interface.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Loader::Agent::CommonCSS::IE8'}->{'000-Framework'} = ['Core.OverviewSmall.IE8.css'];</pre>

1.26.28. Loader::Agent::CommonJS###000-Framework

Description	Value
Description:	List of JS files to always be loaded for the agent interface.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Loader::Agent::CommonJS'}->{'000-Framework'} = ['thirdparty/json/json2.js', 'thirdparty/jquery-1.6.4/jquery.js', 'thirdparty/jquery-ui-1.8.16/jquery-ui.js', 'thirdparty/jquery-validate-1.9/jquery.validate.js', 'thirdparty/stacktrace-0.3/stacktrace.js', 'Core.JavaScriptEnhancements.js', 'Core.Debug.js', 'Core.Data.js', 'Core.Config.js', 'Core.Exception.js', 'Core.JSON.js', 'Core.AJAX.js', 'Core.UI.js', 'Core.UI.IE7Fixes.js', 'Core.UI.Accordion.js',</pre>

Description	Value
	<pre>'Core.UI.Datepicker.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.Form.js', 'Core.Form.ErrorTooltips.js', 'Core.Form.Validate.js', 'Core.App.js', 'Core.Agent.js', 'Core.Agent.Search.js'];</pre>

1.26.29. Loader::Customer::CommonCSS###000-Framework

Description	Value
Description:	List of CSS files to always be loaded for the customer interface.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<pre>\$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.Print.css'];</pre>

1.26.30. Loader::Customer::CommonCSS::IE6###000-Framework

Description	Value
Description:	List of IE6-specific CSS files to always be loaded for the customer interface.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Loader::Customer::CommonCSS::IE6'}->{'000-Framework'} = ['Core.IE6.css', 'Core.Tooltip.IE6.css'];</pre>

1.26.31. Loader::Customer::CommonCSS::IE7###000-Framework

Description	Value
Description:	List of IE7-specific CSS files to always be loaded for the customer interface.

Description	Value
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Loader::Customer::CommonCSS::IE7'}->{'000-Framework'} = ['Core.IE7.css', 'Core.Tooltip.IE7.css', 'Core.Dialog.IE7.css'];</pre>

1.26.32. Loader::Customer::CommonCSS::IE8###000-Framework

Description	Value
Description:	List of IE8-specific CSS files to always be loaded for the customer interface.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Loader::Customer::CommonCSS::IE8'}->{'000-Framework'} = [];</pre>

1.26.33. Loader::Customer::CommonJS###000-Framework

Description	Value
Description:	List of JS files to always be loaded for the customer interface.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Loader::Customer::CommonJS'}->{'000-Framework'} = ['thirdparty/jquery-1.6.4/jquery.js', 'thirdparty/jquery-validate-1.9/jquery.validate.js', 'thirdparty/jquery-ui-1.8.16/jquery-ui.js', 'thirdparty/stacktrace-0.3/stacktrace.js', 'Core.Debug.js', 'Core.Data.js', 'Core.Exception.js', 'Core.JavaScriptEnhancements.js', 'Core.Config.js', 'Core.App.js', 'Core.UI.js', 'Core.UI.IE7Fixes.js', 'Core.UI.Accessibility.js', 'Core.UI.Dialog.js', 'Core.UI.RichTextEditor.js', 'Core.UI.Datepicker.js', 'Core.UI.Popup.js', 'Core.Form.js', 'Core.Form.ErrorTooltips.js', 'Core.Form.Validate.js', 'Core.Customer.js'];</pre>

1.26.34. Loader::Agent::DefaultSelectedSkin

Description	Value
Description:	The agent skin's InternalName which should be used in the agent interface. Please check the available skins in Frontend::Agent::Skins.
Group:	Framework
SubGroup:	Core::Web
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Loader::Agent::DefaultSelectedSkin'} = 'default';</pre>

1.26.35. Loader::Customer::SelectedSkin::HostBased

Description	Value
Description:	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.
Group:	Framework
SubGroup:	Core::Web
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Loader::Customer::SelectedSkin::HostBased'} = { 'host1\\.example\\.com' => 'Someskin1', 'host2\\.example\\.com' => 'Someskin2' };</pre>

1.27. Core::WebUserAgent

1.27.1. WebUserAgent::Timeout

Description	Value
Description:	Sets the timeout (in seconds) for http/ftp downloads.
Group:	Framework
SubGroup:	Core::WebUserAgent
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'WebUserAgent::Timeout'} = '15';</pre>

1.27.2. WebUserAgent::Proxy

Description	Value
Description:	Defines the connections for http/ftp, via a proxy.
Group:	Framework
SubGroup:	Core::WebUserAgent

Description	Value
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'WebUserAgent::Proxy'} = 'http://proxy.sn.no:8001/';</code>

1.28. Crypt::PGP

1.28.1. PGP

Description	Value
Description:	Enables PGP support. When PGP support is enabled for signing and securing mail, it is HIGHLY recommended that the web server be run as the OTRS user. Otherwise, there will be problems with the privileges when accessing .gnupg folder.
Group:	Framework
SubGroup:	Crypt::PGP
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PGP'} = '0';</code>

1.28.2. PGP::Bin

Description	Value
Description:	Defines the path to PGP binary.
Group:	Framework
SubGroup:	Crypt::PGP
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PGP::Bin'} = '/usr/bin/gpg';</code>

1.28.3. PGP::Options

Description	Value
Description:	Sets the options for PGP binary.
Group:	Framework
SubGroup:	Crypt::PGP
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PGP::Options'} = '--homedir /opt/otrs/.gnupg/ --batch --no-tty --yes';</code>

1.28.4. PGP::Key::Password

Description	Value
Description:	Sets the password for private PGP key.
Group:	Framework
SubGroup:	Crypt::PGP

Description	Value
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'PGP::Key::Password'} = { '488A0B8F' => 'SomePassword', 'D2DF79FA' => 'SomePassword' };</pre>

1.28.5. PGP::TrustedNetwork

Description	Value
Description:	Set this to yes if you trust in all your public and private pgp keys, even if they are not certified with a trusted signature.
Group:	Framework
SubGroup:	Crypt::PGP
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PGP::TrustedNetwork'} = '0';</pre>

1.28.6. PGP::Log

Description	Value
Description:	Configure your own log text for PGP.
Group:	Framework
SubGroup:	Crypt::PGP
Valid:	1
Required:	1
Config-Setting:	<pre>\$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.', 'SIG_EXPIRED' => '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.' };</pre>

1.29. Crypt::SMIME

1.29.1. SMIME

Description	Value
Description:	Enables S/MIME support.

Description	Value
Group:	Framework
SubGroup:	Crypt::SMIME
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SMIME'} = '0';</code>

1.29.2. SMIME::Bin

Description	Value
Description:	Defines the path to open ssl binary. It may need a HOME env (<code>\$ENV{HOME} = '/var/lib/wwwrun';</code>).
Group:	Framework
SubGroup:	Crypt::SMIME
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SMIME::Bin'} = '/usr/bin/openssl';</code>

1.29.3. SMIME::CertPath

Description	Value
Description:	Specifies the directory where SSL certificates are stored.
Group:	Framework
SubGroup:	Crypt::SMIME
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SMIME::CertPath'} = '/etc/ssl/certs';</code>

1.29.4. SMIME::PrivatePath

Description	Value
Description:	Specifies the directory where private SSL certificates are stored.
Group:	Framework
SubGroup:	Crypt::SMIME
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'SMIME::PrivatePath'} = '/etc/ssl/private';</code>

1.30. Frontend::Admin::AdminCustomerUser

1.30.1. AdminCustomerUser::RunInitialWildcardSearch

Description	Value
Description:	Runs an initial wildcard search of the existing customer users when accessing the AdminCustomerUser module.

Description	Value
Group:	Framework
SubGroup:	Frontend::Admin::AdminCustomerUser
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'AdminCustomerUser::RunInitialWildcardSearch'} = '1';</code>

1.31. Frontend::Admin::ModuleRegistration

1.31.1. Frontend::Module###Admin

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$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::NavBarModuleAdmin' }, 'NavBarName' => 'Admin', 'Title' => '' }; </pre>

1.31.2. Frontend::Module###AdminInit

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1

Description	Value
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminInit'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarName' => '', 'Title' => 'Init' };</pre>

1.31.3. Frontend::Module###AdminUser

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminUser'} = { 'Description' => 'Create and manage agents.', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Agent', 'Description' => 'Create and manage agents.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Agents', 'Prio' => '100' }, 'NavBarName' => 'Admin', 'Title' => 'Agents' };</pre>

1.31.4. Frontend::Module###AdminGroup

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminGroup'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Agent', 'Description' => 'Create and manage groups.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Groups', 'Prio' => '150' }, 'NavBarName' => 'Admin', 'Title' => 'Groups' };</pre>

Description	Value
	};

1.31.5. Frontend::Module###AdminUserGroup

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminUserGroup'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Agent', 'Description' => 'Link agents to groups.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Agents <-> Groups', 'Prio' => '200' }, 'NavBarName' => 'Admin', 'Title' => 'Agents <-> Groups' }; </pre>

1.31.6. Frontend::Module###AdminCustomerUser

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminCustomerUser'} = { 'Description' => 'Edit Customers', 'Group' => ['admin', 'users'], 'GroupRo' => [''], 'Loader' => { 'JavaScript' => ['Core.Agent.TicketAction.js'] }, 'NavBar' => [{ 'AccessKey' => 'c', 'Block' => 'ItemArea', 'Description' => '', 'Link' => 'Action=AdminCustomerUser;Nav=Agent', 'LinkOption' => '', 'Name' => 'Customers', 'NavBar' => 'Customers', 'Prio' => '9000', }] }; </pre>

Description	Value
	<pre> 'Type' => 'Menu' }], 'NavBarModule' => { 'Block' => 'Customer', 'Description' => 'Create and manage customers.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Customers', 'Prio' => '300' }, 'NavBarName' => 'Customers', 'Title' => 'Customers' }; </pre>

1.31.7. Frontend::Module###AdminCustomerCompany

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Frontend::Module'}->{'AdminCustomerCompany'} = { 'Description' => 'Admin', 'Group' => ['admin', 'users'], 'GroupRo' => [''], 'NavBar' => [{ 'AccessKey' => 'c', 'Block' => 'ItemArea', 'Description' => '', 'Link' => 'Action=AdminCustomerCompany;Nav=Agent', 'LinkOption' => '', 'Name' => 'Companies', 'NavBar' => 'Companies', 'Prio' => '9100', 'Type' => 'Menu' }], 'NavBarModule' => { 'Block' => 'Customer', 'Description' => 'Create and manage companies.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Customer Companies', 'Prio' => '310' }, 'NavBarName' => 'Admin', 'Title' => 'Customer Companies' }; </pre>

1.31.8. Frontend::Module###AdminCustomerUserGroup

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration

Description	Value
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminCustomerUserGroup'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Customer', 'Description' => 'Link customers to groups.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Customers <-> Groups', 'Prio' => '400' }, 'NavBarName' => 'Admin', 'Title' => 'Customers <-> Groups' }; </pre>

1.31.9. Frontend::Module###AdminCustomerUserService

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminCustomerUserService'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Customer', 'Description' => 'Link customers to services.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Customers <-> Services', 'Prio' => '500' }, 'NavBarName' => 'Admin', 'Title' => 'Customers <-> Services' }; </pre>

1.31.10. Frontend::Module###AdminRole

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminRole'} = { 'Description' => 'Admin', 'Group' => ['admin'], }; </pre>

Description	Value
	<pre>'NavBarModule' => { 'Block' => 'Agent', 'Description' => 'Create and manage roles.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Roles', 'Prio' => '600' }, 'NavBarName' => 'Admin', 'Title' => 'Roles' };</pre>

1.31.11. Frontend::Module###AdminRoleUser

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminRoleUser'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Agent', 'Description' => 'Link agents to roles.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Agents <-> Roles', 'Prio' => '700' }, 'NavBarName' => 'Admin', 'Title' => 'Agents <-> Roles' };</pre>

1.31.12. Frontend::Module###AdminRoleGroup

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminRoleGroup'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Agent', 'Description' => 'Link roles to groups.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Roles <-> Groups', 'Prio' => '800' }, 'NavBarName' => 'Admin', 'Title' => 'Roles <-> Groups' };</pre>

Description	Value
	};

1.31.13. Frontend::Module###AdminSMIME

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminSMIME'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Email', 'Description' => 'Manage S/MIME certificates for email encryption.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'S/MIME Certificates', 'Prio' => '1100' }, 'NavBarName' => 'Admin', 'Title' => 'S/MIME Management' }; </pre>

1.31.14. Frontend::Module###AdminPGP

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminPGP'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Email', 'Description' => 'Manage PGP keys for email encryption.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'PGP Keys', 'Prio' => '1200' }, 'NavBarName' => 'Admin', 'Title' => 'PGP Key Management' }; </pre>

1.31.15. Frontend::Module###AdminMailAccount

Description	Value
Description:	Frontend module registration for the agent interface.

Description	Value
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$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::NavBarModuleAdmin', 'Name' => 'PostMaster Mail Accounts', 'Prio' => '100' }, 'NavBarName' => 'Admin', 'Title' => 'Mail Accounts' }; </pre>

1.31.16. Frontend::Module###AdminPostMasterFilter

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminPostMasterFilter'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Email', 'Description' => 'Filter incoming emails.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'PostMaster Filters', 'Prio' => '200' }, 'NavBarName' => 'Admin', 'Title' => 'PostMaster Filters' }; </pre>

1.31.17. Frontend::Module###AdminEmail

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	

Description	Value
	<pre>\$Self->{'Frontend::Module'}->{'AdminEmail'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'System', 'Description' => 'Send notifications to users.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Admin Notification', 'Prio' => '400' }, 'NavBarName' => 'Admin', 'Title' => 'Admin Notification' };</pre>

1.31.18. Frontend::Module###AdminSession

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminSession'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'System', 'Description' => 'Manage existing sessions.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Session Management', 'Prio' => '500' }, 'NavBarName' => 'Admin', 'Title' => 'Session Management' };</pre>

1.31.19. Frontend::Module###AdminPerformanceLog

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminPerformanceLog'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'Loader' => { 'CSS' => ['Core.Agent.Admin.PerformanceLog.css'] }, };</pre>

Description	Value
	<pre>'NavBarModule' => { 'Block' => 'System', 'Description' => 'View performance benchmark results.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Performance Log', 'Prio' => '550' }, 'NavBarName' => 'Admin', 'Title' => 'Performance Log' };</pre>

1.31.20. Frontend::Module###AdminLog

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminLog'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'System', 'Description' => 'View system log messages.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'System Log', 'Prio' => '600' }, 'NavBarName' => 'Admin', 'Title' => 'System Log' };</pre>

1.31.21. Frontend::Module###AdminSelectBox

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminSelectBox'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'System', 'Description' => 'Execute SQL statements.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'SQL Box', 'Prio' => '700' }, 'NavBarName' => 'Admin', 'Title' => 'SQL Box' };</pre>

1.31.22. Frontend::Module###AdminPackageManager

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$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::NavBarModuleAdmin', 'Name' => 'Package Manager', 'Prio' => '1000' }, 'NavBarName' => 'Admin', 'Title' => 'Package Manager' }; </pre>

1.31.23. Frontend::Module###AdminGenericInterfaceDebugger

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminGenericInterfaceDebugger'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'Loader' => { 'CSS' => ['Core.Agent.Admin.GenericInterface.css'], 'CSS_IE7' => ['Core.Agent.Admin.GenericInterface.IE7.css'], 'JavaScript' => ['Core.Agent.Admin.GenericInterfaceDebugger.js'] }, 'Title' => 'GenericInterface Debugger GUI' }; </pre>

1.31.24. Frontend::Module###AdminGenericInterfaceWebservice

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework

Description	Value
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$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::NavBarModuleAdmin', 'Name' => 'Web Services', 'Prio' => '1000' }, 'NavBarName' => 'Admin', 'Title' => 'GenericInterface Web Service GUI' }; </pre>

1.31.25. Frontend::Module###AdminGenericInterfaceTransportHTTPSOAP

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}- >{'AdminGenericInterfaceTransportHTTPSOAP'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'Loader' => { 'CSS' => ['Core.Agent.Admin.GenericInterface.css'] }, 'Title' => 'GenericInterface TransportHTTPSOAP GUI' }; </pre>

1.31.26. Frontend::Module###AdminGenericInterfaceWebserviceHistory

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1

Description	Value
Required:	0
Config-Setting:	<pre> \$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' }; </pre>

1.31.27. Frontend::Module###AdminGenericInterfaceOperationDefault

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$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' }; </pre>

1.31.28. Frontend::Module###AdminGenericInterfaceInvokerDefault

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}- >{'AdminGenericInterfaceInvokerDefault'} = { 'Description' => 'Admin', 'Group' => ['admin'] }; </pre>

Description	Value
	<pre>], 'Loader' => { 'CSS' => ['Core.Agent.Admin.GenericInterface.css'], 'JavaScript' => ['Core.Agent.Admin.GenericInterfaceInvoker.js'] }, 'Title' => 'GenericInterface Invoker GUI' }; </pre>

1.31.29. Frontend::Module###AdminGenericInterfaceMappingSimple

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminGenericInterfaceMappingSimple'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'Loader' => { 'CSS' => ['Core.Agent.Admin.GenericInterface.css'], 'CSS_IE7' => ['Core.Agent.Admin.GenericInterface.IE7.css'], 'JavaScript' => ['Core.Agent.Admin.GenericInterfaceMappingSimple.js'] }, 'Title' => 'GenericInterface Webservice Mapping GUI' }; </pre>

1.32. Frontend::Agent

1.32.1. AgentLogo

Description	Value
Description:	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.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'AgentLogo'} = { 'StyleHeight' => '67px', 'StyleRight' => '38px', 'StyleTop' => '-4px', 'StyleWidth' => '244px', }; </pre>

Description	Value
	'URL' => 'skins/Agent/default/img/logo_bg.png' };

1.32.2. AgentLoginLogo

Description	Value
Description:	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.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'AgentLoginLogo'} = { 'StyleHeight' => '100px', 'URL' => 'skins/Agent/default/img/loginlogo_default.png' };</pre>

1.32.3. LoginURL

Description	Value
Description:	Defines an alternate URL, where the login link refers to.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'LoginURL'} = 'http://host.example.com/login.html';</pre>

1.32.4. LogoutURL

Description	Value
Description:	Defines an alternate URL, where the logout link refers to.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'LogoutURL'} = 'http://host.example.com/thanks-for-using-otrs.html';</pre>

1.32.5. PreApplicationModule###AgentInfo

Description	Value
Description:	Defines a useful module to load specific user options or to display news.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	0

Description	Value
Required:	0
Config-Setting:	<code>\$Self->{'PreApplicationModule'}->{'AgentInfo'} = 'Kernel::Modules::AgentInfo';</code>

1.32.6. InfoKey

Description	Value
Description:	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.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'InfoKey'} = 'wpt22';</code>

1.32.7. InfoFile

Description	Value
Description:	File that is displayed in the Kernel::Modules::AgentInfo module, if located under Kernel/Output/HTML/Standard/AgentInfo.dtl.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'InfoFile'} = 'AgentInfo';</code>

1.32.8. LostPassword

Description	Value
Description:	Activates lost password feature for agents, in the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'LostPassword'} = '1';</code>

1.32.9. ShowMotd

Description	Value
Description:	Shows the message of the day on login screen of the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent

Description	Value
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'ShowMotd'} = '0';</code>

1.32.10. NotificationSubjectLostPasswordToken

Description	Value
Description:	Defines the subject for notification mails sent to agents, with token about new requested password.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'NotificationSubjectLostPasswordToken'} = 'New OTRS password request';</code>

1.32.11. NotificationBodyLostPasswordToken

Description	Value
Description:	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).
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<pre> \$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. '; </pre>

1.32.12. NotificationSubjectLostPassword

Description	Value
Description:	Defines the subject for notification mails sent to agents, about new password.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	1

Description	Value
Required:	1
Config-Setting:	<code>\$Self->{'NotificationSubjectLostPassword'} = 'New OTRS password';</code>

1.32.13. NotificationBodyLostPassword

Description	Value
Description:	Defines the body text for notification mails sent to agents, about new password (after using this link the new password will be sent).
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<pre> \$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 '; </pre>

1.32.14. OpenMainMenuOnHover

Description	Value
Description:	If enabled, the first level of the main menu opens on mouse hover (instead of click only).
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'OpenMainMenuOnHover'} = '0';</code>

1.32.15. Loader::Agent::Skin###000-default

Description	Value
Description:	Default skin for interface.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Loader::Agent::Skin'}->{'000-default'} = { 'Description' => 'This is the default orange - black skin.', 'HomePage' => 'www.otrs.org', }; </pre>

Description	Value
	<pre>'InternalName' => 'default', 'VisibleName' => 'Default' };</pre>

1.32.16. Loader::Agent::Skin###001-ivory

Description	Value
Description:	Balanced white skin by Felix Niklas.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Loader::Agent::Skin'}->{'001-ivory'} = { 'Description' => 'Balanced white skin by Felix Niklas', 'HomePage' => 'www.felixniklas.de', 'InternalName' => 'ivory', 'VisibleName' => 'Ivory' };</pre>

1.32.17. Loader::Agent::Skin###001-slim

Description	Value
Description:	Experimental "Slim" skin which tries to save screen space for power users.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Loader::Agent::Skin'}->{'001-slim'} = { 'Description' => 'Experimental "Slim" skin which tries to save screen space for power users.', 'HomePage' => 'www.otrs.org', 'InternalName' => 'slim', 'VisibleName' => 'Slim' };</pre>

1.32.18. Loader::Agent::DefaultSelectedSkin::HostBased

Description	Value
Description:	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.
Group:	Framework
SubGroup:	Frontend::Agent
Valid:	0
Required:	0

Description	Value
Config-Setting:	<pre>\$Self->{'Loader::Agent::DefaultSelectedSkin::HostBased'} = { 'host1\\.example\\.com' => 'SomeSkin1', 'host2\\.example\\.com' => 'SomeSkin2' };</pre>

1.33. Frontend::Agent::Dashboard

1.33.1. DashboardBackend###0000-ProductNotify

Description	Value
Description:	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.
Group:	Framework
SubGroup:	Frontend::Agent::Dashboard
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'DashboardBackend'}->{'0000-ProductNotify'} = { 'Block' => 'ContentLarge', 'CacheTTLLocal' => '1440', 'Default' => '1', 'Description' => 'News about OTRS releases!', 'Group' => 'admin', 'Module' => 'Kernel::Output::HTML::DashboardProductNotify', 'Title' => 'Product News', 'URL' => 'http://otrs.org/product.xml' };</pre>

1.33.2. DashboardBackend###0400-UserOnline

Description	Value
Description:	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.
Group:	Framework
SubGroup:	Frontend::Agent::Dashboard
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'DashboardBackend'}->{'0400-UserOnline'} = { 'Block' => 'ContentSmall', 'CacheTTLLocal' => '5', 'Default' => '0', 'Description' => '', 'Filter' => 'Agent', 'Group' => '', 'IdleMinutes' => '60', 'Limit' => '10', 'Module' => 'Kernel::Output::HTML::DashboardUserOnline', };</pre>

Description	Value
	<pre>'ShowEmail' => '1', 'SortBy' => 'UserLastname', 'Title' => 'Online' };</pre>

1.33.3. DashboardBackend###0410-RSS

Description	Value
Description:	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.
Group:	Framework
SubGroup:	Frontend::Agent::Dashboard
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'DashboardBackend'}->{'0410-RSS'} = { 'Block' => 'ContentSmall', 'CacheTTL' => '360', 'Default' => '1', 'Description' => '', 'Group' => '', 'Limit' => '6', 'Module' => 'Kernel::Output::HTML::DashboardRSS', 'Title' => 'OTRS News', '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' };</pre>

1.33.4. DashboardBackend###0200-Image

Description	Value
Description:	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.
Group:	Framework
SubGroup:	Frontend::Agent::Dashboard
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'DashboardBackend'}->{'0200-Image'} = { 'Block' => 'ContentLarge', 'Default' => '1', 'Description' => 'Some picture description!', 'Group' => '', 'Height' => '140', 'Link' => 'http://otrs.org/', 'LinkTitle' => 'http://otrs.org/',</pre>

Description	Value
	<pre>'Module' => 'Kernel::Output::HTML::DashboardImage', 'Title' => 'A picture', 'URL' => 'http://www.otrs.com/uploads/pics/ jointhecommunity_02.jpg', 'Width' => '198' };</pre>

1.33.5. DashboardBackend###0210-MOTD

Description	Value
Description:	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.
Group:	Framework
SubGroup:	Frontend::Agent::Dashboard
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'DashboardBackend'}->{'0210-MOTD'} = { 'Block' => 'ContentLarge', 'Default' => '1', 'Group' => '', 'Module' => 'Kernel::Output::HTML::DashboardMOTD', 'Title' => 'Message of the Day' };</pre>

1.33.6. DashboardBackend###0300-IFrame

Description	Value
Description:	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.
Group:	Framework
SubGroup:	Frontend::Agent::Dashboard
Valid:	0
Required:	0
Config-Setting:	<pre>\$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::DashboardIFrame', 'Scrolling' => 'auto', 'Title' => 'A Website', 'URL' => 'http://www.otrs.org/', 'Width' => '1024' };</pre>

1.34. Frontend::Agent::LinkObject

1.34.1. Frontend::AgentLinkObject::WildcardSearch

Description	Value
Description:	Starts a wildcard search of the active object after the link object mask is started.
Group:	Framework
SubGroup:	Frontend::Agent::LinkObject
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Frontend::AgentLinkObject::WildcardSearch'} = '0';</pre>

1.35. Frontend::Agent::ModuleMetaHead

1.35.1. Frontend::HeaderMetaModule###100-Refresh

Description	Value
Description:	Defines the module to generate html refresh headers of html sites.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleMetaHead
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Frontend::HeaderMetaModule'}->{'100-Refresh'} = { 'Module' => 'Kernel::Output::HTML::HeaderMetaRefresh' };</pre>

1.36. Frontend::Agent::ModuleNotify

1.36.1. Frontend::NotifyModule###200-UID-Check

Description	Value
Description:	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).
Group:	Framework
SubGroup:	Frontend::Agent::ModuleNotify
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Frontend::NotifyModule'}->{'200-UID-Check'} = { 'Module' => 'Kernel::Output::HTML::NotificationUIDCheck' };</pre>

1.36.2. Frontend::NotifyModule###300-ShowAgentOnline

Description	Value
Description:	Defines the module that shows all the currently logged in agents in the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleNotify

Description	Value
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::NotifyModule'}->{'300-ShowAgentOnline'} = { 'IdleMinutes' => '60', 'Module' => 'Kernel::Output::HTML::NotificationAgentOnline', 'ShowEmail' => '1' };</pre>

1.36.3. Frontend::NotifyModule###400-ShowCustomerOnline

Description	Value
Description:	Defines the module that shows all the currently logged in customers in the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleNotify
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::NotifyModule'}->{'400-ShowCustomerOnline'} = { 'IdleMinutes' => '60', 'Module' => 'Kernel::Output::HTML::NotificationCustomerOnline', 'ShowEmail' => '1' };</pre>

1.36.4. Frontend::NotifyModule###500-OutofOffice-Check

Description	Value
Description:	Defines the module to display a notification in the agent interface, if the agent is logged in while having out-of-office active.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleNotify
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Frontend::NotifyModule'}->{'500-OutofOffice-Check'} = { 'Module' => 'Kernel::Output::HTML::NotificationOutofOfficeCheck' };</pre>

1.36.5. Frontend::NotifyModule###900-Generic

Description	Value
Description:	Defines the module that shows a generic notification in the agent interface. Either "Text" - if configured - or the contents of "File" will be displayed.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleNotify
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::NotifyModule'}->{'900-Generic'} = {</pre>

Description	Value
	<pre>'File' => '<OTRS_CONFIG_Home>/var/notify.txt', 'Link' => 'http://www.otrs.com', 'Module' => 'Kernel::Output::HTML::NotificationGeneric', 'Priority' => 'Warning', 'Text' => 'The OTRS Website' };</pre>

1.37. Frontend::Agent::ModuleRegistration

1.37.1. Frontend::Module###Logout

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'Logout'} = { 'Description' => 'Logout', 'NavBarName' => '', 'Title' => '' };</pre>

1.37.2. Frontend::Module###AgentDashboard

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentDashboard'} = { 'Description' => 'Agent Dashboard', 'Loader' => { 'JavaScript' => ['thirdparty/flot-0.7/excanvas.js', 'thirdparty/flot-0.7/jquery.flot.js', 'Core.UI.Chart.js', 'Core.UI.DnD.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' => '' };</pre>

1.37.3. Frontend::Module###AgentPreferences

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentPreferences'} = { 'Description' => 'Agent Preferences', 'NavBarName' => 'Preferences', 'Title' => '' };</pre>

1.37.4. Frontend::Module###PictureUpload

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'PictureUpload'} = { 'Description' => 'Picture upload module', 'NavBarName' => 'Ticket', 'Title' => 'Picture-Upload' };</pre>

1.37.5. Frontend::Module###AgentSpelling

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentSpelling'} = { 'Description' => 'Spell checker', 'Loader' => { 'JavaScript' => ['Core.Agent.TicketAction.js'] }, 'NavBarName' => '', 'Title' => 'Spell Checker' };</pre>

1.37.6. Frontend::Module###SpellingInline

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework

Description	Value
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'SpellingInline'} = { 'Description' => 'Spell checker', 'NavBarName' => '', 'Title' => 'Spell Checker' };</pre>

1.37.7. Frontend::Module###AgentBook

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentBook'} = { 'Description' => 'Address book of CustomerUser sources', 'Loader' => { 'JavaScript' => ['Core.Agent.CustomerSearch.js', 'Core.Agent.TicketAction.js'] }, 'NavBarName' => '', 'Title' => 'Address Book' };</pre>

1.37.8. Frontend::Module###AgentLinkObject

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentLinkObject'} = { 'Description' => 'Link Object', 'NavBarName' => '', 'Title' => 'Link Object' };</pre>

1.37.9. Frontend::Module###AgentInfo

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0

Description	Value
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentInfo'} = { 'Description' => 'Generic Info module', 'NavBarName' => '', 'Title' => 'Info' };</pre>

1.37.10. Frontend::Module###AgentSearch

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentSearch'} = { 'Description' => 'Global Search Module', 'NavBarName' => '', 'Title' => 'Search' };</pre>

1.37.11. CustomerFrontend::Module###SpellingInline

Description	Value
Description:	Frontend module registration for the customer interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerFrontend::Module'}->{'SpellingInline'} = { 'Description' => 'Spell checker', 'NavBarName' => '', 'Title' => 'Spell Checker' };</pre>

1.37.12. Frontend::Module###AgentHTMLReference

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentHTMLReference'} = { 'Description' => 'HTML Reference', 'Group' => ['users'], 'GroupRo' => ['users'], 'Loader' => {</pre>

Description	Value
	<pre>'CSS' => ['Core.Agent.HTMLReference.css'], }, 'NavBarName' => '', 'Title' => 'HTML Reference' };</pre>

1.37.13. Frontend::Module###AgentStats

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentStats'} = { 'Description' => 'Stats', 'Group' => ['stats'], 'GroupRo' => ['stats'], 'Loader' => { 'JavaScript' => ['Core.Agent.Stats.js'] }, 'NavBar' => [{ 'AccessKey' => '', 'Block' => 'ItemArea', 'Description' => '', 'Link' => 'Action=AgentStats;Subaction=Overview', 'LinkOption' => '', 'Name' => 'Statistics', 'NavBar' => 'Stats', 'Prio' => '8500', 'Type' => 'Menu' }, { 'AccessKey' => '', 'Block' => '', 'Description' => 'Overview', 'GroupRo' => ['stats'], 'Link' => 'Action=AgentStats;Subaction=Overview', 'LinkOption' => '', 'Name' => 'Overview', 'NavBar' => 'Stats', 'Prio' => '100', 'Type' => '' }], { 'AccessKey' => '', 'Block' => '', 'Description' => 'New', 'Group' => ['stats'], 'Link' => 'Action=AgentStats;Subaction=Add', 'LinkOption' => '', 'Name' => 'New', } }</pre>

Description	Value
	<pre> 'NavBar' => 'Stats', 'Prio' => '200', 'Type' => '' }, { 'AccessKey' => '', 'Block' => '', 'Description' => 'Import', 'Group' => ['stats'], 'Link' => 'Action=AgentStats;Subaction=Import', 'LinkOption' => '', 'Name' => 'Import', 'NavBar' => 'Stats', 'Prio' => '300', 'Type' => '' }], 'NavBarName' => 'Stats', 'Title' => 'Stats' }; </pre>

1.38. Frontend::Agent::NavBarModule

1.38.1. Frontend::NavBarModule###6-CustomerCompany

Description	Value
Description:	Frontend module registration (disable company link if no company feature is used).
Group:	Framework
SubGroup:	Frontend::Agent::NavBarModule
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Frontend::NavBarModule'}->{'6-CustomerCompany'} = { 'Module' => 'Kernel::Output::HTML::NavBarCustomerCompany' }; </pre>

1.39. Frontend::Agent::Preferences

1.39.1. PreferencesTableValue

Description	Value
Description:	Defines the name of the column to store the data in the preferences table.
Group:	Framework
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	1
Config-Setting:	<pre> \$self->{'PreferencesTableValue'} = 'preferences_value'; </pre>

1.39.2. PreferencesTableUserID

Description	Value
Description:	Defines the name of the column to store the user identifier in the preferences table.

Description	Value
Group:	Framework
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'PreferencesTableUserID'} = 'user_id';</pre>

1.39.3. PreferencesView

Description	Value
Description:	Sets the display order of the different items in the preferences view.
Group:	Framework
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'PreferencesView'} = ['User Profile', 'Email Settings', 'Other Settings'];</pre>

1.39.4. PreferencesGroups###Password

Description	Value
Description:	Defines the config parameters of this item, to be shown in the preferences view.
Group:	Framework
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PreferencesGroups'}->{'Password'} = { 'Active' => '1', 'Area' => 'Agent', 'Column' => 'User Profile', 'Label' => 'Change password', 'Module' => 'Kernel::Output::HTML::PreferencesPassword', 'PasswordMaxLoginFailed' => '0', 'PasswordMin2Characters' => '0', 'PasswordMin2Lower2UpperCharacters' => '0', 'PasswordMinSize' => '0', 'PasswordNeedDigit' => '0', 'PasswordRegExp' => '', 'Prio' => '0500' };</pre>

1.39.5. PreferencesGroups###SpellDict

Description	Value
Description:	Defines the config parameters of this item, to be shown in the preferences system view. Take care to maintain the dictionaries installed in the system in the data section.

Description	Value
Group:	Framework
SubGroup:	Frontend::Agent::Preferences
Valid:	0
Required:	0
Config-Setting:	<pre> \$self->{'PreferencesGroups'}->{'SpellDict'} = { 'Active' => '1', 'Column' => 'User Profile', 'Data' => { 'deutsch' => 'Deutsch', 'english' => 'English' }, 'DataSelected' => 'english', 'Key' => 'Default spelling dictionary', 'Label' => 'Spelling Dictionary', 'Module' => 'Kernel::Output::HTML::PreferencesGeneric', 'PrefKey' => 'UserSpellDict', 'Prio' => '2000' }; </pre>

1.39.6. PreferencesGroups###Comment

Description	Value
Description:	Defines the config parameters of this item, to be shown in the preferences view.
Group:	Framework
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'PreferencesGroups'}->{'Comment'} = { 'Active' => '0', 'Block' => 'Input', 'Column' => 'Other Settings', 'Data' => '\$Env{"UserComment"}', 'Key' => 'Comment', 'Label' => 'Comment', 'Module' => 'Kernel::Output::HTML::PreferencesGeneric', 'PrefKey' => 'UserComment', 'Prio' => '6000' }; </pre>

1.39.7. PreferencesGroups###Language

Description	Value
Description:	Defines the config parameters of this item, to be shown in the preferences view.
Group:	Framework
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'PreferencesGroups'}->{'Language'} = { 'Active' => '1', 'Column' => 'User Profile', 'Key' => 'Frontend language', 'Label' => 'Language', </pre>

Description	Value
	<pre>'Module' => 'Kernel::Output::HTML::PreferencesLanguage', 'PrefKey' => 'UserLanguage', 'Prio' => '1000' };</pre>

1.39.8. PreferencesGroups###Skin

Description	Value
Description:	Defines the config parameters of this item, to be shown in the preferences view.
Group:	Framework
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PreferencesGroups'}->{'Skin'} = { 'Active' => '1', 'Column' => 'User Profile', 'Key' => 'Wear this frontend skin', 'Label' => 'Skin', 'Module' => 'Kernel::Output::HTML::PreferencesSkin', 'PrefKey' => 'UserSkin', 'Prio' => '2000' };</pre>

1.39.9. PreferencesGroups###Theme

Description	Value
Description:	Defines the config parameters of this item, to be shown in the preferences view.
Group:	Framework
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PreferencesGroups'}->{'Theme'} = { 'Active' => '1', 'Column' => 'User Profile', 'Key' => 'Frontend theme', 'Label' => 'Theme', 'Module' => 'Kernel::Output::HTML::PreferencesTheme', 'PrefKey' => 'UserTheme', 'Prio' => '3000' };</pre>

1.39.10. PreferencesGroups###OutOfOffice

Description	Value
Description:	Defines the config parameters of this item, to be shown in the preferences view.
Group:	Framework
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0

Description	Value
Config-Setting:	<pre>\$Self->{'PreferencesGroups'}->{'OutOfOffice'} = { 'Active' => '1', 'Block' => 'OutOfOffice', 'Column' => 'User Profile', 'Key' => '', 'Label' => 'Out Of Office Time', 'Module' => 'Kernel::Output::HTML::PreferencesOutOfOffice', 'PrefKey' => 'UserOutOfOffice', 'Prio' => '4000' };</pre>

1.39.11. PreferencesGroups###TimeZone

Description	Value
Description:	Defines the config parameters of this item, to be shown in the preferences view.
Group:	Framework
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PreferencesGroups'}->{'TimeZone'} = { 'Active' => '1', 'Column' => 'User Profile', 'Key' => 'Time Zone', 'Label' => 'Time Zone', 'Module' => 'Kernel::Output::HTML::PreferencesTimeZone', 'PrefKey' => 'UserTimeZone', 'Prio' => '5000' };</pre>

1.39.12. PreferencesGroups###CSVSeparator

Description	Value
Description:	Gives end users the possibility to override the separator character for CSV files, defined in the translation files.
Group:	Framework
SubGroup:	Frontend::Agent::Preferences
Valid:	0
Required:	0
Config-Setting:	<pre>\$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::PreferencesGeneric', 'PrefKey' => 'UserCSVSeparator',</pre>

Description	Value
	'Prio' => '4000' };

1.40. Frontend::Agent::SearchRouter

1.40.1. Frontend::SearchDefault

Description	Value
Description:	Search backend default router.
Group:	Framework
SubGroup:	Frontend::Agent::SearchRouter
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Frontend::SearchDefault'} = 'Action=AgentTicketSearch;Subaction=AJAX';</code>

1.41. Frontend::Agent::Stats

1.41.1. Stats::SearchPageShown

Description	Value
Description:	Defines the default maximum number of search results shown on the overview page.
Group:	Framework
SubGroup:	Frontend::Agent::Stats
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::SearchPageShown'} = '20';</code>

1.41.2. Stats::DefaultSelectedDynamicObject

Description	Value
Description:	Defines the default selection at the drop down menu for dynamic objects (Form: Common Specification).
Group:	Framework
SubGroup:	Frontend::Agent::Stats
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Stats::DefaultSelectedDynamicObject'} = 'Ticket';</code>

1.41.3. Stats::DefaultSelectedPermissions

Description	Value
Description:	Defines the default selection at the drop down menu for permissions (Form: Common Specification).
Group:	Framework
SubGroup:	Frontend::Agent::Stats

Description	Value
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Stats::DefaultSelectedPermissions'} = ['stats'];</pre>

1.41.4. Stats::DefaultSelectedFormat

Description	Value
Description:	Defines the default selection at the drop down menu for stats format (Form: Common Specification). Please insert the format key (see Stats::Format).
Group:	Framework
SubGroup:	Frontend::Agent::Stats
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Stats::DefaultSelectedFormat'} = ['Print', 'CSV'];</pre>

1.41.5. Stats::SearchLimit

Description	Value
Description:	Defines the search limit for the stats.
Group:	Framework
SubGroup:	Frontend::Agent::Stats
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Stats::SearchLimit'} = '500';</pre>

1.41.6. Stats::Format

Description	Value
Description:	Defines all the possible stats output formats.
Group:	Framework
SubGroup:	Frontend::Agent::Stats
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Stats::Format'} = { 'CSV' => 'CSV', 'GD::Graph::area' => 'graph-area', 'GD::Graph::bars' => 'graph-bars', 'GD::Graph::hbars' => 'graph-hbars', 'GD::Graph::lines' => 'graph-lines', 'GD::Graph::linespoints' => 'graph-lines-points', 'GD::Graph::pie' => 'graph-pie', 'GD::Graph::points' => 'graph-points', 'Print' => 'Print' }</pre>

Description	Value
	};

1.41.7. Stats::GraphSize

Description	Value
Description:	Sets the size of the statistic graph.
Group:	Framework
SubGroup:	Frontend::Agent::Stats
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Stats::GraphSize'} = { '1200x800' => '1200x800', '1600x1200' => '1600x1200', '800x600' => '800x600' };</pre>

1.41.8. Stats::TimeType

Description	Value
Description:	Sets the time type which should be shown.
Group:	Framework
SubGroup:	Frontend::Agent::Stats
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Stats::TimeType'} = 'Extended';</pre>

1.41.9. Stats::ExchangeAxis

Description	Value
Description:	Allows agents to exchange the axis of a stat if they generate one.
Group:	Framework
SubGroup:	Frontend::Agent::Stats
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Stats::ExchangeAxis'} = '0';</pre>

1.41.10. Stats::UseAgentElementInStats

Description	Value
Description:	Allows agents to generate individual-related stats.
Group:	Framework
SubGroup:	Frontend::Agent::Stats
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Stats::UseAgentElementInStats'} = '0';</pre>

1.41.11. Stats::CustomerIDAsMultiSelect

Description	Value
Description:	Shows all the customer identifiers in a multi-select field (not useful if you have a lot of customer identifiers).
Group:	Framework
SubGroup:	Frontend::Agent::Stats
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Stats::CustomerIDAsMultiSelect'} = '1';</pre>

1.42. Frontend::Customer

1.42.1. CustomerHeadline

Description	Value
Description:	The headline shown in the customer interface.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'CustomerHeadline'} = 'Example Company Support';</pre>

1.42.2. CustomerLogo

Description	Value
Description:	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.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerLogo'} = { 'StyleHeight' => '50px', 'StyleRight' => '25px', 'StyleTop' => '2px', 'StyleWidth' => '135px', 'URL' => 'skins/Customer/default/img/logo.png' };</pre>

1.42.3. CustomerPanelUserID

Description	Value
Description:	Defines the user identifier for the customer panel.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	1

Description	Value
Config-Setting:	<code>\$Self->{'CustomerPanelUserID'} = '1';</code>

1.42.4. CustomerGroupSupport

Description	Value
Description:	Activates support for customer groups.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CustomerGroupSupport'} = '0';</code>

1.42.5. CustomerGroupAlwaysGroups

Description	Value
Description:	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).
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CustomerGroupAlwaysGroups'} = ['users'];</code>

1.42.6. CustomerPanelLoginURL

Description	Value
Description:	Defines an alternate login URL for the customer panel..
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'CustomerPanelLoginURL'} = 'http://host.example.com/cgi-bin/login.pl';</code>

1.42.7. CustomerPanelLogoutURL

Description	Value
Description:	Defines an alternate logout URL for the customer panel.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	0
Required:	0

Description	Value
Config-Setting:	<pre>\$Self->{'CustomerPanelLogoutURL'} = 'http://host.example.com/cgi-bin/login.pl';</pre>

1.42.8. Frontend::CustomerUser::Item###1-GoogleMaps

Description	Value
Description:	Defines a customer item, which generates a google maps icon at the end of a customer info block.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::CustomerUser::Item'}->{'1-GoogleMaps'} = { 'Attributes' => 'UserStreet;UserCity;UserCountry;', 'CSS' => 'Core.Agent.CustomerUser.GoogleMaps.css', 'CSSClass' => 'GoogleMaps', 'Module' => 'Kernel::Output::HTML::CustomerUserGeneric', 'Required' => 'UserStreet;UserCity;', 'Target' => '_blank', 'Text' => 'Location', 'URL' => 'http://maps.google.com/maps?z=7&q=' };</pre>

1.42.9. Frontend::CustomerUser::Item###2-Google

Description	Value
Description:	Defines a customer item, which generates a google icon at the end of a customer info block.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::CustomerUser::Item'}->{'2-Google'} = { 'Attributes' => 'UserFirstname;UserLastname;', 'CSS' => 'Core.Agent.CustomerUser.Google.css', 'CSSClass' => 'Google', 'Module' => 'Kernel::Output::HTML::CustomerUserGeneric', 'Required' => 'UserFirstname;UserLastname;', 'Target' => '_blank', 'Text' => 'Google', 'URL' => 'http://google.com/search?q=' };</pre>

1.42.10. Frontend::CustomerUser::Item###2-LinkedIn

Description	Value
Description:	Defines a customer item, which generates a LinkedIn icon at the end of a customer info block.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	0

Description	Value
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::CustomerUser::Item'}->{'2-LinkedIn'} = { 'Attributes' => 'UserFirstname;UserLastname;', 'CSS' => 'Core.Agent.CustomerUser.LinkedIn.css', 'CSSClass' => 'LinkedIn', 'Module' => 'Kernel::Output::HTML::CustomerUserGeneric', 'Required' => 'UserFirstname;UserLastname;', 'Target' => '_blank', 'Text' => 'LinkedIn', 'URL' => 'http://www.linkedin.com/commonSearch? type=people&keywords=' };</pre>

1.42.11. Frontend::CustomerUser::Item###3-XING

Description	Value
Description:	Defines a customer item, which generates a XING icon at the end of a customer info block.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::CustomerUser::Item'}->{'3-XING'} = { 'Attributes' => 'UserFirstname;UserLastname;', 'CSS' => 'Core.Agent.CustomerUser.Xing.css', 'CSSClass' => 'Xing', 'Module' => 'Kernel::Output::HTML::CustomerUserGeneric', 'Required' => 'UserFirstname;UserLastname;', 'Target' => '_blank', 'Text' => 'XING', 'URL' => 'https://www.xing.com/app/search?op=search;keywords=' };</pre>

1.42.12. CustomerPanelPreApplicationModule###CustomerAccept

Description	Value
Description:	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.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerPanelPreApplicationModule'}->{'CustomerAccept'} = 'Kernel::Modules::CustomerAccept';</pre>

1.42.13. CustomerPanel::InfoKey

Description	Value
Description:	Defines the key to check with CustomerAccept. If this user preferences key is true, then the message is accepted by the system.

Description	Value
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'CustomerPanel::InfoKey'} = 'CustomerAccept1';</code>

1.42.14. CustomerPanel::InfoFile

Description	Value
Description:	Defines the path of the shown info file, that is located under Kernel/Output/HTML/Standard/CustomerAccept.dtl.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'CustomerPanel::InfoFile'} = 'CustomerAccept';</code>

1.42.15. CustomerPanelLostPassword

Description	Value
Description:	Activates lost password feature for customers.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CustomerPanelLostPassword'} = '1';</code>

1.42.16. CustomerPanelCreateAccount

Description	Value
Description:	Enables customers to create their own accounts.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CustomerPanelCreateAccount'} = '1';</code>

1.42.17. CustomerPanelSubjectLostPasswordToken

Description	Value
Description:	Defines the subject for notification mails sent to customers, with token about new requested password.
Group:	Framework
SubGroup:	Frontend::Customer

Description	Value
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CustomerPanelSubjectLostPasswordToken'} = 'New OTRS password request';</code>

1.42.18. CustomerPanelBodyLostPasswordToken

Description	Value
Description:	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).
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<pre> \$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. '; </pre>

1.42.19. CustomerPanelSubjectLostPassword

Description	Value
Description:	Defines the subject for notification mails sent to customers, about new password.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CustomerPanelSubjectLostPassword'} = 'New OTRS password';</code>

1.42.20. CustomerPanelBodyLostPassword

Description	Value
Description:	Defines the body text for notification mails sent to customers, about new password (after using this link the new password will be sent).
Group:	Framework
SubGroup:	Frontend::Customer

Description	Value
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'CustomerPanelBodyLostPassword'} = 'Hi <OTRS_USERFIRSTNAME>, New password: <OTRS_NEWPW> <OTRS_CONFIG_HttpType>://<OTRS_CONFIG_FQDN>/ <OTRS_CONFIG_ScriptAlias>customer.pl '; </pre>

1.42.21. CustomerPanelSubjectNewAccount

Description	Value
Description:	Defines the subject for notification mails sent to customers, about new account.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'CustomerPanelSubjectNewAccount'} = 'New OTRS Account!'; </pre>

1.42.22. CustomerPanelBodyNewAccount

Description	Value
Description:	Defines the body text for notification mails sent to customers, about new account.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<pre> \$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_HttpType>://<OTRS_CONFIG_FQDN>/ <OTRS_CONFIG_ScriptAlias>customer.pl '; </pre>

1.42.23. Loader::Customer::Skin###000-default

Description	Value
Description:	Default skin for OTRS 3.0 interface.

Description	Value
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Loader::Customer::Skin'}->{'000-default'} = { 'Description' => 'This is the default orange - black skin for OTRS 3.0.', 'HomePage' => 'www.otrs.org', 'InternalName' => 'default', 'VisibleName' => 'Default' };</pre>

1.42.24. Loader::Customer::SelectedSkin

Description	Value
Description:	The customer skin's InternalName which should be used in the customer interface. Please check the available skins in Frontend::Customer::Skins.
Group:	Framework
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Loader::Customer::SelectedSkin'} = 'default';</pre>

1.43. Frontend::Customer::Auth

1.43.1. Customer::AuthModule

Description	Value
Description:	Defines the module to authenticate customers.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Customer::AuthModule'} = 'Kernel::System::CustomerAuth::DB';</pre>

1.43.2. Customer::AuthModule::DB::CryptType

Description	Value
Description:	If "DB" was selected for Customer::AuthModule, the crypt type of passwords must be specified.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Customer::AuthModule::DB::CryptType'} = 'md5';</pre>

1.43.3. Customer::AuthModule::DB::Table

Description	Value
Description:	If "DB" was selected for Customer::AuthModule, the name of the table where your customer data should be stored must be specified.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Customer::AuthModule::DB::Table'} = 'customer_user';</code>

1.43.4. Customer::AuthModule::DB::CustomerKey

Description	Value
Description:	If "DB" was selected for Customer::AuthModule, the name of the column for the CustomerKey in the customer table must be specified.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Customer::AuthModule::DB::CustomerKey'} = 'login';</code>

1.43.5. Customer::AuthModule::DB::CustomerPassword

Description	Value
Description:	If "DB" was selected for Customer::AuthModule, the column name for the CustomerPassword in the customer table must be specified.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Customer::AuthModule::DB::CustomerPassword'} = 'pw';</code>

1.43.6. Customer::AuthModule::DB::DSN

Description	Value
Description:	If "DB" was selected for Customer::AuthModule, the DSN for the connection to the customer table must be specified.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Customer::AuthModule::DB::DSN'} = 'DBI:mysql:database=customerdb;host=customerdbhost';</code>

1.43.7. Customer::AuthModule::DB::User

Description	Value
Description:	If "DB" was selected for Customer::AuthModule, a username to connect to the customer table can be specified.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Customer::AuthModule::DB::User'} = 'some_user';</pre>

1.43.8. Customer::AuthModule::DB::Password

Description	Value
Description:	If "DB" was selected for Customer::AuthModule, a password to connect to the customer table can be specified.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Customer::AuthModule::DB::Password'} = 'some_password';</pre>

1.43.9. Customer::AuthModule::DB::Type

Description	Value
Description:	If "DB" was selected for Customer::AuthModule, a database driver (normally autodetection is used) can be specified.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Customer::AuthModule::DB::Type'} = 'mysql';</pre>

1.43.10. Customer::AuthModule::HTTPBasicAuth::Replace

Description	Value
Description:	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).
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Customer::AuthModule::HTTPBasicAuth::Replace'} = 'example_domain\\';</pre>

1.43.11. Customer::AuthModule::HTTPBasicAuth::ReplaceRegExp

Description	Value
Description:	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.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Customer::AuthModule::HTTPBasicAuth::ReplaceRegExp'} = '^(.+?)@.+?\$';</pre>

1.43.12. Customer::AuthModule::LDAP::Host

Description	Value
Description:	If "LDAP" was selected for Customer::AuthModule, the LDAP host can be specified.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Customer::AuthModule::LDAP::Host'} = 'ldap.example.com';</pre>

1.43.13. Customer::AuthModule::LDAP::BaseDN

Description	Value
Description:	If "LDAP" was selected for Customer::AuthModule, the BaseDN must be specified.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Customer::AuthModule::LDAP::BaseDN'} = 'dc=example,dc=com';</pre>

1.43.14. Customer::AuthModule::LDAP::UID

Description	Value
Description:	If "LDAP" was selected for Customer::AuthModule, the user identifier must be specified.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	

Description	Value
	<code>\$Self->{'Customer::AuthModule::LDAP::UID'} = 'uid';</code>

1.43.15. Customer::AuthModule::LDAP::GroupDN

Description	Value
Description:	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.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Customer::AuthModule::LDAP::GroupDN'} = 'cn=otrsallow,ou=posixGroups,dc=example,dc=com';</code>

1.43.16. Customer::AuthModule::LDAP::AccessAttr

Description	Value
Description:	If "LDAP" was selected for Customer::AuthModule, you can specify access attributes here.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Customer::AuthModule::LDAP::AccessAttr'} = 'memberUid';</code>

1.43.17. Customer::AuthModule::LDAP::UserAttr

Description	Value
Description:	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.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Customer::AuthModule::LDAP::UserAttr'} = 'UID';</code>

1.43.18. Customer::AuthModule::LDAP::SearchUserDN

Description	Value
Description:	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.

Description	Value
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Customer::AuthModule::LDAP::SearchUserDN'} = 'cn=binduser,ou=users,dc=example,dc=com';</pre>

1.43.19. Customer::AuthModule::LDAP::SearchUserPw

Description	Value
Description:	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.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Customer::AuthModule::LDAP::SearchUserPw'} = 'some_password';</pre>

1.43.20. Customer::AuthModule::LDAP::AlwaysFilter

Description	Value
Description:	If "LDAP" was selected, you can add a filter to each LDAP query, e.g. (mail=*), (objectclass=user) or (!objectclass=computer).
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Customer::AuthModule::LDAP::AlwaysFilter'} = '(!objectclass=computer)';</pre>

1.43.21. Customer::AuthModule::LDAP::UserSuffix

Description	Value
Description:	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.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Customer::AuthModule::LDAP::UserSuffix'} = '@domain.com';</pre>

1.43.22. Customer::AuthModule::LDAP::Params

Description	Value
Description:	If "LDAP" was selected for Customer::AuthModule and special paramaters are needed for the Net::LDAP perl module, you can specify them here. See "perldoc Net::LDAP" for more information about the parameters.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Customer::AuthModule::LDAP::Params'} = { 'async' => '0', 'port' => '389', 'timeout' => '120', 'version' => '3' };</pre>

1.43.23. Customer::AuthModule::LDAP::Die

Description	Value
Description:	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.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Customer::AuthModule::LDAP::Die'} = '1';</pre>

1.43.24. Customer::AuthModule::Radius::Host

Description	Value
Description:	If "Radius" was selected for Customer::AuthModule, the radius host must be specified.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Customer::AuthModule::Radius::Host'} = 'radiushost';</pre>

1.43.25. Customer::AuthModule::Radius::Password

Description	Value
Description:	If "Radius" was selected for Customer::AuthModule, the password to authenticate to the radius host must be specified.
Group:	Framework
SubGroup:	Frontend::Customer::Auth

Description	Value
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Customer::AuthModule::Radius::Password'} = 'radiussecret';</pre>

1.43.26. Customer::AuthModule::Radius::Die

Description	Value
Description:	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.
Group:	Framework
SubGroup:	Frontend::Customer::Auth
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Customer::AuthModule::Radius::Die'} = '1';</pre>

1.44. Frontend::Customer::ModuleMetaHead

1.44.1. CustomerFrontend::HeaderMetaModule###1-Refresh

Description	Value
Description:	Defines the module to generate html refresh headers of html sites, in the customer interface.
Group:	Framework
SubGroup:	Frontend::Customer::ModuleMetaHead
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'CustomerFrontend::HeaderMetaModule'}->{'1-Refresh'} = { 'Module' => 'Kernel::Output::HTML::HeaderMetaRefresh' };</pre>

1.45. Frontend::Customer::ModuleNotify

1.45.1. CustomerFrontend::NotifyModule###1-ShowAgentOnline

Description	Value
Description:	Defines the module that shows the currently logged in agents in the customer interface.
Group:	Framework
SubGroup:	Frontend::Customer::ModuleNotify
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerFrontend::NotifyModule'}->{'1-ShowAgentOnline'} = { 'IdleMinutes' => '60', 'Module' => 'Kernel::Output::HTML::NotificationAgentOnline', 'ShowEmail' => '1' };</pre>

Description	Value
	};

1.45.2. CustomerFrontend::NotifyModule###1-ShowCustomerOnline

Description	Value
Description:	Defines the module that shows the currently logged in customers in the customer interface.
Group:	Framework
SubGroup:	Frontend::Customer::ModuleNotify
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerFrontend::NotifyModule'}->{'1-ShowCustomerOnline'} = { 'Module' => 'Kernel::Output::HTML::NotificationCustomerOnline', 'ShowEmail' => '1' };</pre>

1.46. Frontend::Customer::ModuleRegistration

1.46.1. CustomerFrontend::Module###Logout

Description	Value
Description:	Frontend module registration for the customer interface.
Group:	Framework
SubGroup:	Frontend::Customer::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerFrontend::Module'}->{'Logout'} = { 'Description' => 'Logout of customer panel', 'NavBarName' => '', 'Title' => '' };</pre>

1.46.2. CustomerFrontend::Module###CustomerPreferences

Description	Value
Description:	Frontend module registration for the customer interface.
Group:	Framework
SubGroup:	Frontend::Customer::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerFrontend::Module'}->{'CustomerPreferences'} = { 'Description' => 'Customer preferences', 'NavBarName' => '', 'Title' => 'Preferences' };</pre>

1.46.3. CustomerFrontend::Module###CustomerAccept

Description	Value
Description:	Frontend module registration for the customer interface.

Description	Value
Group:	Framework
SubGroup:	Frontend::Customer::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerFrontend::Module'}->{'CustomerAccept'} = { 'Description' => 'To accept login information, such as an EULA or license.', 'NavBarName' => '', 'Title' => 'Info' };</pre>

1.46.4. CustomerFrontend::Module###PictureUpload

Description	Value
Description:	Frontend module registration for the customer interface.
Group:	Framework
SubGroup:	Frontend::Customer::ModuleRegistration
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'CustomerFrontend::Module'}->{'PictureUpload'} = { 'Description' => 'Picture upload module', 'NavBarName' => 'Ticket', 'Title' => 'Picture-Upload' };</pre>

1.47. Frontend::Customer::Preferences

1.47.1. PreferencesTable

Description	Value
Description:	Defines the name of the table, where the customer preferences are stored.
Group:	Framework
SubGroup:	Frontend::Customer::Preferences
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'PreferencesTable'} = 'user_preferences';</pre>

1.47.2. PreferencesTableKey

Description	Value
Description:	Defines the column to store the keys for the preferences table.
Group:	Framework
SubGroup:	Frontend::Customer::Preferences
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<code>\$Self->{'PreferencesTableKey'} = 'preferences_key';</code>

1.47.3. CustomerPreferences

Description	Value
Description:	Defines the parameters for the customer preferences table.
Group:	Framework
SubGroup:	Frontend::Customer::Preferences
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'CustomerPreferences'} = { 'Module' => 'Kernel::System::CustomerUser::Preferences::DB', 'Params' => { 'Table' => 'customer_preferences', 'TableKey' => 'preferences_key', 'TableUserID' => 'user_id', 'TableValue' => 'preferences_value' } }; </pre>

1.47.4. CustomerPreferencesView

Description	Value
Description:	Sets the order of the different items in the customer preferences view.
Group:	Framework
SubGroup:	Frontend::Customer::Preferences
Valid:	1
Required:	1
Config-Setting:	<pre> \$Self->{'CustomerPreferencesView'} = ['User Profile', 'Other Settings']; </pre>

1.47.5. CustomerPreferencesGroups###Password

Description	Value
Description:	Defines all the parameters for this item in the customer preferences.
Group:	Framework
SubGroup:	Frontend::Customer::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'CustomerPreferencesGroups'}->{'Password'} = { 'Active' => '1', 'Area' => 'Customer', 'Column' => 'Other Settings', 'Label' => 'Change password', 'Module' => 'Kernel::Output::HTML::PreferencesPassword', 'PasswordMin2Characters' => '0', 'PasswordMin2Lower2UpperCharacters' => '0', }; </pre>

Description	Value
	<pre>'PasswordMinSize' => '0', 'PasswordNeedDigit' => '0', 'PasswordRegExp' => '', 'Prio' => '1000' };</pre>

1.47.6. CustomerPreferencesGroups###Language

Description	Value
Description:	Defines all the parameters for this item in the customer preferences.
Group:	Framework
SubGroup:	Frontend::Customer::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerPreferencesGroups'}->{'Language'} = { 'Active' => '1', 'Column' => 'User Profile', 'Key' => 'Your language', 'Label' => 'Interface language', 'Module' => 'Kernel::Output::HTML::PreferencesLanguage', 'PrefKey' => 'UserLanguage', 'Prio' => '2000' };</pre>

1.47.7. CustomerPreferencesGroups###Theme

Description	Value
Description:	Defines all the parameters for this item in the customer preferences.
Group:	Framework
SubGroup:	Frontend::Customer::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerPreferencesGroups'}->{'Theme'} = { 'Active' => '0', 'Column' => 'User Profile', 'Key' => 'Select your frontend Theme.', 'Label' => 'Theme', 'Module' => 'Kernel::Output::HTML::PreferencesTheme', 'PrefKey' => 'UserTheme', 'Prio' => '1000' };</pre>

1.47.8. CustomerPreferencesGroups###TimeZone

Description	Value
Description:	Defines all the parameters for this item in the customer preferences.
Group:	Framework
SubGroup:	Frontend::Customer::Preferences
Valid:	1
Required:	0

Description	Value
Config-Setting:	<pre>\$Self->{'CustomerPreferencesGroups'}->{'TimeZone'} = { 'Active' => '1', 'Column' => 'User Profile', 'Key' => 'Time Zone', 'Label' => 'Time Zone', 'Module' => 'Kernel::Output::HTML::PreferencesTimeZone', 'PrefKey' => 'UserTimeZone', 'Prio' => '5000' };</pre>

1.47.9. CustomerPreferencesGroups###PGP

Description	Value
Description:	Defines all the parameters for this item in the customer preferences.
Group:	Framework
SubGroup:	Frontend::Customer::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerPreferencesGroups'}->{'PGP'} = { 'Active' => '1', 'Column' => 'Other Settings', 'Key' => 'PGP Key Upload', 'Label' => 'PGP Key', 'Module' => 'Kernel::Output::HTML::PreferencesPGP', 'PrefKey' => 'UserPGPKey', 'Prio' => '10000' };</pre>

1.47.10. CustomerPreferencesGroups###SMIME

Description	Value
Description:	Defines all the parameters for this item in the customer preferences.
Group:	Framework
SubGroup:	Frontend::Customer::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerPreferencesGroups'}->{'SMIME'} = { 'Active' => '1', 'Column' => 'Other Settings', 'Key' => 'S/MIME Certificate Upload', 'Label' => 'S/MIME Certificate', 'Module' => 'Kernel::Output::HTML::PreferencesSMIME', 'PrefKey' => 'UserSMIMEKey', 'Prio' => '11000' };</pre>

1.48. Frontend::Public

1.48.1. PublicFrontend::CommonParam###Action

Description	Value
Description:	Defines the default value for the action parameter for the public frontend. The action parameter is used in the scripts of the system.

Description	Value
Group:	Framework
SubGroup:	Frontend::Public
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'PublicFrontend::CommonParam'}->{'Action'} = 'PublicDefault';</pre>

1.49. Frontend::Public::ModuleRegistration

1.49.1. PublicFrontend::Module###PublicDefault

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Public::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PublicFrontend::Module'}->{'PublicDefault'} = { 'Description' => 'PublicDefault', 'NavBarName' => '', 'Title' => 'PublicDefault' };</pre>

1.49.2. PublicFrontend::Module###PublicRepository

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Framework
SubGroup:	Frontend::Public::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PublicFrontend::Module'}->{'PublicRepository'} = { 'Description' => 'PublicRepository', 'NavBarName' => '', 'Title' => 'PublicRepository' };</pre>

2. Ticket

2.1. Core

2.1.1. OTRSEscalationEvents::DecayTime

Description	Value
Description:	The duration in minutes after emitting an event, in which the new escalation notify and start events are suppressed.
Group:	Ticket
SubGroup:	Core
Valid:	1

Description	Value
Required:	0
Config-Setting:	<code>\$Self->{'OTRSEscalationEvents::DecayTime'} = '1440';</code>

2.2. Core::FulltextSearch

2.2.1. Ticket::SearchIndexModule

Description	Value
Description:	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.RebuildFulltextIndex.pl".
Group:	Ticket
SubGroup:	Core::FulltextSearch
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::SearchIndexModule'} = 'Kernel::System::Ticket::ArticleSearchIndex::RuntimeDB';</code>

2.2.2. Ticket::SearchIndex::Attribute

Description	Value
Description:	Configures the full-text index. Execute "bin/otrs.RebuildFulltextIndex.pl" in order to generate a new index.
Group:	Ticket
SubGroup:	Core::FulltextSearch
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::SearchIndex::Attribute'} = { 'WordCountMax' => '1000', 'WordLengthMax' => '30', 'WordLengthMin' => '3' };</code>

2.2.3. Ticket::EventModulePost###98-ArticleSearchIndex

Description	Value
Description:	Builds an article index right after the article's creation.
Group:	Ticket
SubGroup:	Core::FulltextSearch
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::EventModulePost'}->{'98-ArticleSearchIndex'} = { 'Event' => '(ArticleCreate ArticleUpdate)', 'Module' => 'Kernel::System::Ticket::Event::ArticleSearchIndex'</code>

Description	Value
	};

2.3. Core::LinkObject

2.3.1. LinkObject::PossibleLink###0200

Description	Value
Description:	Links 2 tickets with a "Normal" type link.
Group:	Ticket
SubGroup:	Core::LinkObject
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'LinkObject::PossibleLink'}->{'0200'} = { 'Object1' => 'Ticket', 'Object2' => 'Ticket', 'Type' => 'Normal' };</pre>

2.3.2. LinkObject::PossibleLink###0201

Description	Value
Description:	Links 2 tickets with a "ParentChild" type link.
Group:	Ticket
SubGroup:	Core::LinkObject
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'LinkObject::PossibleLink'}->{'0201'} = { 'Object1' => 'Ticket', 'Object2' => 'Ticket', 'Type' => 'ParentChild' };</pre>

2.4. Core::PostMaster

2.4.1. PostmasterMaxEmails

Description	Value
Description:	Maximal auto email responses to own email-address a day (Loop-Protection).
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'PostmasterMaxEmails'} = '40';</pre>

2.4.2. PostMasterMaxEmailSize

Description	Value
Description:	Maximal size in KBytes for mails that can be fetched via POP3/POP3S/IMAP/IMAPS (KBytes).

Description	Value
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PostMasterMaxEmailSize'} = '16384';</code>

2.4.3. PostMasterReconnectMessage

Description	Value
Description:	The "bin/PostMasterMailAccount.pl" will reconnect to POP3/POP3S/IMAP/IMAPS host after the specified count of messages.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PostMasterReconnectMessage'} = '20';</code>

2.4.4. LoopProtectionModule

Description	Value
Description:	Default loop protection module.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'LoopProtectionModule'} = 'Kernel::System::PostMaster::LoopProtection::DB';</code>

2.4.5. LoopProtectionLog

Description	Value
Description:	Path for the log file (it only applies if "FS" was selected for LoopProtectionModule and it is mandatory).
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'LoopProtectionLog'} = '<OTRS_CONFIG_Home>/var/log/LoopProtection';</code>

2.4.6. PostmasterAutoHTML2Text

Description	Value
Description:	Converts HTML mails into text messages.
Group:	Ticket

Description	Value
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PostmasterAutoHTML2Text'} = '1';</code>

2.4.7. PostmasterFollowUpSearchInReferences

Description	Value
Description:	Executes follow up checks on In-Reply-To or References headers for mails that don't have a ticket number in the subject.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PostmasterFollowUpSearchInReferences'} = '0';</code>

2.4.8. PostmasterFollowUpSearchInBody

Description	Value
Description:	Executes follow up mail body checks in mails that don't have a ticket number in the subject.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PostmasterFollowUpSearchInBody'} = '0';</code>

2.4.9. PostmasterFollowUpSearchInAttachment

Description	Value
Description:	Executes follow up mail attachments checks in mails that don't have a ticket number in the subject.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PostmasterFollowUpSearchInAttachment'} = '0';</code>

2.4.10. PostmasterFollowUpSearchInRaw

Description	Value
Description:	Executes follow up plain/raw mail checks in mails that don't have a ticket number in the subject.
Group:	Ticket

Description	Value
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PostmasterFollowUpSearchInRaw'} = '0';</code>

2.4.11. PostmasterUserID

Description	Value
Description:	Specifies user id of the postmaster data base.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PostmasterUserID'} = '1';</code>

2.4.12. PostmasterDefaultQueue

Description	Value
Description:	Defines the postmaster default queue.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PostmasterDefaultQueue'} = 'Raw';</code>

2.4.13. PostmasterDefaultPriority

Description	Value
Description:	Defines the default priority of new tickets.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PostmasterDefaultPriority'} = '3 normal';</code>

2.4.14. PostmasterDefaultState

Description	Value
Description:	Defines the default state of new tickets.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1

Description	Value
Config-Setting:	<code>\$Self->{'PostmasterDefaultState'} = 'new';</code>

2.4.15. PostmasterFollowUpState

Description	Value
Description:	Defines the state of a ticket if it gets a follow-up.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PostmasterFollowUpState'} = 'open';</code>

2.4.16. PostmasterFollowUpStateClosed

Description	Value
Description:	Defines the state of a ticket if it gets a follow-up and the ticket was already closed.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'PostmasterFollowUpStateClosed'} = 'open';</code>

2.4.17. PostmasterFollowUpOnUnlockAgentNotifyOnlyToOwner

Description	Value
Description:	Sends agent follow-up notification only to the owner, if a ticket is unlocked (the default is to send the notification to all agents).
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PostmasterFollowUpOnUnlockAgentNotifyOnlyToOwner'} = '0';</code>

2.4.18. PostmasterX-Header

Description	Value
Description:	Defines all the X-headers that should be scanned.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'PostmasterX-Header'} = [</code>

Description	Value
	<pre> '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-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', 'Return-Path', 'X-OTRS-Loop', 'X-OTRS-Info', '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']; </pre>

2.4.19. PostMaster::PreFilterModule###1-Match

Description	Value
Description:	Module to filter and manipulate incoming messages. Block/ignore all spam email with From: noreply@ address.
Group:	Ticket
SubGroup:	Core::PostMaster

Description	Value
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'PostMaster::PreFilterModule'}->{'1-Match'} = { 'Match' => { 'From' => 'noreply@' }, 'Module' => 'Kernel::System::PostMaster::Filter::Match', 'Set' => { 'X-OTRS-Ignore' => 'yes' }, 'StopAfterMatch' => '0' };</pre>

2.4.20. PostMaster::PreFilterModule###2-Match

Description	Value
Description:	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 =>.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	0
Required:	0
Config-Setting:	<pre>\$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' => '[***]' }, 'StopAfterMatch' => '0' };</pre>

2.4.21. PostMaster::PreFilterModule###3-NewTicketReject

Description	Value
Description:	Blocks all the incoming emails that do not have a valid ticket number in subject with From: @example.com address.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'PostMaster::PreFilterModule'}->{'3-NewTicketReject'} = { 'Match' => { 'From' => '@example.com' }, 'Module' => 'Kernel::System::PostMaster::Filter::NewTicketReject', 'Set' => { 'X-OTRS-Ignore' => 'yes' }, 'StopAfterMatch' => '0' };</pre>

2.4.22. PostMaster::PreFilterModule::NewTicketReject::Sender

Description	Value
Description:	Defines the sender for rejected emails.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'PostMaster::PreFilterModule::NewTicketReject::Sender'} = 'noreply@example.com';</pre>

2.4.23. PostMaster::PreFilterModule::NewTicketReject::Subject

Description	Value
Description:	Defines the subject for rejected emails.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'PostMaster::PreFilterModule::NewTicketReject::Subject'} = 'Email Rejected';</pre>

2.4.24. PostMaster::PreFilterModule::NewTicketReject::Body

Description	Value
Description:	Defines the body text for rejected emails.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<pre>\$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 ';</pre>

2.4.25. PostMaster::PreFilterModule###4-CMD

Description	Value
Description:	CMD example setup. Ignores emails where external CMD returns some output on STDOUT (email will be piped into STDIN of some.bin).
Group:	Ticket

Description	Value
SubGroup:	Core::PostMaster
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'PostMaster::PreFilterModule'}->{'4-CMD'} = { 'CMD' => '/usr/bin/some.bin', 'Module' => 'Kernel::System::PostMaster::Filter::CMD', 'Set' => { 'X-OTRS-Ignore' => 'yes' } };</pre>

2.4.26. PostMaster::PreFilterModule###5-SpamAssassin

Description	Value
Description:	Spam Assassin example setup. Ignores emails that are marked with SpamAssassin.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	0
Required:	0
Config-Setting:	<pre>\$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' } };</pre>

2.4.27. PostMaster::PreFilterModule###6-SpamAssassin

Description	Value
Description:	Spam Assassin example setup. Moves marked mails to spam queue.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	0
Required:	0
Config-Setting:	<pre>\$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' } };</pre>

2.4.28. PostMaster::PreFilterModule###000-MatchDBSource

Description	Value
Description:	Module to use database filter storage.
Group:	Ticket
SubGroup:	Core::PostMaster

Description	Value
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'PostMaster::PreFilterModule'}->{'000-MatchDBSource'} = { 'Module' => 'Kernel::System::PostMaster::Filter::MatchDBSource' };</pre>

2.4.29. PostMaster::PostFilterModule###000-FollowUpArticleTypeCheck

Description	Value
Description:	Module to check if arrived emails should be marked as email-internal (because of original forwarded internal email it college). ArticleType and SenderType define the values for the arrived email/article.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PostMaster::PostFilterModule'}->{'000-FollowUpArticleTypeCheck'} = { 'ArticleType' => 'email-internal', 'Module' => 'Kernel::System::PostMaster::Filter::FollowUpArticleTypeCheck', 'SenderType' => 'customer' };</pre>

2.4.30. SendNoAutoResponseRegExp

Description	Value
Description:	If this regex matches, no message will be send by the autoresponder.
Group:	Ticket
SubGroup:	Core::PostMaster
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'SendNoAutoResponseRegExp'} = '(MAILER-DAEMON postmaster abuse)@.+?\.\.+?';</pre>

2.5. Core::Stats

2.5.1. Stats::DynamicObjectRegistration###Ticket

Description	Value
Description:	Module to generate ticket statistics.
Group:	Ticket
SubGroup:	Core::Stats
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Stats::DynamicObjectRegistration'}->{'Ticket'} = {</pre>

Description	Value
	'Module' => 'Kernel::System::Stats::Dynamic::Ticket' };

2.5.2. Stats::DynamicObjectRegistration###TicketList

Description	Value
Description:	Determines if the statistics module may generate ticket lists.
Group:	Ticket
SubGroup:	Core::Stats
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Stats::DynamicObjectRegistration'}->{'TicketList'} = { 'Module' => 'Kernel::System::Stats::Dynamic::TicketList' };</pre>

2.5.3. Stats::DynamicObjectRegistration###TicketAccountedTime

Description	Value
Description:	Module to generate accounted time ticket statistics.
Group:	Ticket
SubGroup:	Core::Stats
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Stats::DynamicObjectRegistration'}-> {'TicketAccountedTime'} = { 'Module' => 'Kernel::System::Stats::Dynamic::TicketAccountedTime' };</pre>

2.5.4. Stats::DynamicObjectRegistration###TicketSolutionResponseTime

Description	Value
Description:	Module to generate ticket solution and response time statistics.
Group:	Ticket
SubGroup:	Core::Stats
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Stats::DynamicObjectRegistration'}-> {'TicketSolutionResponseTime'} = { 'Module' => 'Kernel::System::Stats::Dynamic::TicketSolutionResponseTime' };</pre>

2.6. Core::Ticket

2.6.1. Ticket::Hook

Description	Value
Description:	The identifier for a ticket, e.g. Ticket#, Call#, MyTicket#. The default is Ticket#.

Description	Value
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Hook'} = 'Ticket#';</code>

2.6.2. Ticket::HookDivider

Description	Value
Description:	The divider between TicketHook and ticket number. E.g ': '.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::HookDivider'} = '';</code>

2.6.3. Ticket::SubjectSize

Description	Value
Description:	Max size of the subjects in an email reply.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::SubjectSize'} = '100';</code>

2.6.4. Ticket::SubjectRe

Description	Value
Description:	The text at the beginning of the subject in an email reply, e.g. RE, AW, or AS.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::SubjectRe'} = 'Re';</code>

2.6.5. Ticket::SubjectFwd

Description	Value
Description:	The text at the beginning of the subject when an email is forwarded, e.g. FW, Fwd, or WG.

Description	Value
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::SubjectFwd'} = 'Fwd';</code>

2.6.6. Ticket::SubjectFormat

Description	Value
Description:	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 last case you should enable PostmasterFollowupSearchInRaw or PostmasterFollowUpSearchInReferences to recognize followups based on email headers and/or body.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::SubjectFormat'} = 'Left';</code>

2.6.7. Ticket::CustomQueue

Description	Value
Description:	Name of custom queue. The custom queue is a queue selection of your preferred queues and can be selected in the preferences settings.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::CustomQueue'} = 'My Queues';</code>

2.6.8. Ticket::NewArticleIgnoreSystemSender

Description	Value
Description:	Ignore article with system sender type for new article feature (e. g. auto responses or email notifications).
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<code>\$Self->{'Ticket::NewArticleIgnoreSystemSender'} = '0';</code>

2.6.9. Ticket::ChangeOwnerToEveryone

Description	Value
Description:	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.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::ChangeOwnerToEveryone'} = '0';</code>

2.6.10. Ticket::Responsible

Description	Value
Description:	Enables ticket responsible feature, to keep track of a specific ticket.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Responsible'} = '0';</code>

2.6.11. Ticket::ResponsibleAutoSet

Description	Value
Description:	Automatically sets the owner of a ticket as the responsible for it (if ticket responsible feature is enabled).
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::ResponsibleAutoSet'} = '1';</code>

2.6.12. Ticket::Type

Description	Value
Description:	Allows defining new types for ticket (if ticket type feature is enabled).
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1

Description	Value
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Type'} = '0';</code>

2.6.13. Ticket::Service

Description	Value
Description:	Allows defining services and SLAs for tickets (e. g. email, desktop, network, ...), and escalation attributes for SLAs (if ticket service/SLA feature is enabled).
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Service'} = '0';</code>

2.6.14. Ticket::Service::Default::UnknownCustomer

Description	Value
Description:	Allows default services to be selected also for non existing customers.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Service::Default::UnknownCustomer'} = '0';</code>

2.6.15. Ticket::ArchiveSystem

Description	Value
Description:	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.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::ArchiveSystem'} = '0';</code>

2.6.16. Ticket::CustomerArchiveSystem

Description	Value
Description:	Activates the ticket archive system search in the customer interface.
Group:	Ticket

Description	Value
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::CustomerArchiveSystem'} = '0';</code>

2.6.17. Ticket::NumberGenerator

Description	Value
Description:	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).
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::NumberGenerator'} = 'Kernel::System::Ticket::Number::DateChecksum';</code>

2.6.18. Ticket::NumberGenerator::MinCounterSize

Description	Value
Description:	Sets the minimal ticket counter size (if "AutoIncrement" was selected as TicketNumberGenerator). Default is 5, this means the counter starts from 10000.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::NumberGenerator::MinCounterSize'} = '5';</code>

2.6.19. Ticket::NumberGenerator::CheckSystemID

Description	Value
Description:	Checks the SystemID in ticket number detection for follow-ups (use "No" if SystemID has been changed after using the system).
Group:	Ticket
SubGroup:	Core::Ticket

Description	Value
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::NumberGenerator::CheckSystemID'} = '1';</code>

2.6.20. Ticket::CounterLog

Description	Value
Description:	Log file for the ticket counter.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::CounterLog'} = '<OTRS_CONFIG_Home>/var/log/TicketCounter.log';</code>

2.6.21. Ticket::IndexModule

Description	Value
Description:	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 script "bin/otrs.RebuildTicketIndex.pl" for initial index update.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::IndexModule'} = 'Kernel::System::Ticket::IndexAccelerator::RuntimeDB';</code>

2.6.22. Ticket::StorageModule

Description	Value
Description:	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.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<code>\$Self->{'Ticket::StorageModule'} = 'Kernel::System::Ticket::ArticleStorageDB';</code>

2.6.23. ArticleDir

Description	Value
Description:	Specifies the directory to store the data in, if "FS" was selected for TicketStorageModule.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'ArticleDir'} = '<OTRS_CONFIG_Home>/var/article';</code>

2.6.24. Ticket::EventModulePost###100-ArchiveRestore

Description	Value
Description:	Restores a ticket from the archive (only if the event is a state change, from closed to any open available state).
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::EventModulePost'}->{'100-ArchiveRestore'} = { 'Event' => 'TicketStateUpdate', 'Module' => 'Kernel::System::Ticket::Event::ArchiveRestore' };</code>

2.6.25. Ticket::EventModulePost###110-AcceleratorUpdate

Description	Value
Description:	Updates the ticket index accelerator.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::EventModulePost'}->{'110-AcceleratorUpdate'} = { 'Event' => 'TicketStateUpdate TicketQueueUpdate TicketLockUpdate', 'Module' => 'Kernel::System::Ticket::Event::TicketAcceleratorUpdate' };</code>

2.6.26. Ticket::EventModulePost###120-ForceOwnerResetOnMove

Description	Value
Description:	Resets and unlocks the owner of a ticket if it was moved to another queue.
Group:	Ticket

Description	Value
SubGroup:	Core::Ticket
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::EventModulePost'}->{'120-ForceOwnerResetOnMove'} = { 'Event' => 'TicketQueueUpdate', 'Module' => 'Kernel::System::Ticket::Event::ForceOwnerReset' };</pre>

2.6.27. Ticket::EventModulePost###130-ForceStateChangeOnLock

Description	Value
Description:	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.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::EventModulePost'}->{'130-ForceStateChangeOnLock'} = { 'Event' => 'TicketLockUpdate', 'Module' => 'Kernel::System::Ticket::Event::ForceState', 'new' => 'open' };</pre>

2.6.28. Ticket::EventModulePost###140-ResponsibleAutoSet

Description	Value
Description:	Automatically sets the responsible of a ticket (if it is not set yet) after the first owner update.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::EventModulePost'}->{'140-ResponsibleAutoSet'} = { 'Event' => 'TicketOwnerUpdate', 'Module' => 'Kernel::System::Ticket::Event::ResponsibleAutoSet' };</pre>

2.6.29. Ticket::EventModulePost###150-TicketPendingTimeReset

Description	Value
Description:	Sets the PendingTime of a ticket to 0 if the state is changed to a non-pending state.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0

Description	Value
Config-Setting:	<pre>\$Self->{'Ticket::EventModulePost'}->{'150-TicketPendingTimeReset'} = { 'Event' => 'TicketStateUpdate', 'Module' => 'Kernel::System::Ticket::Event::TicketPendingTimeReset' };</pre>

2.6.30. Ticket::EventModulePost###500-NotificationEvent

Description	Value
Description:	Sends the notifications which are configured in the admin interface under "Notification (Event)".
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::EventModulePost'}->{'500-NotificationEvent'} = { 'Event' => '', 'Module' => 'Kernel::System::Ticket::Event::NotificationEvent', 'Transaction' => '1' };</pre>

2.6.31. Ticket::EventModulePost###900-EscalationIndex

Description	Value
Description:	Updates the ticket escalation index after a ticket attribute got updated.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::EventModulePost'}->{'900-EscalationIndex'} = { 'Event' => 'TicketSLAUpdate TicketQueueUpdate TicketStateUpdate TicketCreate ArticleCreate', 'Module' => 'Kernel::System::Ticket::Event::TicketEscalationIndex' };</pre>

2.6.32. Ticket::EventModulePost###900-EscalationStopEvents

Description	Value
Description:	Ticket event module that triggers the escalation stop events.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::EventModulePost'}->{'900-EscalationStopEvents'} = { 'Event' => 'TicketSLAUpdate TicketQueueUpdate TicketStateUpdate ArticleCreate', 'Module' => 'Kernel::System::Ticket::Event::TriggerEscalationStopEvents'</pre>

Description	Value
	};

2.6.33. Ticket::EventModulePost###910-ForceUnlockOnMove

Description	Value
Description:	Forces to unlock tickets after being moved to another queue.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::EventModulePost'}->{'910-ForceUnlockOnMove'} = { 'Event' => 'TicketQueueUpdate', 'Module' => 'Kernel::System::Ticket::Event::ForceUnlock' };</pre>

2.6.34. Ticket::EventModulePost###920-TicketArticleNewMessageUpdate

Description	Value
Description:	Update Ticket "Seen" flag if every article got seen or a new Article got created.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::EventModulePost'}->{'920- TicketArticleNewMessageUpdate'} = { 'Event' => 'ArticleCreate ArticleFlagSet', 'Module' => 'Kernel::System::Ticket::Event::TicketNewMessageUpdate' };</pre>

2.6.35. Ticket::CustomModule###001-CustomModule

Description	Value
Description:	Overloads (redefines) existing functions in Kernel::System::Ticket. Used to easily add customizations.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::CustomModule'}->{'001-CustomModule'} = 'Kernel::System::Ticket::Custom';</pre>

2.6.36. Ticket::ViewableSenderTypes

Description	Value
Description:	Defines the default viewable sender types of a ticket (default: customer).

Description	Value
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::ViewableSenderTypes'} = ['\customer\'];</pre>

2.6.37. Ticket::ViewableLocks

Description	Value
Description:	Defines the viewable locks of a ticket. Default: unlock, tmp_lock.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::ViewableLocks'} = ['\unlock\'', '\tmp_lock\'];</pre>

2.6.38. Ticket::ViewableStateType

Description	Value
Description:	Defines the valid state types for a ticket.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::ViewableStateType'} = ['new', 'open', 'pending reminder', 'pending auto'];</pre>

2.6.39. Ticket::UnlockStateType

Description	Value
Description:	Defines the valid states for unlocked tickets. To unlock tickets the script "bin/otrs.UnlockTickets.pl" can be used.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::UnlockStateType'} = [</pre>

Description	Value
	<code>'new', 'open'];</code>

2.6.40. Ticket::PendingNotificationOnlyToOwner

Description	Value
Description:	Sends reminder notifications of unlocked ticket after reaching the reminder date (only sent to ticket owner).
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::PendingNotificationOnlyToOwner'} = '0';</code>

2.6.41. Ticket::PendingNotificationNotToResponsible

Description	Value
Description:	Disables sending reminder notifications to the responsible agent of a ticket (Ticket::Responsible needs to be activated).
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::PendingNotificationNotToResponsible'} = '0';</code>

2.6.42. Ticket::PendingReminderStateType

Description	Value
Description:	Defines the state type of the reminder for pending tickets.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::PendingReminderStateType'} = ['pending reminder'];</code>

2.6.43. Ticket::PendingAutoStateType

Description	Value
Description:	Determines the possible states for pending tickets that changed state after reaching time limit.
Group:	Ticket
SubGroup:	Core::Ticket

Description	Value
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::PendingAutoStateType'} = ['pending auto'];</pre>

2.6.44. Ticket::StateAfterPending

Description	Value
Description:	Defines which states should be set automatically (Content), after the pending time of state (Key) has been reached.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::StateAfterPending'} = { 'pending auto close+' => 'closed successful', 'pending auto close-' => 'closed unsuccessful' };</pre>

2.6.45. System::Permission

Description	Value
Description:	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.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'System::Permission'} = ['ro', 'move_into', 'create', 'note', 'owner', 'priority', 'rw'];</pre>

2.6.46. Ticket::Permission###1-OwnerCheck

Description	Value
Description:	Module to check the owner of a ticket.
Group:	Ticket

Description	Value
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Permission'}->{'1-OwnerCheck'} = { 'Granted' => '1', 'Module' => 'Kernel::System::Ticket::Permission::OwnerCheck', 'Required' => '0' };</pre>

2.6.47. Ticket::Permission###2-ResponsibleCheck

Description	Value
Description:	Module to check the agent responsible of a ticket.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Permission'}->{'2-ResponsibleCheck'} = { 'Granted' => '1', 'Module' => 'Kernel::System::Ticket::Permission::ResponsibleCheck', 'Required' => '0' };</pre>

2.6.48. Ticket::Permission###3-GroupCheck

Description	Value
Description:	Module to check if a user is in a special group. Access is granted, if the user is in the specified group and has ro and rw permissions.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Permission'}->{'3-GroupCheck'} = { 'Granted' => '1', 'Module' => 'Kernel::System::Ticket::Permission::GroupCheck', 'Required' => '0' };</pre>

2.6.49. Ticket::Permission###4-WatcherCheck

Description	Value
Description:	Module to check the watcher agents of a ticket.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	

Description	Value
	<pre>\$Self->{'Ticket::Permission'}->{'4-WatcherCheck'} = { 'Granted' => '1', 'Module' => 'Kernel::System::Ticket::Permission::WatcherCheck', 'Required' => '0' };</pre>

2.6.50. CustomerTicket::Permission###1-GroupCheck

Description	Value
Description:	Module to check the group permissions for the access to customer tickets.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerTicket::Permission'}->{'1-GroupCheck'} = { 'Granted' => '0', 'Module' => 'Kernel::System::Ticket::CustomerPermission::GroupCheck', 'Required' => '1' };</pre>

2.6.51. CustomerTicket::Permission###2-CustomerUserIDCheck

Description	Value
Description:	Grants access, if the customer ID of the ticket matches the customer user's ID and the customer user has group permissions on the queue the ticket is in.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerTicket::Permission'}->{'2-CustomerUserIDCheck'} = { 'Granted' => '1', 'Module' => 'Kernel::System::Ticket::CustomerPermission::CustomerUserIDCheck', 'Required' => '0' };</pre>

2.6.52. CustomerTicket::Permission###3-CustomerIDCheck

Description	Value
Description:	Module to check customer permissions.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerTicket::Permission'}->{'3-CustomerIDCheck'} = { 'Granted' => '1',</pre>

Description	Value
	<pre>'Module' => 'Kernel::System::Ticket::CustomerPermission::CustomerIDCheck', 'Required' => '0' };</pre>

2.6.53. Ticket::DefineEmailFrom

Description	Value
Description:	Defines how the From field from the emails (sent from answers and email tickets) should look like.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::DefineEmailFrom'} = 'SystemAddressName';</pre>

2.6.54. Ticket::DefineEmailFromSeparator

Description	Value
Description:	Defines the separator between the agents real name and the given queue email address.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::DefineEmailFromSeparator'} = 'via';</pre>

2.6.55. CustomerNotifyJustToRealCustomer

Description	Value
Description:	Sends customer notifications just to the mapped customer. Normally, if no customer is mapped, the latest customer sender gets the notification.
Group:	Ticket
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'CustomerNotifyJustToRealCustomer'} = '0';</pre>

2.6.56. AgentSelfNotifyOnAction

Description	Value
Description:	Specifies if an agent should receive email notification of his own actions.
Group:	Ticket

Description	Value
SubGroup:	Core::Ticket
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'AgentSelfNotifyOnAction'} = '0';</code>

2.7. Core::TicketACL

2.7.1. Ticket::Acl::Module###1-Ticket::Acl::Module

Description	Value
Description:	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).
Group:	Ticket
SubGroup:	Core::TicketACL
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Acl::Module'}->{'1-Ticket::Acl::Module'} = { 'Module' => 'Kernel::System::Ticket::Acl::CloseParentAfterClosedChilds', 'State' => ['closed successful', 'closed unsuccessful'] };</code>

2.7.2. TicketACL::Default::Action

Description	Value
Description:	Default ACL values for ticket actions.
Group:	Ticket
SubGroup:	Core::TicketACL
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'TicketACL::Default::Action'} = {};</code>

2.8. Core::TicketBulkAction

2.8.1. Ticket::Frontend::BulkFeature

Description	Value
Description:	Enables ticket bulk action feature for the agent frontend to work on more than one ticket at a time.
Group:	Ticket
SubGroup:	Core::TicketBulkAction
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<code>\$Self->{'Ticket::Frontend::BulkFeature'} = '1';</code>

2.8.2. Ticket::Frontend::BulkFeatureGroup

Description	Value
Description:	Enables ticket bulk action feature only for the listed groups.
Group:	Ticket
SubGroup:	Core::TicketBulkAction
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::BulkFeatureGroup'} = ['admin', 'users'];</code>

2.9. Core::TicketDynamicFieldDefault

2.9.1. Ticket::EventModulePost###TicketDynamicFieldDefault

Description	Value
Description:	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.
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::EventModulePost'}->{'TicketDynamicFieldDefault'} = { 'Module' => 'Kernel::System::Ticket::Event::TicketDynamicFieldDefault', 'Transaction' => '1' };</code>

2.9.2. Ticket::TicketDynamicFieldDefault###Element1

Description	Value
Description:	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://doc.otrs.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element1'} = { 'Event' => 'TicketCreate', 'Name' => 'Field1', 'Value' => 'Default' };</code>

2.9.3. Ticket::TicketDynamicFieldDefault###Element2

Description	Value
Description:	Configures a default TicketDymnicField 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://doc.otrs.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element2'} = { 'Event' => '', 'Name' => '', 'Value' => '' };</pre>

2.9.4. Ticket::TicketDynamicFieldDefault###Element3

Description	Value
Description:	Configures a default TicketDymnicField 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://doc.otrs.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element3'} = { 'Event' => '', 'Name' => '', 'Value' => '' };</pre>

2.9.5. Ticket::TicketDynamicFieldDefault###Element4

Description	Value
Description:	Configures a default TicketDymnicField 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://doc.otrs.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element4'} = { 'Event' => '', 'Name' => '', 'Value' => '' };</pre>

Description	Value
	};

2.9.6. Ticket::TicketDynamicFieldDefault###Element5

Description	Value
Description:	Configures a default TicketDymnicField 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://doc.otrs.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element5'} = { 'Event' => '', 'Name' => '', 'Value' => '' };</pre>

2.9.7. Ticket::TicketDynamicFieldDefault###Element6

Description	Value
Description:	Configures a default TicketDymnicField 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://doc.otrs.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element6'} = { 'Event' => '', 'Name' => '', 'Value' => '' };</pre>

2.9.8. Ticket::TicketDynamicFieldDefault###Element7

Description	Value
Description:	Configures a default TicketDymnicField 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://doc.otrs.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	

Description	Value
	<pre>\$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element7'} = { 'Event' => '', 'Name' => '', 'Value' => '' };</pre>

2.9.9. Ticket::TicketDynamicFieldDefault###Element8

Description	Value
Description:	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://doc.otrs.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element8'} = { 'Event' => '', 'Name' => '', 'Value' => '' };</pre>

2.9.10. Ticket::TicketDynamicFieldDefault###Element9

Description	Value
Description:	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://doc.otrs.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element9'} = { 'Event' => '', 'Name' => '', 'Value' => '' };</pre>

2.9.11. Ticket::TicketDynamicFieldDefault###Element10

Description	Value
Description:	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://doc.otrs.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault

Description	Value
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element10'} = { 'Event' => '', 'Name' => '', 'Value' => '' };</pre>

2.9.12. Ticket::TicketDynamicFieldDefault###Element11

Description	Value
Description:	Configures a default TicketDymnicField 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://doc.otrs.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element11'} = { 'Event' => '', 'Name' => '', 'Value' => '' };</pre>

2.9.13. Ticket::TicketDynamicFieldDefault###Element12

Description	Value
Description:	Configures a default TicketDymnicField 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://doc.otrs.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element12'} = { 'Event' => '', 'Name' => '', 'Value' => '' };</pre>

2.9.14. Ticket::TicketDynamicFieldDefault###Element13

Description	Value
Description:	Configures a default TicketDymnicField 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://doc.otrs.org/), chapter "Ticket Event Module".

Description	Value
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element13'} = { 'Event' => '', 'Name' => '', 'Value' => '' };</pre>

2.9.15. Ticket::TicketDynamicFieldDefault###Element14

Description	Value
Description:	Configures a default TicketDynsicField 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://doc.otrs.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element14'} = { 'Event' => '', 'Name' => '', 'Value' => '' };</pre>

2.9.16. Ticket::TicketDynamicFieldDefault###Element15

Description	Value
Description:	Configures a default TicketDynsicField 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://doc.otrs.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element15'} = { 'Event' => '', 'Name' => '', 'Value' => '' };</pre>

2.9.17. Ticket::TicketDynamicFieldDefault###Element16

Description	Value
Description:	Configures a default TicketDynsicField setting. "Name" defines the dynamic field which should be used, "Value" is the data that

Description	Value
	will be set, and "Event" defines the trigger event. Please check the developer manual (http://doc.otrs.org/), chapter "Ticket Event Module".
Group:	Ticket
SubGroup:	Core::TicketDynamicFieldDefault
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::TicketDynamicFieldDefault'}->{'Element16'} = { 'Event' => '', 'Name' => '', 'Value' => '' };</pre>

2.10. Core::TicketWatcher

2.10.1. Ticket::Watcher

Description	Value
Description:	Enables or disables the ticket watcher feature, to keep track of tickets without being the owner nor the responsible.
Group:	Ticket
SubGroup:	Core::TicketWatcher
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Watcher'} = '0';</pre>

2.10.2. Ticket::WatcherGroup

Description	Value
Description:	Enables ticket watcher feature only for the listed groups.
Group:	Ticket
SubGroup:	Core::TicketWatcher
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::WatcherGroup'} = ['admin', 'users'];</pre>

2.11. Frontend::Admin::ModuleRegistration

2.11.1. Frontend::Module###AdminQueue

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1

Description	Value
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminQueue'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Queue', 'Description' => 'Create and manage queues.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Queues', 'Prio' => '100' }, 'NavBarName' => 'Admin', 'Title' => 'Queues' };</pre>

2.11.2. Frontend::Module###AdminResponse

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminResponse'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Queue', 'Description' => 'Create and manage response templates.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Responses', 'Prio' => '200' }, 'NavBarName' => 'Admin', 'Title' => 'Responses' };</pre>

2.11.3. Frontend::Module###AdminQueueResponses

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminQueueResponses'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Queue',</pre>

Description	Value
	<pre>'Description' => 'Link responses to queues.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Responses <-> Queues', 'Prio' => '300' }, 'NavBarName' => 'Admin', 'Title' => 'Responses <-> Queues' };</pre>

2.11.4. Frontend::Module###AdminAutoResponse

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminAutoResponse'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Queue', 'Description' => 'Create and manage responses that are automatically sent.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Auto Responses', 'Prio' => '400' }, 'NavBarName' => 'Admin', 'Title' => 'Auto Responses' };</pre>

2.11.5. Frontend::Module###AdminQueueAutoResponse

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminQueueAutoResponse'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Queue', 'Description' => 'Link queues to auto responses.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Auto Responses <-> Queues', 'Prio' => '500' }, 'NavBarName' => 'Admin', 'Title' => 'Auto Responses <-> Queues' };</pre>

2.11.6. Frontend::Module###AdminAttachment

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Frontend::Module'}->{'AdminAttachment'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Queue', 'Description' => 'Create and manage attachments.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Attachments', 'Prio' => '600' }, 'NavBarName' => 'Admin', 'Title' => 'Attachments' }; </pre>

2.11.7. Frontend::Module###AdminResponseAttachment

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Frontend::Module'}->{'AdminResponseAttachment'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Queue', 'Description' => 'Link attachments to responses templates.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Attachments <-> Responses', 'Prio' => '700' }, 'NavBarName' => 'Admin', 'Title' => 'Attachments <-> Responses' }; </pre>

2.11.8. Frontend::Module###AdminSalutation

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1

Description	Value
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminSalutation'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Queue', 'Description' => 'Create and manage salutations.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Salutations', 'Prio' => '800' }, 'NavBarName' => 'Admin', 'Title' => 'Salutations' }; </pre>

2.11.9. Frontend::Module###AdminSignature

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminSignature'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Queue', 'Description' => 'Create and manage signatures.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Signatures', 'Prio' => '900' }, 'NavBarName' => 'Admin', 'Title' => 'Signatures' }; </pre>

2.11.10. Frontend::Module###AdminSystemAddress

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminSystemAddress'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Email', </pre>

Description	Value
	<pre>'Description' => 'Set sender email addresses for this system.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Email Addresses', 'Prio' => '300' }, 'NavBarName' => 'Admin', 'Title' => 'Email Addresses' };</pre>

2.11.11. Frontend::Module###AdminNotification

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminNotification'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Ticket', 'Description' => 'Manage notifications that are sent to agents.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Agent Notifications', 'Prio' => '400' }, 'NavBarName' => 'Admin', 'Title' => 'Agent Notifications' };</pre>

2.11.12. Frontend::Module###AdminNotificationEvent

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminNotificationEvent'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Ticket', 'Description' => 'Create and manage event based notifications.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Notifications (Event)', 'Prio' => '400' }, 'NavBarName' => 'Admin', 'Title' => 'Notifications (Event)' };</pre>

2.11.13. Frontend::Module###AdminService

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminService'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Ticket', 'Description' => 'Create and manage services.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Services', 'Prio' => '900' }, 'NavBarName' => 'Admin', 'Title' => 'Services' }; </pre>

2.11.14. Frontend::Module###AdminSLA

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminSLA'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Ticket', 'Description' => 'Create and manage Service Level Agreements (SLAs).', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Service Level Agreements', 'Prio' => '1000' }, 'NavBarName' => 'Admin', 'Title' => 'Service Level Agreements' }; </pre>

2.11.15. Frontend::Module###AdminType

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1

Description	Value
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminType'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Ticket', 'Description' => 'Create and manage ticket types.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Types', 'Prio' => '700' }, 'NavBarName' => 'Admin', 'Title' => 'Types' }; </pre>

2.11.16. Frontend::Module###AdminState

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminState'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Ticket', 'Description' => 'Create and manage ticket states.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'States', 'Prio' => '800' }, 'NavBarName' => 'Admin', 'Title' => 'States' }; </pre>

2.11.17. Frontend::Module###AdminPriority

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AdminPriority'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'Ticket', </pre>

Description	Value
	<pre>'Description' => 'Create and manage ticket priorities.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'Priorities', 'Prio' => '850' }, 'NavBarName' => 'Admin', 'Title' => 'Priorities' };</pre>

2.11.18. Frontend::Module###AdminGenericAgent

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Admin::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AdminGenericAgent'} = { 'Description' => 'Admin', 'Group' => ['admin'], 'NavBarModule' => { 'Block' => 'System', 'Description' => 'Manage periodic tasks.', 'Module' => 'Kernel::Output::HTML::NavBarModuleAdmin', 'Name' => 'GenericAgent', 'Prio' => '300' }, 'NavBarName' => 'Admin', 'Title' => 'GenericAgent' };</pre>

2.12. Frontend::Agent

2.12.1. Ticket::Frontend::PendingDiffTime

Description	Value
Description:	Time in seconds that gets added to the actual time if setting a pending-state (default: 86400 = 1 day).
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::PendingDiffTime'} = '86400';</pre>

2.12.2. Ticket::Frontend::ListType

Description	Value
Description:	Shows existing parent/child queue lists in the system in the form of a tree or a list.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1

Description	Value
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::ListType'} = 'tree';</code>

2.12.3. Ticket::Frontend::TextAreaEmail

Description	Value
Description:	Permitted width for compose email windows.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::TextAreaEmail'} = '82';</code>

2.12.4. Ticket::Frontend::TextAreaNote

Description	Value
Description:	Permitted width for compose note windows.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::TextAreaNote'} = '78';</code>

2.12.5. Ticket::Frontend::InformAgentMaxSize

Description	Value
Description:	Max size (in rows) of the informed agents box in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::InformAgentMaxSize'} = '3';</code>

2.12.6. Ticket::Frontend::InvolvedAgentMaxSize

Description	Value
Description:	Max size (in rows) of the involved agents box in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<code>\$Self->{'Ticket::Frontend::InvolvedAgentMaxSize'} = '3';</code>

2.12.7. Ticket::Frontend::CustomerInfoCompose

Description	Value
Description:	Shows the customer user information (phone and email) in the compose screen.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerInfoCompose'} = '1';</code>

2.12.8. Ticket::Frontend::CustomerInfoComposeMaxSize

Description	Value
Description:	Max size (in characters) of the customer information table (phone and email) in the compose screen.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerInfoComposeMaxSize'} = '22';</code>

2.12.9. Ticket::Frontend::CustomerInfoZoom

Description	Value
Description:	Shows the customer user's info in the ticket zoom view.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerInfoZoom'} = '1';</code>

2.12.10. Ticket::Frontend::CustomerInfoZoomMaxSize

Description	Value
Description:	Maximum size (in characters) of the customer information table in the ticket zoom view.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerInfoZoomMaxSize'} = '22';</code>

2.12.11. Ticket::Frontend::AccountTime

Description	Value
Description:	Activates time accounting.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AccountTime'} = '1';</code>

2.12.12. Ticket::Frontend::TimeUnits

Description	Value
Description:	Sets the preferred time units (e.g. work units, hours, minutes).
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::TimeUnits'} = '(work units)';</code>

2.12.13. Ticket::Frontend::NeedAccountedTime

Description	Value
Description:	Defines if time accounting is mandatory in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::NeedAccountedTime'} = '0';</code>

2.12.14. Ticket::Frontend::BulkAccountedTime

Description	Value
Description:	Defines if time accounting must be set to all tickets in bulk action.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::BulkAccountedTime'} = '1';</code>

2.12.15. Ticket::Frontend::NeedSpellCheck

Description	Value
Description:	Defines if composed messages have to be spell checked in the agent interface.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::NeedSpellCheck'} = '0';</code>

2.12.16. Ticket::Frontend::NewOwnerSelection

Description	Value
Description:	Shows an owner selection in phone and email tickets in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::NewOwnerSelection'} = '1';</code>

2.12.17. Ticket::Frontend::NewResponsibleSelection

Description	Value
Description:	Show a responsible selection in phone and email tickets in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::NewResponsibleSelection'} = '1';</code>

2.12.18. Ticket::Frontend::NewQueueSelectionType

Description	Value
Description:	Defines the receipt target of the phone ticket and the sender of the email ticket ("Queue" shows all queues, "SystemAddress" displays all system addresses) in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::NewQueueSelectionType'} = 'Queue';</code>

2.12.19. Ticket::Frontend::NewQueueSelectionString

Description	Value
Description:	Determines the strings that will be shown as receipt (To:) of the phone ticket and as sender (From:) of the email ticket in the agent

Description	Value
	interface. For Queue as NewQueueSelectionType "<Queue>" shows the names of the queues and for SystemAddress "<Realname> <<Email>>" shows the name and email of the receipt.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::NewQueueSelectionString'} = '<Queue>';</pre>

2.12.20. Ticket::Frontend::NewQueueOwnSelection

Description	Value
Description:	Determines which options will be valid of the receipt (phone ticket) and the sender (email ticket) in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::NewQueueOwnSelection'} = { '1' => 'First Queue!', '2' => 'Second Queue!' };</pre>

2.12.21. Ticket::Frontend::ShowCustomerTickets

Description	Value
Description:	Shows customer history tickets in AgentTicketPhone, AgentTicketEmail and AgentTicketCustomer.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::ShowCustomerTickets'} = '1';</pre>

2.12.22. NewTicketInNewWindow::Enabled

Description	Value
Description:	If enabled, TicketPhone and TicketEmail will be open in new windows.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<code>\$Self->{'NewTicketInNewWindow::Enabled'} = '0';</code>

2.12.23. CustomerDBLink

Description	Value
Description:	Defines an external link to the database of the customer (e.g. 'http://yourhost/customer.php?CID=\$Data{"CustomerID"}' or "").
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CustomerDBLink'} = '\$Env{"CGIHandle"}?Action=AgentTicketCustomer;TicketID=\$Data{"TicketID"}';</code>

2.12.24. CustomerDBLinkTarget

Description	Value
Description:	Defines the target attribute in the link to external customer database. E.g. 'target="cdb"'.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CustomerDBLinkTarget'} = '';</code>

2.12.25. Frontend::CommonObject###QueueObject

Description	Value
Description:	Path of the file that stores all the settings for the QueueObject object for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::CommonObject'}->{'QueueObject'} = 'Kernel::System::Queue';</code>

2.12.26. Frontend::CommonObject###TicketObject

Description	Value
Description:	Path of the file that stores all the settings for the TicketObject for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<code>\$Self->{'Frontend::CommonObject'}->{'TicketObject'} = 'Kernel::System::Ticket';</code>

2.12.27. Frontend::CommonParam###Action

Description	Value
Description:	Defines the default used Frontend-Module if no Action parameter given in the url on the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::CommonParam'}->{'Action'} = 'AgentDashboard';</code>

2.12.28. Frontend::CommonParam###QueueID

Description	Value
Description:	Default queue ID used by the system in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::CommonParam'}->{'QueueID'} = '0';</code>

2.12.29. Frontend::CommonParam###TicketID

Description	Value
Description:	Default ticket ID used by the system in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Frontend::CommonParam'}->{'TicketID'} = '';</code>

2.13. Frontend::Agent::CustomerSearch

2.13.1. Ticket::Frontend::CustomerSearchAutoComplete###Active

Description	Value
Description:	Enables or disables the autocomplete feature for the customer search in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::CustomerSearch
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerSearchAutoComplete'}->{'Active'} = '1';</code>

2.13.2. Ticket::Frontend::CustomerSearchAutoComplete###MinQueryLength

Description	Value
Description:	Sets the minimum number of characters before autocomplete query is sent.
Group:	Ticket
SubGroup:	Frontend::Agent::CustomerSearch
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerSearchAutoComplete'}->{'MinQueryLength'} = '2';</pre>

2.13.3. Ticket::Frontend::CustomerSearchAutoComplete###QueryDelay

Description	Value
Description:	Delay time between autocomplete queries in milliseconds.
Group:	Ticket
SubGroup:	Frontend::Agent::CustomerSearch
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerSearchAutoComplete'}->{'QueryDelay'} = '100';</pre>

2.13.4. Ticket::Frontend::CustomerSearchAutoComplete###MaxResultsDisplayed

Description	Value
Description:	Sets the number of search results to be displayed for the autocomplete feature.
Group:	Ticket
SubGroup:	Frontend::Agent::CustomerSearch
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerSearchAutoComplete'}->{'MaxResultsDisplayed'} = '20';</pre>

2.13.5. Ticket::Frontend::CustomerSearchAutoComplete::DynamicWidth

Description	Value
Description:	Determines if the search results container for the autocomplete feature should adjust its width dynamically.
Group:	Ticket
SubGroup:	Frontend::Agent::CustomerSearch
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerSearchAutoComplete::DynamicWidth'} = '1';</pre>

2.14. Frontend::Agent::Dashboard

2.14.1. DashboardBackend###0100-TicketPendingReminder

Description	Value
Description:	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.
Group:	Ticket
SubGroup:	Frontend::Agent::Dashboard
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'DashboardBackend'}->{'0100-TicketPendingReminder'} = { 'Attributes' => 'TicketPendingTimeOlderMinutes=1;StateType=pending reminder;SortBy=PendingTime;OrderBy=Down;', 'Block' => 'ContentLarge', 'CacheTTLLocal' => '0.5', 'Default' => '1', 'Description' => 'All tickets with a reminder set where the reminder date has been reached', 'Filter' => 'Locked', 'Group' => '', 'Limit' => '10', 'Module' => 'Kernel::Output::HTML::DashboardTicketGeneric', 'Permission' => 'rw', 'Time' => 'UntilTime', 'Title' => 'Reminder Tickets' }; </pre>

2.14.2. DashboardBackend###0110-TicketEscalation

Description	Value
Description:	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.
Group:	Ticket
SubGroup:	Frontend::Agent::Dashboard
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'DashboardBackend'}->{'0110-TicketEscalation'} = { 'Attributes' => 'TicketEscalationTimeOlderMinutes=1;SortBy=EscalationTime;OrderBy=Down;', 'Block' => 'ContentLarge', 'CacheTTLLocal' => '0.5', 'Default' => '1', 'Description' => 'All escalated tickets', 'Filter' => 'All', 'Group' => '', 'Limit' => '10', 'Module' => 'Kernel::Output::HTML::DashboardTicketGeneric', </pre>

Description	Value
	<pre>'Permission' => 'rw', 'Time' => 'EscalationTime', 'Title' => 'Escalated Tickets' };</pre>

2.14.3. DashboardBackend###0120-TicketNew

Description	Value
Description:	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.
Group:	Ticket
SubGroup:	Frontend::Agent::Dashboard
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'DashboardBackend'}->{'0120-TicketNew'} = { 'Attributes' => 'StateType=new;', 'Block' => 'ContentLarge', 'CacheTTLLocal' => '0.5', 'Default' => '1', 'Description' => 'All new tickets, these tickets have not been worked on yet', 'Filter' => 'All', 'Group' => '', 'Limit' => '10', 'Module' => 'Kernel::Output::HTML::DashboardTicketGeneric', 'Permission' => 'rw', 'Time' => 'Age', 'Title' => 'New Tickets' };</pre>

2.14.4. DashboardBackend###0130-TicketOpen

Description	Value
Description:	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.
Group:	Ticket
SubGroup:	Frontend::Agent::Dashboard
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'DashboardBackend'}->{'0130-TicketOpen'} = { 'Attributes' => 'StateType=open;', 'Block' => 'ContentLarge', 'CacheTTLLocal' => '0.5', 'Default' => '1', 'Description' => 'All open tickets, these tickets have already been worked on, but need a response', 'Filter' => 'All',</pre>

Description	Value
	<pre>'Group' => '', 'Limit' => '10', 'Module' => 'Kernel::Output::HTML::DashboardTicketGeneric', 'Permission' => 'rw', 'Time' => 'Age', 'Title' => 'Open Tickets / Need to be answered' };</pre>

2.14.5. DashboardBackend###0250-TicketStats

Description	Value
Description:	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.
Group:	Ticket
SubGroup:	Frontend::Agent::Dashboard
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'DashboardBackend'}->{'0250-TicketStats'} = { 'Block' => 'ContentSmall', 'CacheTTL' => '30', 'Closed' => '1', 'Created' => '1', 'Default' => '1', 'Group' => '', 'Module' => 'Kernel::Output::HTML::DashboardTicketStatsGeneric', 'Permission' => 'rw', 'Title' => '7 Day Stats' };</pre>

2.14.6. DashboardBackend###0260-TicketCalendar

Description	Value
Description:	Parameters for the dashboard backend of the ticket 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.
Group:	Ticket
SubGroup:	Frontend::Agent::Dashboard
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'DashboardBackend'}->{'0260-TicketCalendar'} = { 'Block' => 'ContentSmall', 'CacheTTL' => '2', 'Default' => '1', 'Group' => '', 'Limit' => '6', 'Module' => 'Kernel::Output::HTML::DashboardCalendar', 'OwnerOnly' => '', 'Permission' => 'rw', 'Title' => 'Upcoming Events' };</pre>

Description	Value
	};

2.15. Frontend::Agent::ModuleMetaHead

2.15.1. Frontend::HeaderMetaModule###2-TicketSearch

Description	Value
Description:	Module to generate html OpenSearch profile for short ticket search in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleMetaHead
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::HeaderMetaModule'}->{'2-TicketSearch'} = { 'Action' => 'AgentTicketSearch', 'Module' => 'Kernel::Output::HTML::HeaderMetaTicketSearch' };</pre>

2.16. Frontend::Agent::ModuleNotify

2.16.1. Frontend::NotifyModule###5-Ticket::TicketEscalation

Description	Value
Description:	Module to show notifications and escalations (ShownMax: max. shown escalations, EscalationInMinutes: Show ticket which will escalation in, CacheTime: Cache of calculated escalations in seconds).
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleNotify
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::NotifyModule'}->{'5-Ticket::TicketEscalation'} = { 'CacheTime' => '40', 'EscalationInMinutes' => '120', 'Module' => 'Kernel::Output::HTML::NotificationAgentTicketEscalation', 'ShownMax' => '25' };</pre>

2.17. Frontend::Agent::ModuleRegistration

2.17.1. Frontend::Module###AgentTicketQueue

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketQueue'} = {</pre>

Description	Value
	<pre>'Description' => 'Overview of all open Tickets', 'Loader' => { 'CSS' => ['Core.AgentTicketQueue.css'] }, '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' };</pre>

2.17.2. Frontend::Module###AgentTicketPhone

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$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' };</pre>

Description	Value
	};

2.17.3. Frontend::Module###AgentTicketPhoneOutbound

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketPhoneOutbound'} = { 'Description' => 'Phone Call', 'Loader' => { 'JavaScript' => ['Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Phone-Ticket' };</pre>

2.17.4. Frontend::Module###AgentTicketPhoneInbound

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketPhoneInbound'} = { 'Description' => 'Incoming Phone Call', 'Loader' => { 'JavaScript' => ['Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Phone-Ticket' };</pre>

2.17.5. Frontend::Module###AgentTicketEmail

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketEmail'} = { 'Description' => 'Create new email ticket', 'Loader' => { 'JavaScript' => ['Core.Agent.CustomerSearch.js', </pre>

Description	Value
	<pre> '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' }; </pre>

2.17.6. Frontend::Module###AgentTicketSearch

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Frontend::Module'}->{'AgentTicketSearch'} = { 'Description' => 'Search Ticket', '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' }; </pre>

2.17.7. Frontend::Module###AgentTicketLockedView

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Frontend::Module'}->{'AgentTicketLockedView'} = { </pre>

Description	Value
	<pre>'Description' => 'Locked Tickets', 'NavBarName' => 'Ticket', 'Title' => 'Locked Tickets' };</pre>

2.17.8. Frontend::Module###AgentTicketResponsibleView

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketResponsibleView'} = { 'Description' => 'Responsible Tickets', 'NavBarName' => 'Ticket', 'Title' => 'Responsible Tickets' };</pre>

2.17.9. Frontend::Module###AgentTicketWatchView

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketWatchView'} = { 'Description' => 'Watched Tickets', 'NavBarName' => 'Ticket', 'Title' => 'Watched Tickets' };</pre>

2.17.10. Frontend::Module###AgentCustomerSearch

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentCustomerSearch'} = { 'Description' => 'AgentCustomerSearch', 'NavBarName' => 'Ticket', 'Title' => 'AgentCustomerSearch' };</pre>

2.17.11. Frontend::Module###AgentTicketStatusView

Description	Value
Description:	Frontend module registration for the agent interface.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Frontend::Module'}->{'AgentTicketStatusView'} = { 'Description' => 'Overview of all open tickets', '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' }; </pre>

2.17.12. Frontend::Module###AgentTicketEscalationView

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Frontend::Module'}->{'AgentTicketEscalationView'} = { 'Description' => 'Overview of all escalated tickets', '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' }; </pre>

2.17.13. Frontend::Module###AgentZoom

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration

Description	Value
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentZoom'} = { 'Description' => 'compat module for AgentZoom to AgentTicketZoom', 'NavBarName' => 'Ticket', 'Title' => '' };</pre>

2.17.14. Frontend::Module###AgentTicketZoom

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketZoom'} = { 'Description' => 'Ticket Zoom', 'Loader' => { 'JavaScript' => ['thirdparty/jquery-tablesorter-2.0.5/jquery.tablesorter.js', 'Core.UI.Table.Sort.js', 'Core.Agent.TicketZoom.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Zoom' };</pre>

2.17.15. Frontend::Module###AgentTicketAttachment

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketAttachment'} = { 'Description' => 'To download attachments', 'NavBarName' => 'Ticket', 'Title' => '' };</pre>

2.17.16. Frontend::Module###AgentTicketPlain

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0

Description	Value
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketPlain'} = { 'Description' => 'Ticket plain view of an email', 'NavBarName' => 'Ticket', 'Title' => 'Plain' };</pre>

2.17.17. Frontend::Module###AgentTicketNote

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketNote'} = { 'Description' => 'Ticket Note', 'Loader' => { 'JavaScript' => ['Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Note' };</pre>

2.17.18. Frontend::Module###AgentTicketMerge

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketMerge'} = { 'Description' => 'Ticket Merge', 'NavBarName' => 'Ticket', 'Title' => 'Merge' };</pre>

2.17.19. Frontend::Module###AgentTicketPending

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketPending'} = { 'Description' => 'Ticket Pending', 'Loader' => { 'JavaScript' => [</pre>

Description	Value
	<pre>'Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Pending' };</pre>

2.17.20. Frontend::Module###AgentTicketWatcher

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketWatcher'} = { 'Description' => 'A TicketWatcher Module', 'NavBarName' => 'Ticket-Watcher', 'Title' => 'Ticket-Watcher' };</pre>

2.17.21. Frontend::Module###AgentTicketPriority

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketPriority'} = { 'Description' => 'Ticket Priority', 'Loader' => { 'JavaScript' => ['Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Priority' };</pre>

2.17.22. Frontend::Module###AgentTicketLock

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketLock'} = { 'Description' => 'Ticket Lock', 'NavBarName' => 'Ticket', 'Title' => 'Lock' };</pre>

Description	Value
	};

2.17.23. Frontend::Module###AgentTicketMove

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Frontend::Module'}->{'AgentTicketMove'} = { 'Description' => 'Ticket Move', 'Loader' => { 'JavaScript' => ['Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Move' }; </pre>

2.17.24. Frontend::Module###AgentTicketHistory

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Frontend::Module'}->{'AgentTicketHistory'} = { 'Description' => 'Ticket History', 'NavBarName' => 'Ticket', 'Title' => 'History' }; </pre>

2.17.25. Frontend::Module###AgentTicketOwner

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Frontend::Module'}->{'AgentTicketOwner'} = { 'Description' => 'Ticket Owner', 'Loader' => { 'JavaScript' => ['Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Owner' }; </pre>

Description	Value
	};

2.17.26. Frontend::Module###AgentTicketResponsible

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AgentTicketResponsible'} = { 'Description' => 'Ticket Responsible', 'Loader' => { 'JavaScript' => ['Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Responsible' }; </pre>

2.17.27. Frontend::Module###AgentTicketCompose

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AgentTicketCompose'} = { 'Description' => 'Ticket Compose email Answer', 'Loader' => { 'JavaScript' => ['Core.Agent.CustomerSearch.js', 'Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Compose' }; </pre>

2.17.28. Frontend::Module###AgentTicketBounce

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::Module'}->{'AgentTicketBounce'} = { 'Description' => 'Ticket Compose Bounce Email', 'NavBarName' => 'Ticket', }; </pre>

Description	Value
	'Title' => 'Bounce' };

2.17.29. Frontend::Module###AgentTicketForward

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketForward'} = { 'Description' => 'Ticket Forward Email', 'Loader' => { 'JavaScript' => ['Core.Agent.CustomerSearch.js', 'Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Forward' };</pre>

2.17.30. Frontend::Module###AgentTicketCustomer

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketCustomer'} = { 'Description' => 'Ticket Customer', 'Loader' => { 'JavaScript' => ['Core.Agent.CustomerSearch.js', 'Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Customer' };</pre>

2.17.31. Frontend::Module###AgentTicketClose

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketClose'} = {</pre>

Description	Value
	<pre>'Description' => 'Ticket Close', 'Loader' => { 'JavaScript' => ['Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Close' };</pre>

2.17.32. Frontend::Module###AgentTicketFreeText

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketFreeText'} = { 'Description' => 'Ticket FreeText', 'Loader' => { 'JavaScript' => ['Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Free Fields' };</pre>

2.17.33. Frontend::Module###AgentTicketPrint

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketPrint'} = { 'Description' => 'Ticket Print', 'NavBarName' => 'Ticket', 'Title' => 'Print' };</pre>

2.17.34. Frontend::Module###AgentTicketBulk

Description	Value
Description:	Frontend module registration for the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Module'}->{'AgentTicketBulk'} = {</pre>

Description	Value
	<pre>'Description' => 'Ticket bulk module', 'Loader' => { 'JavaScript' => ['Core.Agent.TicketAction.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Bulk-Action' };</pre>

2.18. Frontend::Agent::Preferences

2.18.1. PreferencesGroups###NewTicketNotify

Description	Value
Description:	Parameters for the NewTicketNotify object in the preferences view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PreferencesGroups'}->{'NewTicketNotify'} = { 'Active' => '1', 'Column' => 'Email Settings', 'Data' => { '0' => 'No', '1' => 'Yes' }, 'DataSelected' => '0', 'Desc' => 'Send me a notification if there is a new ticket in "My Queues".', 'Key' => 'Send new ticket notifications', 'Label' => 'New ticket notification', 'Module' => 'Kernel::Output::HTML::PreferencesGeneric', 'PrefKey' => 'UserSendNewTicketNotification', 'Prio' => '1000' };</pre>

2.18.2. PreferencesGroups###FollowUpNotify

Description	Value
Description:	Parameters for the FollowUpNotify object in the preference view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PreferencesGroups'}->{'FollowUpNotify'} = { 'Active' => '1', 'Column' => 'Email Settings', 'Data' => { '0' => 'No', '1' => 'Yes' }, 'DataSelected' => '0', 'Desc' => 'Send me a notification if a customer sends a follow up and I\'m the owner of the ticket or the ticket is unlocked and is in one of my subscribed queues.', };</pre>

Description	Value
	<pre>'Key' => 'Send ticket follow up notifications', 'Label' => 'Ticket follow up notification', 'Module' => 'Kernel::Output::HTML::PreferencesGeneric', 'PrefKey' => 'UserSendFollowUpNotification', 'Prio' => '2000' };</pre>

2.18.3. PreferencesGroups###LockTimeoutNotify

Description	Value
Description:	Parameters for the LockTimeoutNotify object in the preference view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PreferencesGroups'}->{'LockTimeoutNotify'} = { 'Active' => '1', 'Column' => 'Email Settings', 'Data' => { '0' => 'No', '1' => 'Yes' }, 'DataSelected' => '0', 'Desc' => 'Send me a notification if a ticket is unlocked by the system.', 'Key' => 'Send ticket lock timeout notifications', 'Label' => 'Ticket lock timeout notification', 'Module' => 'Kernel::Output::HTML::PreferencesGeneric', 'PrefKey' => 'UserSendLockTimeoutNotification', 'Prio' => '3000' };</pre>

2.18.4. PreferencesGroups###MoveNotify

Description	Value
Description:	Parameters for the MoveNotify object in the preference view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PreferencesGroups'}->{'MoveNotify'} = { 'Active' => '1', 'Column' => 'Email Settings', 'Data' => { '0' => 'No', '1' => 'Yes' }, 'DataSelected' => '0', 'Desc' => 'Send me a notification if a ticket is moved into one of "My Queues".', 'Key' => 'Send ticket move notifications', 'Label' => 'Ticket move notification', 'Module' => 'Kernel::Output::HTML::PreferencesGeneric', 'PrefKey' => 'UserSendMoveNotification', 'Prio' => '4000' };</pre>

2.18.5. PreferencesGroups###WatcherNotify

Description	Value
Description:	Parameters for the WatcherNotify object in the preference view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'PreferencesGroups'}->{'WatcherNotify'} = { 'Active' => '1', 'Column' => 'Email Settings', 'Data' => { '0' => 'No', '1' => 'Yes' }, 'DataSelected' => '0', 'Desc' => 'Send me the same notifications for my watched tickets that the ticket owners will get.', 'Key' => 'Send ticket watch notifications', 'Label' => 'Ticket watch notification', 'Module' => 'Kernel::Output::HTML::PreferencesTicketWatcher', 'PrefKey' => 'UserSendWatcherNotification', 'Prio' => '5000' }; </pre>

2.18.6. PreferencesGroups###CustomQueue

Description	Value
Description:	Parameters for the CustomQueue object in the preference view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'PreferencesGroups'}->{'CustomQueue'} = { 'Active' => '1', 'Column' => 'Other Settings', 'Desc' => 'Your queue selection of your favorite queues. You also get notified about those queues via email if enabled.', 'Key' => 'My Queues', 'Label' => 'My Queues', 'Module' => 'Kernel::Output::HTML::PreferencesCustomQueue', 'Permission' => 'ro', 'Prio' => '1000' }; </pre>

2.18.7. PreferencesGroups###RefreshTime

Description	Value
Description:	Parameters for the RefreshTime object in the preference view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Preferences
Valid:	1

Description	Value
Required:	0
Config-Setting:	<pre> \$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' => 'Refresh Overviews after', 'Label' => 'Overview Refresh Time', 'Module' => 'Kernel::Output::HTML::PreferencesGeneric', 'PrefKey' => 'UserRefreshTime', 'Prio' => '2000' }; </pre>

2.18.8. PreferencesGroups###TicketOverviewSmallPageShown

Description	Value
Description:	Parameters for the pages (in which the tickets are shown) of the small ticket overview.
Group:	Ticket
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre> \$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::PreferencesGeneric', 'PrefKey' => 'UserTicketOverviewSmallPageShown', 'Prio' => '8000' }; </pre>

2.18.9. PreferencesGroups###TicketOverviewMediumPageShown

Description	Value
Description:	Parameters for the pages (in which the tickets are shown) of the medium ticket overview.
Group:	Ticket
SubGroup:	Frontend::Agent::Preferences

Description	Value
Valid:	1
Required:	0
Config-Setting:	<pre> \$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::PreferencesGeneric', 'PrefKey' => 'UserTicketOverviewMediumPageShown', 'Prio' => '8100' }; </pre>

2.18.10. PreferencesGroups###TicketOverviewPreviewPageShown

Description	Value
Description:	Parameters for the pages (in which the tickets are shown) of the ticket preview overview.
Group:	Ticket
SubGroup:	Frontend::Agent::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre> \$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::PreferencesGeneric', 'PrefKey' => 'UserTicketOverviewPreviewPageShown', 'Prio' => '8200' }; </pre>

2.18.11. PreferencesGroups###CreateNextMask

Description	Value
Description:	Parameters for the CreateNextMask object in the preference view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Preferences

Description	Value
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'PreferencesGroups'}->{'CreateNextMask'} = { 'Active' => '1', 'Column' => 'Other Settings', 'Data' => { '' => 'CreateTicket', 'AgentTicketZoom' => 'TicketZoom' }, 'DataSelected' => '', 'Key' => 'Show this screen after I created a new ticket', 'Label' => 'Screen after new ticket', 'Module' => 'Kernel::Output::HTML::PreferencesGeneric', 'PrefKey' => 'UserCreateNextMask', 'Prio' => '3000' };</pre>

2.19. Frontend::Agent::SearchRouter

2.19.1. Frontend::Search###Ticket

Description	Value
Description:	Search backend router.
Group:	Ticket
SubGroup:	Frontend::Agent::SearchRouter
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::Search'}->{'Ticket'} = { '^AgentTicket' => 'Action=AgentTicketSearch;Subaction=AJAX' };</pre>

2.20. Frontend::Agent::Ticket::ArticleAttachmentModule

2.20.1. Ticket::Frontend::ArticleAttachmentModule###1-Download

Description	Value
Description:	Shows a link to download article attachments in the zoom view of the article in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ArticleAttachmentModule
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::ArticleAttachmentModule'}->{'1-Download'} = { 'Module' => 'Kernel::Output::HTML::ArticleAttachmentDownload' };</pre>

2.20.2. Ticket::Frontend::ArticleAttachmentModule###2-HTML-Viewer

Description	Value
Description:	Shows a link to access article attachments via a html online viewer in the zoom view of the article in the agent interface.
Group:	Ticket

Description	Value
SubGroup:	Frontend::Agent::Ticket::ArticleAttachmentModule
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::ArticleAttachmentModule'}->{'2-HTML-Viewer'} = { 'Module' => 'Kernel::Output::HTML::ArticleAttachmentHTMLViewer' };</pre>

2.21. Frontend::Agent::Ticket::ArticleComposeModule

2.21.1. Ticket::Frontend::ArticleComposeModule###1-SignEmail

Description	Value
Description:	Module to compose signed messages (PGP or S/MIME).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ArticleComposeModule
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::ArticleComposeModule'}->{'1-SignEmail'} = { 'Module' => 'Kernel::Output::HTML::ArticleComposeSign' };</pre>

2.21.2. Ticket::Frontend::ArticleComposeModule###2-CryptEmail

Description	Value
Description:	Module to crypt composed messages (PGP or S/MIME).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ArticleComposeModule
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::ArticleComposeModule'}->{'2-CryptEmail'} = { 'Module' => 'Kernel::Output::HTML::ArticleComposeCrypt' };</pre>

2.22. Frontend::Agent::Ticket::ArticleViewModule

2.22.1. Ticket::Frontend::ArticleViewModule###1-PGP

Description	Value
Description:	Agent interface article notification module to check PGP.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ArticleViewModule
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::ArticleViewModule'}->{'1-PGP'} = { 'Module' => 'Kernel::Output::HTML::ArticleCheckPGP' };</pre>

2.22.2. Ticket::Frontend::ArticleViewModule###1-SMIME

Description	Value
Description:	Agent interface module to check incoming emails in the Ticket-Zoom-View if the S/MIME-key is available and true.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ArticleViewModule
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::ArticleViewModule'}->{'1-SMIME'} = { 'Module' => 'Kernel::Output::HTML::ArticleCheckSMIME' };</pre>

2.23. Frontend::Agent::Ticket::ArticleViewModulePre

2.23.1. Ticket::Frontend::ArticlePreViewModule###1-PGP

Description	Value
Description:	Agent interface article notification module to check PGP.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ArticleViewModulePre
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::ArticlePreViewModule'}->{'1-PGP'} = { 'Module' => 'Kernel::Output::HTML::ArticleCheckPGP' };</pre>

2.23.2. Ticket::Frontend::ArticlePreViewModule###1-SMIME

Description	Value
Description:	Agent interface article notification module to check S/MIME.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ArticleViewModulePre
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::ArticlePreViewModule'}->{'1-SMIME'} = { 'Module' => 'Kernel::Output::HTML::ArticleCheckSMIME' };</pre>

2.24. Frontend::Agent::Ticket::MenuModule

2.24.1. Ticket::Frontend::MenuModule###000-Back

Description	Value
Description:	Shows a link in the menu to go back in the ticket zoom view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1

Description	Value
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::MenuModule'}->{'000-Back'} = { 'Action' => '', 'Description' => 'Back', 'Link' => '\$Env{"LastScreenOverview"};TicketID= \$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Back', 'PopupType' => '', 'Target' => '' };</pre>

2.24.2. Ticket::Frontend::MenuModule###100-Lock

Description	Value
Description:	Shows a link in the menu to lock/unlock tickets in the ticket zoom view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::MenuModule'}->{'100-Lock'} = { 'Action' => 'AgentTicketLock', 'Module' => 'Kernel::Output::HTML::TicketMenuLock', 'Name' => 'Lock', 'Target' => '' };</pre>

2.24.3. Ticket::Frontend::MenuModule###200-History

Description	Value
Description:	Shows a link in the menu to access the history of a ticket in the ticket zoom view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::MenuModule'}->{'200-History'} = { 'Action' => 'AgentTicketHistory', 'Description' => 'Show the ticket history', 'Link' => 'Action=AgentTicketHistory;TicketID=\$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'History', 'PopupType' => 'TicketHistory', 'Target' => '' };</pre>

2.24.4. Ticket::Frontend::MenuModule###210-Print

Description	Value
Description:	Shows a link in the menu to print a ticket or an article in the ticket zoom view of the agent interface.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Ticket::Frontend::MenuModule'}->{'210-Print'} = { 'Action' => 'AgentTicketPrint', 'Description' => 'Print this ticket', 'Link' => 'Action=AgentTicketPrint;TicketID=\$QData{"TicketID"}', 'LinkParam' => 'target="print"', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Print', 'PopupType' => 'TicketAction', 'Target' => '' }; </pre>

2.24.5. Ticket::Frontend::MenuModule###300-Priority

Description	Value
Description:	Shows a link in the menu to see the priority of a ticket in the ticket zoom view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Ticket::Frontend::MenuModule'}->{'300-Priority'} = { 'Action' => 'AgentTicketPriority', 'Description' => 'Change the ticket priority', 'Link' => 'Action=AgentTicketPriority;TicketID= \$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Priority', 'PopupType' => 'TicketAction', 'Target' => '' }; </pre>

2.24.6. Ticket::Frontend::MenuModule###310-FreeText

Description	Value
Description:	Shows a link in the menu to add a free text field in the ticket zoom view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'Ticket::Frontend::MenuModule'}->{'310-FreeText'} = { 'Action' => 'AgentTicketFreeText', 'Description' => 'Change the free fields for this ticket', 'Link' => 'Action=AgentTicketFreeText;TicketID= \$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Free Fields', 'PopupType' => 'TicketAction', 'Target' => '' }; </pre>

2.24.7. Ticket::Frontend::MenuModule###320-Link

Description	Value
Description:	Shows a link in the menu that allows linking a ticket with another object in the ticket zoom view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::MenuModule'}->{'320-Link'} = { 'Action' => 'AgentLinkObject', 'Description' => 'Link this ticket to other objects', 'Link' => 'Action=AgentLinkObject;SourceObject=Ticket;SourceKey= \$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Link', 'PopupType' => 'TicketAction', 'Target' => '' }; </pre>

2.24.8. Ticket::Frontend::MenuModule###400-Owner

Description	Value
Description:	Shows a link in the menu to see the owner of a ticket in the ticket zoom view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::MenuModule'}->{'400-Owner'} = { 'Action' => 'AgentTicketOwner', 'Description' => 'Change the owner for this ticket', 'Link' => 'Action=AgentTicketOwner;TicketID=\$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Owner', 'PopupType' => 'TicketAction', 'Target' => '' }; </pre>

2.24.9. Ticket::Frontend::MenuModule###410-Responsible

Description	Value
Description:	Shows a link in the menu to see the responsible agent of a ticket in the ticket zoom view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::MenuModule'}->{'410-Responsible'} = { 'Action' => 'AgentTicketResponsible', 'Description' => 'Change the responsible person for this ticket', 'Link' => 'Action=AgentTicketResponsible;TicketID= \$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuResponsible', }; </pre>

Description	Value
	<pre>'Name' => 'Responsible', 'PopupType' => 'TicketAction', 'Target' => '' };</pre>

2.24.10. Ticket::Frontend::MenuModule###420-Customer

Description	Value
Description:	Shows a link in the menu to see the customer who requested the ticket in the ticket zoom view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::MenuModule'}->{'420-Customer'} = { 'Action' => 'AgentTicketCustomer', 'Description' => 'Change the customer for this ticket', 'Link' => 'Action=AgentTicketCustomer;TicketID= \$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Customer', 'PopupType' => 'TicketAction', 'Target' => '' };</pre>

2.24.11. Ticket::Frontend::MenuModule###420-Note

Description	Value
Description:	Shows a link in the menu to add a note in the ticket zoom view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::MenuModule'}->{'420-Note'} = { 'Action' => 'AgentTicketNote', 'Description' => 'Add a note to this ticket', 'Link' => 'Action=AgentTicketNote;TicketID=\$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Note', 'PopupType' => 'TicketAction', 'Target' => '' };</pre>

2.24.12. Ticket::Frontend::MenuModule###430-Merge

Description	Value
Description:	Shows a link in the menu that allows merging tickets in the ticket zoom view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0

Description	Value
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::MenuModule'}->{'430-Merge'} = { 'Action' => 'AgentTicketMerge', 'Description' => 'Merge into a different ticket', 'Link' => 'Action=AgentTicketMerge;TicketID=\$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Merge', 'PopupType' => 'TicketAction', 'Target' => '' }; </pre>

2.24.13. Ticket::Frontend::MenuModule###440-Pending

Description	Value
Description:	Shows a link in the menu to set a ticket as pending in the ticket zoom view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::MenuModule'}->{'440-Pending'} = { 'Action' => 'AgentTicketPending', 'Description' => 'Set this ticket to pending', 'Link' => 'Action=AgentTicketPending;TicketID=\$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Pending', 'PopupType' => 'TicketAction', 'Target' => '' }; </pre>

2.24.14. Ticket::Frontend::MenuModule###448-Watch

Description	Value
Description:	Shows a link in the menu for subscribing / unsubscribing from a ticket in the ticket zoom view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::MenuModule'}->{'448-Watch'} = { 'Action' => 'AgentTicketWatcher', 'Module' => 'Kernel::Output::HTML::TicketMenuTicketWatcher', 'Name' => 'Watch', 'Target' => '' }; </pre>

2.24.15. Ticket::Frontend::MenuModule###450-Close

Description	Value
Description:	Shows a link in the menu to close a ticket in the ticket zoom view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule

Description	Value
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::MenuModule'}->{'450-Close'} = { 'Action' => 'AgentTicketClose', 'Description' => 'Close this ticket', 'Link' => 'Action=AgentTicketClose;TicketID=\$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Close', 'PopupType' => 'TicketAction', 'Target' => '' };</pre>

2.24.16. Ticket::Frontend::MenuModule###460-Delete

Description	Value
Description:	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".
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::MenuModule'}->{'460-Delete'} = { 'Action' => 'AgentTicketMove', 'Description' => 'Delete this ticket', 'Link' => 'Action=AgentTicketMove;TicketID=\$Data{"TicketID"};DestQueue=Delete', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Delete', 'PopupType' => '', 'Target' => '' };</pre>

2.24.17. Ticket::Frontend::MenuModule###470-Spam

Description	Value
Description:	Shows a link to set a ticket as spam 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".
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModule
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::MenuModule'}->{'470-Spam'} = { 'Action' => 'AgentTicketMove', 'Description' => 'Mark as Spam!', 'Link' => 'Action=AgentTicketMove;TicketID=\$Data{"TicketID"};DestQueue=Delete', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Spam', 'PopupType' => '', 'Target' => '' };</pre>

Description	Value
	};

2.25. Frontend::Agent::Ticket::MenuModulePre

2.25.1. Ticket::Frontend::PreMenuModule###100-Lock

Description	Value
Description:	Shows a link in the menu to lock / unlock a ticket in the ticket overviews of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModulePre
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::PreMenuModule'}->{'100-Lock'} = { 'Action' => 'AgentTicketLock', 'Module' => 'Kernel::Output::HTML::TicketMenuLock', 'Name' => 'Lock', 'PopupType' => '', 'Target' => '' };</pre>

2.25.2. Ticket::Frontend::PreMenuModule###200-Zoom

Description	Value
Description:	Shows a link in the menu to zoom a ticket in the ticket overviews of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModulePre
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::PreMenuModule'}->{'200-Zoom'} = { 'Action' => 'AgentTicketZoom', 'Description' => 'Look into a ticket!', 'Link' => 'Action=AgentTicketZoom;TicketID=\$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Zoom', 'PopupType' => '', 'Target' => '' };</pre>

2.25.3. Ticket::Frontend::PreMenuModule###210-History

Description	Value
Description:	Shows a link in the menu to see the history of a ticket in every ticket overview of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModulePre
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::PreMenuModule'}->{'210-History'} = { 'Action' => 'AgentTicketHistory', 'Description' => 'Show the ticket history',</pre>

Description	Value
	<pre>'Link' => 'Action=AgentTicketHistory;TicketID=\$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'History', 'PopupType' => 'TicketHistory', 'Target' => '' };</pre>

2.25.4. Ticket::Frontend::PreMenuModule###300-Priority

Description	Value
Description:	Shows a link in the menu to set the priority of a ticket in every ticket overview of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModulePre
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::PreMenuModule'}->{'300-Priority'} = { 'Action' => 'AgentTicketPriority', 'Description' => 'Change the priority for this ticket', 'Link' => 'Action=AgentTicketPriority;TicketID= \$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Priority', 'PopupType' => 'TicketAction', 'Target' => '' };</pre>

2.25.5. Ticket::Frontend::PreMenuModule###420-Note

Description	Value
Description:	Shows a link in the menu to add a note to a ticket in every ticket overview of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModulePre
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::PreMenuModule'}->{'420-Note'} = { 'Action' => 'AgentTicketNote', 'Description' => 'Add a note to this ticket', 'Link' => 'Action=AgentTicketNote;TicketID=\$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Note', 'PopupType' => 'TicketAction', 'Target' => '' };</pre>

2.25.6. Ticket::Frontend::PreMenuModule###440-Close

Description	Value
Description:	Shows a link in the menu to close a ticket in every ticket overview of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModulePre
Valid:	1

Description	Value
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::PreMenuModule'}->{'440-Close'} = { 'Action' => 'AgentTicketClose', 'Description' => 'Close this ticket', 'Link' => 'Action=AgentTicketClose;TicketID=\$QData{"TicketID"}', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Close', 'PopupType' => 'TicketAction', 'Target' => '' }; </pre>

2.25.7. Ticket::Frontend::PreMenuModule###445-Move

Description	Value
Description:	Shows a link in the menu to move a ticket in every ticket overview of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModulePre
Valid:	1
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::PreMenuModule'}->{'445-Move'} = { 'Action' => 'AgentTicketMove', 'Description' => 'Change queue!', 'Module' => 'Kernel::Output::HTML::TicketMenuMove', 'Name' => 'Move' }; </pre>

2.25.8. Ticket::Frontend::PreMenuModule###450-Delete

Description	Value
Description:	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".
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModulePre
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'Ticket::Frontend::PreMenuModule'}->{'450-Delete'} = { 'Action' => 'AgentTicketMove', 'Description' => 'Delete this ticket', 'Link' => 'Action=AgentTicketMove;TicketID=\$Data{"TicketID"};DestQueue=Delete', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Delete', 'PopupType' => '', 'Target' => '' }; </pre>

2.25.9. Ticket::Frontend::PreMenuModule###460-Spam

Description	Value
Description:	Shows a link in the menu to set a ticket as spam in every ticket overview of the agent interface. Additional access control to show

Description	Value
	or not show this link can be done by using Key "Group" and Content like "rw:group1;move_into:group2".
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::MenuModulePre
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::PreMenuModule'}->{'460-Spam'} = { 'Action' => 'AgentTicketMove', 'Description' => 'Mark as Spam!', 'Link' => 'Action=AgentTicketMove;TicketID= \$Data{"TicketID"};DestQueue=Delete', 'Module' => 'Kernel::Output::HTML::TicketMenuGeneric', 'Name' => 'Spam', 'PopupType' => '', 'Target' => '' };</pre>

2.26. Frontend::Agent::Ticket::ViewBounce

2.26.1. Ticket::Frontend::AgentTicketBounce###Permission

Description	Value
Description:	Required permissions to use the ticket bounce screen in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBounce
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketBounce'}->{'Permission'} = 'bounce';</pre>

2.26.2. Ticket::Frontend::AgentTicketBounce###RequiredLock

Description	Value
Description:	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).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBounce
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketBounce'}->{'RequiredLock'} = '1';</pre>

2.26.3. Ticket::Frontend::AgentTicketBounce###StateDefault

Description	Value
Description:	Defines the default next state of a ticket after being bounced, in the ticket bounce screen of the agent interface.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBounce
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketBounce'}->{'StateDefault'} = 'closed successful';</pre>

2.26.4. Ticket::Frontend::AgentTicketBounce###StateType

Description	Value
Description:	Defines the next state of a ticket after being bounced, in the ticket bounce screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBounce
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketBounce'}->{'StateType'} = ['open', 'closed'];</pre>

2.26.5. Ticket::Frontend::BounceText

Description	Value
Description:	Defines the default ticket bounced notification for customer/sender in the ticket bounce screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBounce
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::BounceText'} = 'Your email with ticket number "<OTRS_TICKET>" is bounced to "<OTRS_BOUNCE_TO>". Contact this address for further information.';</pre>

2.27. Frontend::Agent::Ticket::ViewBulk

2.27.1. Ticket::Frontend::AgentTicketBulk###RequiredLock

Description	Value
Description:	Automatically lock and set owner to current Agent after selecting for an Bulk Action.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBulk
Valid:	1
Required:	0
Config-Setting:	

Description	Value
	<code>\$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'RequiredLock'} = '1';</code>

2.27.2. Ticket::Frontend::AgentTicketBulk###TicketType

Description	Value
Description:	Sets the ticket type in the ticket bulk screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBulk
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'TicketType'} = '1';</code>

2.27.3. Ticket::Frontend::AgentTicketBulk###Owner

Description	Value
Description:	Sets the ticket owner in the ticket bulk screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBulk
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'Owner'} = '1';</code>

2.27.4. Ticket::Frontend::AgentTicketBulk###Responsible

Description	Value
Description:	Sets the responsible agent of the ticket in the ticket bulk screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBulk
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'Responsible'} = '1';</code>

2.27.5. Ticket::Frontend::AgentTicketBulk###State

Description	Value
Description:	If a note is added by an agent, sets the state of a ticket in the ticket bulk screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBulk
Valid:	1
Required:	0

Description	Value
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'State'} = '1';</code>

2.27.6. Ticket::Frontend::AgentTicketBulk###StateType

Description	Value
Description:	Defines the next state of a ticket after adding a note, in the ticket bulk screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBulk
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'StateType'} = ['open', 'closed', 'pending reminder', 'pending auto'];</code>

2.27.7. Ticket::Frontend::AgentTicketBulk###StateDefault

Description	Value
Description:	Defines the default next state of a ticket after adding a note, in the ticket bulk screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBulk
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'StateDefault'} = 'open';</code>

2.27.8. Ticket::Frontend::AgentTicketBulk###Priority

Description	Value
Description:	Shows the ticket priority options in the ticket bulk screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBulk
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'Priority'} = '1';</code>

2.27.9. Ticket::Frontend::AgentTicketBulk###PriorityDefault

Description	Value
Description:	Defines the default ticket priority in the ticket bulk screen of the agent interface.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBulk
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'PriorityDefault'} = '3 normal';</pre>

2.27.10. Ticket::Frontend::AgentTicketBulk###ArticleTypeDefault

Description	Value
Description:	Defines the default type of the note in the ticket bulk screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBulk
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'ArticleTypeDefault'} = 'note-internal';</pre>

2.27.11. Ticket::Frontend::AgentTicketBulk###ArticleTypes

Description	Value
Description:	Specifies the different note types that will be used in the system.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewBulk
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketBulk'}->{'ArticleTypes'} = { 'note-external' => '1', 'note-internal' => '1', 'note-report' => '0' };</pre>

2.28. Frontend::Agent::Ticket::ViewClose

2.28.1. Ticket::Frontend::AgentTicketClose###Permission

Description	Value
Description:	Required permissions to use the close ticket screen in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'Permission'} = 'close';</pre>

2.28.2. Ticket::Frontend::AgentTicketClose###RequiredLock

Description	Value
Description:	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).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'RequiredLock'} = '1';</code>

2.28.3. Ticket::Frontend::AgentTicketClose###TicketType

Description	Value
Description:	Sets the ticket type in the close ticket screen of the agent interface (Ticket::Type needs to be activated).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'TicketType'} = '0';</code>

2.28.4. Ticket::Frontend::AgentTicketClose###Service

Description	Value
Description:	Sets the service in the close ticket screen of the agent interface (Ticket::Service needs to be activated).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'Service'} = '0';</code>

2.28.5. Ticket::Frontend::AgentTicketClose###Owner

Description	Value
Description:	Sets the ticket owner in the close ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'Owner'} = '0';</code>

2.28.6. Ticket::Frontend::AgentTicketClose###OwnerMandatory

Description	Value
Description:	Sets if ticket owner must be selected by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'OwnerMandatory'} = '0';</pre>

2.28.7. Ticket::Frontend::AgentTicketClose###Responsible

Description	Value
Description:	Sets the responsible agent of the ticket in the close ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'Responsible'} = '0';</pre>

2.28.8. Ticket::Frontend::AgentTicketClose###State

Description	Value
Description:	If a note is added by an agent, sets the state of a ticket in the close ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'State'} = '1';</pre>

2.28.9. Ticket::Frontend::AgentTicketClose###StateType

Description	Value
Description:	Defines the next state of a ticket after adding a note, in the close ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'StateType'} = ['closed'];</pre>

2.28.10. Ticket::Frontend::AgentTicketClose###StateDefault

Description	Value
Description:	Defines the default next state of a ticket after adding a note, in the close ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'StateDefault'} = 'closed successful';</pre>

2.28.11. Ticket::Frontend::AgentTicketClose###Note

Description	Value
Description:	Allows adding notes in the close ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'Note'} = '1';</pre>

2.28.12. Ticket::Frontend::AgentTicketClose###Subject

Description	Value
Description:	Sets the default subject for notes added in the close ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'Subject'} = '\$Text{"Close"}';</pre>

2.28.13. Ticket::Frontend::AgentTicketClose###Body

Description	Value
Description:	Sets the default body text for notes added in the close ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'Body'} = '';</pre>

2.28.14. Ticket::Frontend::AgentTicketClose###InvolvedAgent

Description	Value
Description:	Shows a list of all the involved agents on this ticket, in the close ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'InvolvedAgent'} = '0';</pre>

2.28.15. Ticket::Frontend::AgentTicketClose###InformAgent

Description	Value
Description:	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.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'InformAgent'} = '0';</pre>

2.28.16. Ticket::Frontend::AgentTicketClose###ArticleTypeDefault

Description	Value
Description:	Defines the default type of the note in the close ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'ArticleTypeDefault'} = 'note-internal';</pre>

2.28.17. Ticket::Frontend::AgentTicketClose###ArticleTypes

Description	Value
Description:	Specifies the different note types that will be used in the system.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0

Description	Value
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'ArticleTypes'} = { 'note-external' => '0', 'note-internal' => '1', 'note-report' => '0' };</pre>

2.28.18. Ticket::Frontend::AgentTicketClose###Priority

Description	Value
Description:	Shows the ticket priority options in the close ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'Priority'} = '0';</pre>

2.28.19. Ticket::Frontend::AgentTicketClose###PriorityDefault

Description	Value
Description:	Defines the default ticket priority in the close ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'PriorityDefault'} = '3 normal';</pre>

2.28.20. Ticket::Frontend::AgentTicketClose###Title

Description	Value
Description:	Shows the title fields in the close ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'Title'} = '0';</pre>

2.28.21. Ticket::Frontend::AgentTicketClose###HistoryType

Description	Value
Description:	Defines the history type for the close ticket screen action, which gets used for ticket history in the agent interface.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'HistoryType'} = 'AddNote';</code>

2.28.22. Ticket::Frontend::AgentTicketClose###HistoryComment

Description	Value
Description:	Defines the history comment for the close ticket screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'HistoryComment'} = '%Close';</code>

2.28.23. Ticket::Frontend::AgentTicketClose###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket close screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewClose
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketClose'}->{'DynamicField'} = {};</code>

2.29. Frontend::Agent::Ticket::ViewCompose

2.29.1. Ticket::Frontend::AgentTicketCompose###Permission

Description	Value
Description:	Required permissions to use the ticket compose screen in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewCompose
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketCompose'}->{'Permission'} = 'compose';</code>

2.29.2. Ticket::Frontend::AgentTicketCompose###RequiredLock

Description	Value
Description:	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).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewCompose
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketCompose'}->{'RequiredLock'} = '1';</pre>

2.29.3. Ticket::Frontend::AgentTicketCompose###StateDefault

Description	Value
Description:	Defines the default next state of a ticket if it is composed / answered in the ticket compose screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewCompose
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketCompose'}->{'StateDefault'} = 'open';</pre>

2.29.4. Ticket::Frontend::AgentTicketCompose###StateType

Description	Value
Description:	Defines the next possible states after composing / answering a ticket in the ticket compose screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewCompose
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketCompose'}->{'StateType'} = ['open', 'closed', 'pending auto', 'pending reminder'];</pre>

2.29.5. Ticket::Frontend::AgentTicketCompose###ArticleTypes

Description	Value
Description:	Specifies the different article types that will be used in the system.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewCompose
Valid:	1

Description	Value
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketCompose'}->{'ArticleTypes'} = ['email-external', 'email-internal'];</pre>

2.29.6. Ticket::Frontend::ResponseFormat

Description	Value
Description:	Defines the format of responses in the ticket compose screen of the agent interface (\$QData{"OrigFrom"} is From 1:1, \$QData{"OrigFromName"} is only realname of From).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewCompose
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::ResponseFormat'} = '\$QData{"Salutation"} \$QData{"StdResponse"} \$QData{"Signature"} \$TimeShort{"\$QData{"Created"}"} - \$QData{"OrigFromName"} \$Text{"wrote"}: \$QData{"Body"} ';</pre>

2.29.7. Ticket::Frontend::Quote

Description	Value
Description:	Defines the used character for email quotes in the ticket compose screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewCompose
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::Quote'} = '>';</pre>

2.29.8. Ticket::Frontend::ComposeAddCustomerAddress

Description	Value
Description:	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.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewCompose
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::ComposeAddCustomerAddress'} = '1';</pre>

2.29.9. Ticket::Frontend::ComposeReplaceSenderAddress

Description	Value
Description:	Replaces the original sender with current customer's email address on compose answer in the ticket compose screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewCompose
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::ComposeReplaceSenderAddress'} = '0';</code>

2.29.10. Ticket::Frontend::ComposeExcludeCcRecipients

Description	Value
Description:	Uses Cc recipients in reply Cc list on compose an email answer in the ticket compose screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewCompose
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::ComposeExcludeCcRecipients'} = '0';</code>

2.29.11. Ticket::Frontend::AgentTicketCompose###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket compose screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewCompose
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketCompose'}->{'DynamicField'} = {};</code>

2.30. Frontend::Agent::Ticket::ViewCustomer

2.30.1. Ticket::Frontend::AgentTicketCustomer###Permission

Description	Value
Description:	Required permissions to change the customer of a ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewCustomer
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<code>\$Self->{'Ticket::Frontend::AgentTicketCustomer'}->{'Permission'} = 'customer';</code>

2.30.2. Ticket::Frontend::AgentTicketCustomer###RequiredLock

Description	Value
Description:	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).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewCustomer
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketCustomer'}->{'RequiredLock'} = '0';</code>

2.31. Frontend::Agent::Ticket::ViewEmailNew

2.31.1. Ticket::Frontend::AgentTicketEmail###Priority

Description	Value
Description:	Sets the default priority for new email tickets in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEmailNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'Priority'} = '3 normal';</code>

2.31.2. Ticket::Frontend::AgentTicketEmail###ArticleType

Description	Value
Description:	Sets the default article type for new email tickets in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEmailNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'ArticleType'} = 'email-external';</code>

2.31.3. Ticket::Frontend::AgentTicketEmail###SenderType

Description	Value
Description:	Sets the default sender type for new email tickets in the agent interface.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEmailNew
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'SenderType'} = 'agent';</pre>

2.31.4. Ticket::Frontend::AgentTicketEmail###Subject

Description	Value
Description:	Sets the default subject for new email tickets (e.g. 'email Outbound') in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEmailNew
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'Subject'} = '';</pre>

2.31.5. Ticket::Frontend::AgentTicketEmail###Body

Description	Value
Description:	Sets the default text for new email tickets in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEmailNew
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'Body'} = '';</pre>

2.31.6. Ticket::Frontend::AgentTicketEmail###StateDefault

Description	Value
Description:	Sets the default next ticket state, after the creation of an email ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEmailNew
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'StateDefault'} = 'open';</pre>

2.31.7. Ticket::Frontend::AgentTicketEmail###StateType

Description	Value
Description:	Determines the next possible ticket states, after the creation of a new email ticket in the agent interface.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEmailNew
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'StateType'} = ['open', 'pending auto', 'pending reminder', 'closed'];</pre>

2.31.8. Ticket::Frontend::AgentTicketEmail###HistoryType

Description	Value
Description:	Defines the history type for the email ticket screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEmailNew
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'HistoryType'} = 'EmailAgent';</pre>

2.31.9. Ticket::Frontend::AgentTicketEmail###HistoryComment

Description	Value
Description:	Defines the history comment for the email ticket screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEmailNew
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'HistoryComment'} = '';</pre>

2.31.10. Ticket::Frontend::AgentTicketEmail###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket email screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEmailNew
Valid:	1
Required:	0
Config-Setting:	

Description	Value
	<code>\$Self->{'Ticket::Frontend::AgentTicketEmail'}->{'DynamicField'} = {};</code>

2.32. Frontend::Agent::Ticket::ViewEscalation

2.32.1. Ticket::Frontend::AgentTicketEscalationView###TicketPermission

Description	Value
Description:	Defines the required permission to show a ticket in the escalation view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEscalation
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketEscalationView'}->{'TicketPermission'} = 'ro';</code>

2.32.2. Ticket::Frontend::AgentTicketEscalationView###ViewableTicketsPage

Description	Value
Description:	Shows all open tickets (even if they are locked) in the escalation view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEscalation
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketEscalationView'}->{'ViewableTicketsPage'} = '50';</code>

2.32.3. Ticket::Frontend::AgentTicketEscalationView###SortBy::Default

Description	Value
Description:	Defines the default ticket attribute for ticket sorting in the escalation view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEscalation
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketEscalationView'}->{'SortBy::Default'} = 'EscalationTime';</code>

2.32.4. Ticket::Frontend::AgentTicketEscalationView###Order::Default

Description	Value
Description:	Defines the default ticket order (after priority sort) in the escalation view of the agent interface. Up: oldest on top. Down: latest on top.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewEscalation
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketEscalationView'}->{'Order::Default'} = 'Up';</pre>

2.33. Frontend::Agent::Ticket::ViewForward

2.33.1. Ticket::Frontend::AgentTicketForward###Permission

Description	Value
Description:	Required permissions to use the ticket forward screen in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewForward
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketForward'}->{'Permission'} = 'forward';</pre>

2.33.2. Ticket::Frontend::AgentTicketForward###RequiredLock

Description	Value
Description:	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).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewForward
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketForward'}->{'RequiredLock'} = '1';</pre>

2.33.3. Ticket::Frontend::AgentTicketForward###StateDefault

Description	Value
Description:	Defines the default next state of a ticket after being forwarded, in the ticket forward screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewForward
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketForward'}->{'StateDefault'} = 'closed successful';</pre>

2.33.4. Ticket::Frontend::AgentTicketForward###StateType

Description	Value
Description:	Defines the next possible states after forwarding a ticket in the ticket forward screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewForward
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketForward'}->{'StateType'} = ['open', 'closed', 'pending reminder', 'pending auto'];</pre>

2.33.5. Ticket::Frontend::AgentTicketForward###ArticleTypeDefault

Description	Value
Description:	Defines the default type of forwarded message in the ticket forward screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewForward
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketForward'}->{'ArticleTypeDefault'} = 'email-external';</pre>

2.33.6. Ticket::Frontend::AgentTicketForward###ArticleTypes

Description	Value
Description:	Specifies the different article types that will be used in the system.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewForward
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketForward'}->{'ArticleTypes'} = ['email-external', 'email-internal'];</pre>

2.33.7. Ticket::Frontend::AgentTicketForward###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket forward screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket

Description	Value
SubGroup:	Frontend::Agent::Ticket::ViewForward
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketForward'}->{'DynamicField'} = {};</pre>

2.34. Frontend::Agent::Ticket::ViewFreeText

2.34.1. Ticket::Frontend::AgentTicketFreeText###Permission

Description	Value
Description:	Required permissions to use the ticket free text screen in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'Permission'} = 'rw';</pre>

2.34.2. Ticket::Frontend::AgentTicketFreeText###RequiredLock

Description	Value
Description:	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).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'RequiredLock'} = '0';</pre>

2.34.3. Ticket::Frontend::AgentTicketFreeText###TicketType

Description	Value
Description:	Sets the ticket type in the ticket free text screen of the agent interface (Ticket::Type needs to be activated).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'TicketType'} = '1';</pre>

2.34.4. Ticket::Frontend::AgentTicketFreeText###Service

Description	Value
Description:	Sets the service in the ticket free text screen of the agent interface (Ticket::Service needs to be activated).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'Service'} = '1';</pre>

2.34.5. Ticket::Frontend::AgentTicketFreeText###Owner

Description	Value
Description:	Sets the ticket owner in the ticket free text screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'Owner'} = '0';</pre>

2.34.6. Ticket::Frontend::AgentTicketFreeText###OwnerMandatory

Description	Value
Description:	Sets if ticket owner must be selected by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'OwnerMandatory'} = '0';</pre>

2.34.7. Ticket::Frontend::AgentTicketFreeText###Responsible

Description	Value
Description:	Sets the responsible agent of the ticket in the ticket free text screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'Responsible'} = '0';</pre>

2.34.8. Ticket::Frontend::AgentTicketFreeText###State

Description	Value
Description:	If a note is added by an agent, sets the state of a ticket in the ticket free text screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'State'} = '0';</code>

2.34.9. Ticket::Frontend::AgentTicketFreeText###StateType

Description	Value
Description:	Defines the next state of a ticket after adding a note, in the ticket free text screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'StateType'} = ['open', 'closed', 'pending reminder', 'pending auto'];</code>

2.34.10. Ticket::Frontend::AgentTicketFreeText###StateDefault

Description	Value
Description:	Defines the default next state of a ticket after adding a note, in the ticket free text screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'StateDefault'} = 'open';</code>

2.34.11. Ticket::Frontend::AgentTicketFreeText###Note

Description	Value
Description:	Allows adding notes in the ticket free text screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1

Description	Value
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'Note'} = '0';</code>

2.34.12. Ticket::Frontend::AgentTicketFreeText###Subject

Description	Value
Description:	Defines the default subject of a note in the ticket free text screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'Subject'} = '\$Text{"Note"}';</code>

2.34.13. Ticket::Frontend::AgentTicketFreeText###Body

Description	Value
Description:	Defines the default body of a note in the ticket free text screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'Body'} = '';</code>

2.34.14. Ticket::Frontend::AgentTicketFreeText###InvolvedAgent

Description	Value
Description:	Shows a list of all the involved agents on this ticket, in the ticket free text screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'InvolvedAgent'} = '0';</code>

2.34.15. Ticket::Frontend::AgentTicketFreeText###InformAgent

Description	Value
Description:	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.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'InformAgent'} = '0';</pre>

2.34.16. Ticket::Frontend::AgentTicketFreeText###ArticleTypeDefault

Description	Value
Description:	Defines the default type of the note in the ticket free text screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'ArticleTypeDefault'} = 'note-internal';</pre>

2.34.17. Ticket::Frontend::AgentTicketFreeText###ArticleTypes

Description	Value
Description:	Specifies the different note types that will be used in the system.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'ArticleTypes'} = { 'note-external' => '1', 'note-internal' => '1', 'note-report' => '0' };</pre>

2.34.18. Ticket::Frontend::AgentTicketFreeText###Priority

Description	Value
Description:	Shows the ticket priority options in the ticket free text screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'Priority'} = '0';</pre>

2.34.19. Ticket::Frontend::AgentTicketFreeText###PriorityDefault

Description	Value
Description:	Defines the default ticket priority in the ticket free text screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'PriorityDefault'} = '3 normal';</pre>

2.34.20. Ticket::Frontend::AgentTicketFreeText###Title

Description	Value
Description:	Shows the title fields in the ticket free text screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'Title'} = '1';</pre>

2.34.21. Ticket::Frontend::AgentTicketFreeText###HistoryType

Description	Value
Description:	Defines the history type for the ticket free text screen action, which gets used for ticket history.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'HistoryType'} = 'AddNote';</pre>

2.34.22. Ticket::Frontend::AgentTicketFreeText###HistoryComment

Description	Value
Description:	Defines the history comment for the ticket free text screen action, which gets used for ticket history.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'HistoryComment'} = '%FreeText';</pre>

2.34.23. Ticket::Frontend::AgentTicketFreeText###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket free text screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewFreeText
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketFreeText'}->{'DynamicField'} = {};</pre>

2.35. Frontend::Agent::Ticket::ViewHistory

2.35.1. Ticket::Frontend::HistoryOrder

Description	Value
Description:	Shows the ticket history (reverse ordered) in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewHistory
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::HistoryOrder'} = 'normal';</pre>

2.36. Frontend::Agent::Ticket::ViewMailbox

2.36.1. Ticket::Frontend::AgentTicketLockedView###SortBy::Default

Description	Value
Description:	Defines the default ticket attribute for ticket sorting in the locked ticket view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMailbox
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketLockedView'}->{'SortBy::Default'} = 'Age';</pre>

2.36.2. Ticket::Frontend::AgentTicketLockedView###Order::Default

Description	Value
Description:	Defines the default ticket order in the ticket locked view of the agent interface. Up: oldest on top. Down: latest on top.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMailbox
Valid:	1
Required:	1

Description	Value
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketLockedView'}->{'Order::Default'} = 'Up';</pre>

2.36.3. Ticket::Frontend::AgentTicketResponsibleView###SortBy::Default

Description	Value
Description:	Defines the default ticket attribute for ticket sorting in the responsible view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMailbox
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketResponsibleView'}->{'SortBy::Default'} = 'Age';</pre>

2.36.4. Ticket::Frontend::AgentTicketResponsibleView###Order::Default

Description	Value
Description:	Defines the default ticket order in the responsible view of the agent interface. Up: oldest on top. Down: latest on top.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMailbox
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketResponsibleView'}->{'Order::Default'} = 'Up';</pre>

2.36.5. Ticket::Frontend::AgentTicketWatchView###SortBy::Default

Description	Value
Description:	Defines the default ticket attribute for ticket sorting in the watch view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMailbox
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketWatchView'}->{'SortBy::Default'} = 'Age';</pre>

2.36.6. Ticket::Frontend::AgentTicketWatchView###Order::Default

Description	Value
Description:	Defines the default ticket order in the watch view of the agent interface. Up: oldest on top. Down: latest on top.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMailbox

Description	Value
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketWatchView'}->{'Order::Default'} = 'Up';</pre>

2.37. Frontend::Agent::Ticket::ViewMerge

2.37.1. Ticket::Frontend::AgentTicketMerge###Permission

Description	Value
Description:	Required permissions to use the ticket merge screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMerge
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketMerge'}->{'Permission'} = 'rw';</pre>

2.37.2. Ticket::Frontend::AgentTicketMerge###RequiredLock

Description	Value
Description:	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).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMerge
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketMerge'}->{'RequiredLock'} = '1';</pre>

2.37.3. Ticket::Frontend::MergeText

Description	Value
Description:	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.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMerge
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::MergeText'} = 'Your email with ticket number "<OTRS_TICKET>" is merged to "<OTRS_MERGE_TO_TICKET>";</pre>

2.37.4. Ticket::Frontend::AutomaticMergeText

Description	Value
Description:	When tickets are merged, a note will be added automatically to the ticket which is no longer active. In this text area you can define this text (This text cannot be changed by the agent).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMerge
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AutomaticMergeText'} = 'Merged Ticket <OTRS_TICKET> to <OTRS_MERGE_TO_TICKET>.';</pre>

2.38. Frontend::Agent::Ticket::ViewMove

2.38.1. Ticket::Frontend::MoveType

Description	Value
Description:	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.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMove
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::MoveType'} = 'form';</pre>

2.38.2. Ticket::Frontend::AgentTicketMove###State

Description	Value
Description:	Allows to set a new ticket state in the move ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMove
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketMove'}->{'State'} = '1';</pre>

2.38.3. Ticket::DefaultNextMoveStateType

Description	Value
Description:	Defines the next state of a ticket after being moved to another queue, in the move ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMove
Valid:	1

Description	Value
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::DefaultNextMoveStateType'} = ['open', 'closed'];</pre>

2.38.4. Ticket::Frontend::AgentTicketMove###Priority

Description	Value
Description:	Shows the ticket priority options in the move ticket screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMove
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketMove'}->{'Priority'} = '0';</pre>

2.38.5. Ticket::Frontend::AgentTicketMove###NextScreen

Description	Value
Description:	Determines the next screen after the ticket is moved. LastScreenOverview will return to search results, queueview, dashboard or the like, LastScreenView will return to TicketZoom.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMove
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketMove'}->{'NextScreen'} = 'LastScreenView';</pre>

2.38.6. Ticket::Frontend::AgentTicketMove###Subject

Description	Value
Description:	Sets the default subject for notes added in the ticket move screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMove
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketMove'}->{'Subject'} = '\$Text{"Change Queue"}';</pre>

2.38.7. Ticket::Frontend::AgentTicketMove###Body

Description	Value
Description:	Sets the default body text for notes added in the ticket move screen of the agent interface.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMove
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketMove'}->{'Body'} = '';</code>

2.38.8. Ticket::Frontend::AgentTicketMove###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket move screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewMove
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketMove'}->{'DynamicField'} = {};</code>

2.39. Frontend::Agent::Ticket::ViewNote

2.39.1. Ticket::Frontend::AgentTicketNote###Permission

Description	Value
Description:	Required permissions to use the ticket note screen in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'Permission'} = 'note';</code>

2.39.2. Ticket::Frontend::AgentTicketNote###RequiredLock

Description	Value
Description:	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).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'RequiredLock'} = '0';</code>

2.39.3. Ticket::Frontend::AgentTicketNote###TicketType

Description	Value
Description:	Sets the ticket type in the ticket note screen of the agent interface (Ticket::Type needs to be activated).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'TicketType'} = '0';</code>

2.39.4. Ticket::Frontend::AgentTicketNote###Service

Description	Value
Description:	Sets the service in the ticket note screen of the agent interface (Ticket::Service needs to be activated).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'Service'} = '0';</code>

2.39.5. Ticket::Frontend::AgentTicketNote###Owner

Description	Value
Description:	Sets the ticket owner in the ticket note screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'Owner'} = '0';</code>

2.39.6. Ticket::Frontend::AgentTicketNote###OwnerMandatory

Description	Value
Description:	Sets if ticket owner must be selected by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'OwnerMandatory'} = '0';</code>

2.39.7. Ticket::Frontend::AgentTicketNote###Responsible

Description	Value
Description:	Sets the responsible agent of the ticket in the ticket note screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'Responsible'} = '0';</pre>

2.39.8. Ticket::Frontend::AgentTicketNote###State

Description	Value
Description:	If a note is added by an agent, sets the state of a ticket in the ticket note screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'State'} = '0';</pre>

2.39.9. Ticket::Frontend::AgentTicketNote###StateType

Description	Value
Description:	Defines the next state of a ticket after adding a note, in the ticket note screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'StateType'} = ['open', 'closed', 'pending reminder', 'pending auto'];</pre>

2.39.10. Ticket::Frontend::AgentTicketNote###StateDefault

Description	Value
Description:	Defines the default next state of a ticket after adding a note, in the ticket note screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	0
Required:	0

Description	Value
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'StateDefault'} = 'open';</pre>

2.39.11. Ticket::Frontend::AgentTicketNote###Note

Description	Value
Description:	Allows adding notes in the ticket note screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'Note'} = '1';</pre>

2.39.12. Ticket::Frontend::AgentTicketNote###Subject

Description	Value
Description:	Sets the default subject for notes added in the ticket note screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'Subject'} = '\$Text{"Note"}';</pre>

2.39.13. Ticket::Frontend::AgentTicketNote###Body

Description	Value
Description:	Sets the default body text for notes added in the ticket note screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'Body'} = '';</pre>

2.39.14. Ticket::Frontend::AgentTicketNote###InvolvedAgent

Description	Value
Description:	Shows a list of all the involved agents on this ticket, in the ticket note screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	0
Required:	0

Description	Value
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'InvolvedAgent'} = '0';</pre>

2.39.15. Ticket::Frontend::AgentTicketNote###InformAgent

Description	Value
Description:	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.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'InformAgent'} = '0';</pre>

2.39.16. Ticket::Frontend::AgentTicketNote###ArticleTypeDefault

Description	Value
Description:	Defines the default type of the note in the ticket note screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'ArticleTypeDefault'} = 'note-internal';</pre>

2.39.17. Ticket::Frontend::AgentTicketNote###ArticleTypes

Description	Value
Description:	Specifies the different note types that will be used in the system.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'ArticleTypes'} = { 'note-external' => '1', 'note-internal' => '1', 'note-report' => '0' };</pre>

2.39.18. Ticket::Frontend::AgentTicketNote###Priority

Description	Value
Description:	Shows the ticket priority options in the ticket note screen of the agent interface.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'Priority'} = '0';</code>

2.39.19. Ticket::Frontend::AgentTicketNote###PriorityDefault

Description	Value
Description:	Defines the default ticket priority in the ticket note screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'PriorityDefault'} = '3 normal';</code>

2.39.20. Ticket::Frontend::AgentTicketNote###Title

Description	Value
Description:	Shows the title fields in the ticket note screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'Title'} = '0';</code>

2.39.21. Ticket::Frontend::AgentTicketNote###HistoryType

Description	Value
Description:	Defines the history type for the ticket note screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'HistoryType'} = 'AddNote';</code>

2.39.22. Ticket::Frontend::AgentTicketNote###HistoryComment

Description	Value
Description:	Defines the history comment for the ticket note screen action, which gets used for ticket history in the agent interface.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'HistoryComment'} = '%Note';</pre>

2.39.23. Ticket::Frontend::AgentTicketNote###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket note screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewNote
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketNote'}->{'DynamicField'} = {};</pre>

2.40. Frontend::Agent::Ticket::ViewOwner

2.40.1. Ticket::Frontend::AgentTicketOwner###Permission

Description	Value
Description:	Required permissions to use the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'Permission'} = 'owner';</pre>

2.40.2. Ticket::Frontend::AgentTicketOwner###RequiredLock

Description	Value
Description:	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).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	

Description	Value
	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'RequiredLock'} = '0';</code>

2.40.3. Ticket::Frontend::AgentTicketOwner###TicketType

Description	Value
Description:	Sets the ticket type in the ticket owner screen of a zoomed ticket in the agent interface (Ticket::Type needs to be activated).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'TicketType'} = '0';</code>

2.40.4. Ticket::Frontend::AgentTicketOwner###Service

Description	Value
Description:	Sets the service in the ticket owner screen of a zoomed ticket in the agent interface (Ticket::Service needs to be activated).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'Service'} = '0';</code>

2.40.5. Ticket::Frontend::AgentTicketOwner###Owner

Description	Value
Description:	Sets the ticket owner in the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'Owner'} = '1';</code>

2.40.6. Ticket::Frontend::AgentTicketOwner###OwnerMandatory

Description	Value
Description:	Sets if ticket owner must be selected by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	

Description	Value
	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'OwnerMandatory'} = '1';</code>

2.40.7. Ticket::Frontend::AgentTicketOwner###Responsible

Description	Value
Description:	Sets the responsible agent of the ticket in the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'Responsible'} = '0';</code>

2.40.8. Ticket::Frontend::AgentTicketOwner###State

Description	Value
Description:	If a note is added by an agent, sets the state of the ticket in the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'State'} = '0';</code>

2.40.9. Ticket::Frontend::AgentTicketOwner###StateType

Description	Value
Description:	Defines the next state of a ticket after adding a note, in the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'StateType'} = ['open', 'pending reminder', 'pending auto'];</code>

2.40.10. Ticket::Frontend::AgentTicketOwner###StateDefault

Description	Value
Description:	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.
Group:	Ticket

Description	Value
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'StateDefault'} = 'open';</code>

2.40.11. Ticket::Frontend::AgentTicketOwner###Note

Description	Value
Description:	Allows adding notes in the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'Note'} = '1';</code>

2.40.12. Ticket::Frontend::AgentTicketOwner###Subject

Description	Value
Description:	Sets the default subject for notes added in the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'Subject'} = '\$Text{"Owner Update"}!';</code>

2.40.13. Ticket::Frontend::AgentTicketOwner###Body

Description	Value
Description:	Sets the default body text for notes added in the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'Body'} = '';</code>

2.40.14. Ticket::Frontend::AgentTicketOwner###InvolvedAgent

Description	Value
Description:	Shows a list of all the involved agents on this ticket, in the ticket owner screen of a zoomed ticket in the agent interface.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'InvolvedAgent'} = '0';</pre>

2.40.15. Ticket::Frontend::AgentTicketOwner###InformAgent

Description	Value
Description:	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.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'InformAgent'} = '0';</pre>

2.40.16. Ticket::Frontend::AgentTicketOwner###ArticleTypeDefault

Description	Value
Description:	Defines the default type of the note in the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'ArticleTypeDefault'} = 'note-internal';</pre>

2.40.17. Ticket::Frontend::AgentTicketOwner###ArticleTypes

Description	Value
Description:	Specifies the different note types that will be used in the system.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'ArticleTypes'} = { 'note-external' => '0', 'note-internal' => '1', 'note-report' => '0' };</pre>

2.40.18. Ticket::Frontend::AgentTicketOwner###Priority

Description	Value
Description:	Shows the ticket priority options in the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'Priority'} = '0';</code>

2.40.19. Ticket::Frontend::AgentTicketOwner###PriorityDefault

Description	Value
Description:	Defines the default ticket priority in the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'PriorityDefault'} = '3 normal';</code>

2.40.20. Ticket::Frontend::AgentTicketOwner###Title

Description	Value
Description:	Shows the title fields in the ticket owner screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'Title'} = '0';</code>

2.40.21. Ticket::Frontend::AgentTicketOwner###HistoryType

Description	Value
Description:	Defines the history type for the ticket owner screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'HistoryType'} = 'AddNote';</code>

2.40.22. Ticket::Frontend::AgentTicketOwner###HistoryComment

Description	Value
Description:	Defines the history comment for the ticket owner screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'HistoryComment'} = '%Owner';</pre>

2.40.23. Ticket::Frontend::AgentTicketOwner###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket owner screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewOwner
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketOwner'}->{'DynamicField'} = {};</pre>

2.41. Frontend::Agent::Ticket::ViewPending

2.41.1. Ticket::Frontend::AgentTicketPending###Permission

Description	Value
Description:	Required permissions to use the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'Permission'} = 'pending';</pre>

2.41.2. Ticket::Frontend::AgentTicketPending###RequiredLock

Description	Value
Description:	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).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1

Description	Value
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'RequiredLock'} = '1';</code>

2.41.3. Ticket::Frontend::AgentTicketPending###TicketType

Description	Value
Description:	Sets the ticket type in the ticket pending screen of a zoomed ticket in the agent interface (Ticket::Type needs to be activated).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'TicketType'} = '0';</code>

2.41.4. Ticket::Frontend::AgentTicketPending###Service

Description	Value
Description:	Sets the service in the ticket pending screen of a zoomed ticket in the agent interface (Ticket::Service needs to be activated).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'Service'} = '0';</code>

2.41.5. Ticket::Frontend::AgentTicketPending###Owner

Description	Value
Description:	Sets the ticket owner in the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'Owner'} = '0';</code>

2.41.6. Ticket::Frontend::AgentTicketPending###OwnerMandatory

Description	Value
Description:	Sets if ticket owner must be selected by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1

Description	Value
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'OwnerMandatory'} = '0';</pre>

2.41.7. Ticket::Frontend::AgentTicketPending###Responsible

Description	Value
Description:	Sets the responsible agent of the ticket in the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'Responsible'} = '0';</pre>

2.41.8. Ticket::Frontend::AgentTicketPending###State

Description	Value
Description:	If a note is added by an agent, sets the state of the ticket in the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'State'} = '1';</pre>

2.41.9. Ticket::Frontend::AgentTicketPending###StateType

Description	Value
Description:	Defines the next state of a ticket after adding a note, in the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'StateType'} = ['pending reminder', 'pending auto'];</pre>

2.41.10. Ticket::Frontend::AgentTicketPending###StateDefault

Description	Value
Description:	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.
Group:	Ticket

Description	Value
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'StateDefault'} = 'pending reminder';</code>

2.41.11. Ticket::Frontend::AgentTicketPending###Note

Description	Value
Description:	Allows adding notes in the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'Note'} = '1';</code>

2.41.12. Ticket::Frontend::AgentTicketPending###Subject

Description	Value
Description:	Sets the default subject for notes added in the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'Subject'} = '\$Text{"Pending"}!';</code>

2.41.13. Ticket::Frontend::AgentTicketPending###Body

Description	Value
Description:	Sets the default body text for notes added in the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'Body'} = '';</code>

2.41.14. Ticket::Frontend::AgentTicketPending###InvolvedAgent

Description	Value
Description:	Shows a list of all the involved agents on this ticket, in the ticket pending screen of a zoomed ticket in the agent interface.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'InvolvedAgent'} = '0';</pre>

2.41.15. Ticket::Frontend::AgentTicketPending###InformAgent

Description	Value
Description:	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.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'InformAgent'} = '0';</pre>

2.41.16. Ticket::Frontend::AgentTicketPending###ArticleTypeDefault

Description	Value
Description:	Defines the default type of the note in the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'ArticleTypeDefault'} = 'note-internal';</pre>

2.41.17. Ticket::Frontend::AgentTicketPending###ArticleTypes

Description	Value
Description:	Specifies the different note types that will be used in the system.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'ArticleTypes'} = { 'note-external' => '0', 'note-internal' => '1', 'note-report' => '0' };</pre>

2.41.18. Ticket::Frontend::AgentTicketPending###Priority

Description	Value
Description:	Shows the ticket priority options in the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'Priority'} = '0';</pre>

2.41.19. Ticket::Frontend::AgentTicketPending###PriorityDefault

Description	Value
Description:	Defines the default ticket priority in the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'PriorityDefault'} = '3 normal';</pre>

2.41.20. Ticket::Frontend::AgentTicketPending###Title

Description	Value
Description:	Shows the title fields in the ticket pending screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'Title'} = '0';</pre>

2.41.21. Ticket::Frontend::AgentTicketPending###HistoryType

Description	Value
Description:	Defines the history type for the ticket pending screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'HistoryType'} = 'AddNote';</pre>

2.41.22. Ticket::Frontend::AgentTicketPending###HistoryComment

Description	Value
Description:	Defines the history comment for the ticket pending screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'HistoryComment'} = '%Pending';</pre>

2.41.23. Ticket::Frontend::AgentTicketPending###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket pending screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPending
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPending'}->{'DynamicField'} = {};</pre>

2.42. Frontend::Agent::Ticket::ViewPhoneInbound

2.42.1. Ticket::Frontend::AgentTicketPhoneInbound###Permission

Description	Value
Description:	Required permissions to use the ticket phone inbound screen in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneInbound
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPhoneInbound'}->{'Permission'} = 'phone';</pre>

2.42.2. Ticket::Frontend::AgentTicketPhoneInbound###RequiredLock

Description	Value
Description:	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).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneInbound

Description	Value
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPhoneInbound'}->{'RequiredLock'} = '0';</pre>

2.42.3. Ticket::Frontend::AgentTicketPhoneInbound###ArticleType

Description	Value
Description:	Defines the default type of the note in the ticket phone inbound screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneInbound
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPhoneInbound'}->{'ArticleType'} = 'phone';</pre>

2.42.4. Ticket::Frontend::AgentTicketPhoneInbound###SenderType

Description	Value
Description:	Defines the default sender type for phone tickets in the ticket phone inbound screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneInbound
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPhoneInbound'}->{'SenderType'} = 'customer';</pre>

2.42.5. Ticket::Frontend::AgentTicketPhoneInbound###Subject

Description	Value
Description:	Defines the default subject for phone tickets in the ticket phone inbound screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneInbound
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPhoneInbound'}->{'Subject'} = '\$Text{"Phone call"}!';</pre>

2.42.6. Ticket::Frontend::AgentTicketPhoneInbound###Body

Description	Value
Description:	Defines the default note body text for phone tickets in the ticket phone inbound screen of the agent interface.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneInbound
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPhoneInbound'}->{'Body'} = '';</pre>

2.42.7. Ticket::Frontend::AgentTicketPhoneInbound###State

Description	Value
Description:	Defines the default ticket next state after adding a phone note in the ticket phone inbound screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneInbound
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPhoneInbound'}->{'State'} = 'open';</pre>

2.42.8. Ticket::Frontend::AgentTicketPhoneInbound###StateType

Description	Value
Description:	Next possible ticket states after adding a phone note in the ticket phone inbound screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneInbound
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPhoneInbound'}->{'StateType'} = ['open', 'pending auto', 'pending reminder', 'closed'];</pre>

2.42.9. Ticket::Frontend::AgentTicketPhoneInbound###HistoryType

Description	Value
Description:	Defines the history type for the ticket phone inbound screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneInbound
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPhoneInbound'}->{'HistoryType'} = 'PhoneCallCustomer';</pre>

2.42.10. Ticket::Frontend::AgentTicketPhoneInbound###HistoryComment

Description	Value
Description:	Defines the history comment for the ticket phone inbound screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneInbound
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPhoneInbound'}->{'HistoryComment'} = '';</pre>

2.42.11. Ticket::Frontend::AgentTicketPhoneInbound###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket phone inbound screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneInbound
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPhoneInbound'}->{'DynamicField'} = {};</pre>

2.43. Frontend::Agent::Ticket::ViewPhoneNew

2.43.1. Ticket::Frontend::AgentTicketPhone###Priority

Description	Value
Description:	Sets the default priority for new phone tickets in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneNew
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'Priority'} = '3 normal';</pre>

2.43.2. Ticket::Frontend::AgentTicketPhone###ArticleType

Description	Value
Description:	Sets the default article type for new phone tickets in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneNew
Valid:	1
Required:	1

Description	Value
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'ArticleType'} = 'phone';</code>

2.43.3. Ticket::Frontend::AgentTicketPhone###SenderType

Description	Value
Description:	Sets the default sender type for new phone ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'SenderType'} = 'customer';</code>

2.43.4. Ticket::Frontend::AgentTicketPhone::AllowMultipleFrom

Description	Value
Description:	Controls if more than one from entry can be set in the new phone ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhone::AllowMultipleFrom'} = '1';</code>

2.43.5. Ticket::Frontend::AgentTicketPhone###Subject

Description	Value
Description:	Sets the default subject for new phone tickets (e.g. 'Phone call') in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'Subject'} = '';</code>

2.43.6. Ticket::Frontend::AgentTicketPhone###Body

Description	Value
Description:	Sets the default note text for new telephone tickets. E.g 'New ticket via call' in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneNew
Valid:	1

Description	Value
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'Body'} = '';</code>

2.43.7. Ticket::Frontend::AgentTicketPhone###StateDefault

Description	Value
Description:	Sets the default next state for new phone tickets in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'StateDefault'} = 'open';</code>

2.43.8. Ticket::Frontend::AgentTicketPhone###StateType

Description	Value
Description:	Determines the next possible ticket states, after the creation of a new phone ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'StateType'} = ['open', 'pending auto', 'pending reminder', 'closed'];</code>

2.43.9. Ticket::Frontend::AgentTicketPhone###HistoryType

Description	Value
Description:	Defines the history type for the phone ticket screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'HistoryType'} = 'PhoneCallCustomer';</code>

2.43.10. Ticket::Frontend::AgentTicketPhone###HistoryComment

Description	Value
Description:	Defines the history comment for the phone ticket screen action, which gets used for ticket history in the agent interface.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneNew
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'HistoryComment'} = '';</pre>

2.43.11. Ticket::Frontend::AgentTicketPhone###SplitLinkType

Description	Value
Description:	Sets the default link type of splitted tickets in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneNew
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'SplitLinkType'} = { 'Direction' => 'Target', 'LinkType' => 'ParentChild' };</pre>

2.43.12. Ticket::Frontend::AgentTicketPhone###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket phone screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneNew
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPhone'}->{'DynamicField'} = {};</pre>

2.44. Frontend::Agent::Ticket::ViewPhoneOutbound

2.44.1. Ticket::Frontend::AgentTicketPhoneOutbound###Permission

Description	Value
Description:	Required permissions to use the ticket phone outbound screen in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneOutbound
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}->{'Permission'} = 'phone';</pre>

2.44.2. Ticket::Frontend::AgentTicketPhoneOutbound###RequiredLock

Description	Value
Description:	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).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneOutbound
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}->{'RequiredLock'} = '1';</pre>

2.44.3. Ticket::Frontend::AgentTicketPhoneOutbound###ArticleType

Description	Value
Description:	Defines the default type of the note in the ticket phone outbound screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneOutbound
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}->{'ArticleType'} = 'phone';</pre>

2.44.4. Ticket::Frontend::AgentTicketPhoneOutbound###SenderType

Description	Value
Description:	Defines the default sender type for phone tickets in the ticket phone outbound screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneOutbound
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}->{'SenderType'} = 'agent';</pre>

2.44.5. Ticket::Frontend::AgentTicketPhoneOutbound###Subject

Description	Value
Description:	Defines the default subject for phone tickets in the ticket phone outbound screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneOutbound
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<code>\$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}->{'Subject'} = '\$Text{"Phone call"}!';</code>

2.44.6. Ticket::Frontend::AgentTicketPhoneOutbound###Body

Description	Value
Description:	Defines the default note body text for phone tickets in the ticket phone outbound screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneOutbound
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}->{'Body'} = '';</code>

2.44.7. Ticket::Frontend::AgentTicketPhoneOutbound###State

Description	Value
Description:	Defines the default ticket next state after adding a phone note in the ticket phone outbound screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneOutbound
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}->{'State'} = 'closed successful';</code>

2.44.8. Ticket::Frontend::AgentTicketPhoneOutbound###StateType

Description	Value
Description:	Next possible ticket states after adding a phone note in the ticket phone outbound screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneOutbound
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}->{'StateType'} = ['open', 'pending auto', 'pending reminder', 'closed'];</code>

2.44.9. Ticket::Frontend::AgentTicketPhoneOutbound###HistoryType

Description	Value
Description:	Defines the history type for the ticket phone outbound screen action, which gets used for ticket history in the agent interface.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneOutbound
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}->{'HistoryType'} = 'PhoneCallAgent';</pre>

2.44.10. Ticket::Frontend::AgentTicketPhoneOutbound###HistoryComment

Description	Value
Description:	Defines the history comment for the ticket phone outbound screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneOutbound
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}->{'HistoryComment'} = '';</pre>

2.44.11. Ticket::Frontend::AgentTicketPhoneOutbound###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket phone outbound screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPhoneOutbound
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPhoneOutbound'}->{'DynamicField'} = {};</pre>

2.45. Frontend::Agent::Ticket::ViewPrint

2.45.1. Ticket::Frontend::AgentTicketPrint###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket print screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPrint
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPrint'}->{'DynamicField'} = {};</pre>

2.46. Frontend::Agent::Ticket::ViewPriority

2.46.1. Ticket::Frontend::AgentTicketPriority###Permission

Description	Value
Description:	Required permissions to use the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'Permission'} = 'priority';</pre>

2.46.2. Ticket::Frontend::AgentTicketPriority###RequiredLock

Description	Value
Description:	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).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'RequiredLock'} = '1';</pre>

2.46.3. Ticket::Frontend::AgentTicketPriority###TicketType

Description	Value
Description:	Sets the ticket type in the ticket priority screen of a zoomed ticket in the agent interface (Ticket::Type needs to be activated).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'TicketType'} = '0';</pre>

2.46.4. Ticket::Frontend::AgentTicketPriority###Service

Description	Value
Description:	Sets the service in the ticket priority screen of a zoomed ticket in the agent interface (Ticket::Service needs to be activated).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0

Description	Value
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'Service'} = '0';</code>

2.46.5. Ticket::Frontend::AgentTicketPriority###Owner

Description	Value
Description:	Sets the ticket owner in the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'Owner'} = '0';</code>

2.46.6. Ticket::Frontend::AgentTicketPriority###OwnerMandatory

Description	Value
Description:	Sets if ticket owner must be selected by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'OwnerMandatory'} = '0';</code>

2.46.7. Ticket::Frontend::AgentTicketPriority###Responsible

Description	Value
Description:	Sets the responsible agent of the ticket in the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'Responsible'} = '0';</code>

2.46.8. Ticket::Frontend::AgentTicketPriority###State

Description	Value
Description:	If a note is added by an agent, sets the state of the ticket in the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0

Description	Value
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'State'} = '0';</code>

2.46.9. Ticket::Frontend::AgentTicketPriority###StateType

Description	Value
Description:	Defines the next state of a ticket after adding a note, in the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'StateType'} = ['open', 'pending reminder', 'pending auto'];</code>

2.46.10. Ticket::Frontend::AgentTicketPriority###StateDefault

Description	Value
Description:	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.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'StateDefault'} = 'open';</code>

2.46.11. Ticket::Frontend::AgentTicketPriority###Note

Description	Value
Description:	Allows adding notes in the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'Note'} = '1';</code>

2.46.12. Ticket::Frontend::AgentTicketPriority###Subject

Description	Value
Description:	Sets the default subject for notes added in the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority

Description	Value
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'Subject'} = '\$Text{"Priority Update"}!';</code>

2.46.13. Ticket::Frontend::AgentTicketPriority###Body

Description	Value
Description:	Sets the default body text for notes added in the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'Body'} = '';</code>

2.46.14. Ticket::Frontend::AgentTicketPriority###InvolvedAgent

Description	Value
Description:	Shows a list of all the involved agents on this ticket, in the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'InvolvedAgent'} = '0';</code>

2.46.15. Ticket::Frontend::AgentTicketPriority###InformAgent

Description	Value
Description:	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.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'InformAgent'} = '0';</code>

2.46.16. Ticket::Frontend::AgentTicketPriority###ArticleTypeDefault

Description	Value
Description:	Defines the default type of the note in the ticket priority screen of a zoomed ticket in the agent interface.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'ArticleTypeDefault'} = 'note-internal';</pre>

2.46.17. Ticket::Frontend::AgentTicketPriority###ArticleTypes

Description	Value
Description:	Specifies the different note types that will be used in the system.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'ArticleTypes'} = { 'note-external' => '0', 'note-internal' => '1', 'note-report' => '0' };</pre>

2.46.18. Ticket::Frontend::AgentTicketPriority###Priority

Description	Value
Description:	Shows the ticket priority options in the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'Priority'} = '1';</pre>

2.46.19. Ticket::Frontend::AgentTicketPriority###PriorityDefault

Description	Value
Description:	Defines the default ticket priority in the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'PriorityDefault'} = '3 normal';</pre>

2.46.20. Ticket::Frontend::AgentTicketPriority###Title

Description	Value
Description:	Shows the title fields in the ticket priority screen of a zoomed ticket in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'Title'} = '0';</code>

2.46.21. Ticket::Frontend::AgentTicketPriority###HistoryType

Description	Value
Description:	Defines the history type for the ticket priority screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'HistoryType'} = 'AddNote';</code>

2.46.22. Ticket::Frontend::AgentTicketPriority###HistoryComment

Description	Value
Description:	Defines the history comment for the ticket priority screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'HistoryComment'} = '%Priority';</code>

2.46.23. Ticket::Frontend::AgentTicketPriority###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket priority screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewPriority
Valid:	1
Required:	0
Config-Setting:	

Description	Value
	<code>\$Self->{'Ticket::Frontend::AgentTicketPriority'}->{'DynamicField'} = {};</code>

2.47. Frontend::Agent::Ticket::ViewQueue

2.47.1. Ticket::Frontend::AgentTicketQueue###StripEmptyLines

Description	Value
Description:	Strips empty lines on the ticket preview in the queue view.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewQueue
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketQueue'}->{'StripEmptyLines'} = '0';</code>

2.47.2. Ticket::Frontend::AgentTicketQueue###ViewAllPossibleTickets

Description	Value
Description:	Shows all both ro and rw queues in the queue view.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewQueue
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketQueue'}->{'ViewAllPossibleTickets'} = '0';</code>

2.47.3. Ticket::Frontend::AgentTicketQueue###HighlightAge1

Description	Value
Description:	Sets the age in minutes (first level) for highlighting queues that contain untouched tickets.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewQueue
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketQueue'}->{'HighlightAge1'} = '1440';</code>

2.47.4. Ticket::Frontend::AgentTicketQueue###HighlightAge2

Description	Value
Description:	Sets the age in minutes (second level) for highlighting queues that contain untouched tickets.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewQueue
Valid:	1
Required:	1

Description	Value
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketQueue'}->{'HighlightAge2'} = '2880';</code>

2.47.5. Ticket::Frontend::AgentTicketQueue###Blink

Description	Value
Description:	Activates a blinking mechanism of the queue that contains the oldest ticket.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewQueue
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketQueue'}->{'Blink'} = '1';</code>

2.47.6. Ticket::Frontend::AgentTicketQueue###QueueSort

Description	Value
Description:	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.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewQueue
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketQueue'}->{'QueueSort'} = { '3' => '0', '7' => '1' };</code>

2.47.7. Ticket::Frontend::AgentTicketQueue###SortBy::Default

Description	Value
Description:	Defines the default sort criteria for all queues displayed in the queue view.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewQueue
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketQueue'}->{'SortBy::Default'} = 'Age';</code>

2.47.8. Ticket::Frontend::AgentTicketQueue###PreSort::ByPriority

Description	Value
Description:	Defines if a pre-sorting by priority should be done in the queue view.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewQueue
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketQueue'}->{'PreSort::ByPriority'} = '1';</pre>

2.47.9. Ticket::Frontend::AgentTicketQueue###Order::Default

Description	Value
Description:	Defines the default sort order for all queues in the queue view, after priority sort.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewQueue
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketQueue'}->{'Order::Default'} = 'Up';</pre>

2.48. Frontend::Agent::Ticket::ViewResponsible

2.48.1. Ticket::Frontend::AgentTicketResponsible###Permission

Description	Value
Description:	Required permissions to use the ticket responsible screen in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'Permission'} = 'responsible';</pre>

2.48.2. Ticket::Frontend::AgentTicketResponsible###RequiredLock

Description	Value
Description:	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).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'RequiredLock'} = '0';</pre>

2.48.3. Ticket::Frontend::AgentTicketResponsible###TicketType

Description	Value
Description:	Sets the ticket type in the ticket responsible screen of the agent interface (Ticket::Type needs to be activated).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'TicketType'} = '0';</code>

2.48.4. Ticket::Frontend::AgentTicketResponsible###Service

Description	Value
Description:	Sets the service in the ticket responsible screen of the agent interface (Ticket::Service needs to be activated).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'Service'} = '0';</code>

2.48.5. Ticket::Frontend::AgentTicketResponsible###Owner

Description	Value
Description:	Sets the ticket owner in the ticket responsible screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'Owner'} = '0';</code>

2.48.6. Ticket::Frontend::AgentTicketResponsible###OwnerMandatory

Description	Value
Description:	Sets if ticket owner must be selected by the agent.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'OwnerMandatory'} = '0';</code>

2.48.7. Ticket::Frontend::AgentTicketResponsible###Responsible

Description	Value
Description:	Sets the responsible agent of the ticket in the ticket responsible screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'Responsible'} = '1';</pre>

2.48.8. Ticket::Frontend::AgentTicketResponsible###State

Description	Value
Description:	If a note is added by an agent, sets the state of a ticket in the ticket responsible screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'State'} = '0';</pre>

2.48.9. Ticket::Frontend::AgentTicketResponsible###StateType

Description	Value
Description:	Defines the next state of a ticket after adding a note, in the ticket responsible screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'StateType'} = ['open', 'pending reminder', 'pending auto'];</pre>

2.48.10. Ticket::Frontend::AgentTicketResponsible###StateDefault

Description	Value
Description:	Defines the default next state of a ticket after adding a note, in the ticket responsible screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1

Description	Value
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'StateDefault'} = 'open';</pre>

2.48.11. Ticket::Frontend::AgentTicketResponsible###Note

Description	Value
Description:	Allows adding notes in the ticket responsible screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'Note'} = '1';</pre>

2.48.12. Ticket::Frontend::AgentTicketResponsible###Subject

Description	Value
Description:	Sets the default subject for notes added in the ticket responsible screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'Subject'} = '\$Text{"Responsible Update"}!';</pre>

2.48.13. Ticket::Frontend::AgentTicketResponsible###Body

Description	Value
Description:	Sets the default body text for notes added in the ticket responsible screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'Body'} = '';</pre>

2.48.14. Ticket::Frontend::AgentTicketResponsible###InvolvedAgent

Description	Value
Description:	Shows a list of all the involved agents on this ticket, in the ticket responsible screen of the agent interface.
Group:	Ticket

Description	Value
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}- >{'InvolvedAgent'} = '0';</pre>

2.48.15. Ticket::Frontend::AgentTicketResponsible###InformAgent

Description	Value
Description:	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.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}- >{'InformAgent'} = '0';</pre>

2.48.16. Ticket::Frontend::AgentTicketResponsible###ArticleTypeDefault

Description	Value
Description:	Defines the default type of the note in the ticket responsible screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}- >{'ArticleTypeDefault'} = 'note-internal';</pre>

2.48.17. Ticket::Frontend::AgentTicketResponsible###ArticleTypes

Description	Value
Description:	Specifies the different note types that will be used in the system.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}- >{'ArticleTypes'} = { 'note-external' => '0', 'note-internal' => '1', 'note-report' => '0' };</pre>

2.48.18. Ticket::Frontend::AgentTicketResponsible###Priority

Description	Value
Description:	Shows the ticket priority options in the ticket responsible screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'Priority'} = '0';</pre>

2.48.19. Ticket::Frontend::AgentTicketResponsible###PriorityDefault

Description	Value
Description:	Defines the default ticket priority in the ticket responsible screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'PriorityDefault'} = '3 normal';</pre>

2.48.20. Ticket::Frontend::AgentTicketResponsible###Title

Description	Value
Description:	Shows the title fields in the ticket responsible screen of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'Title'} = '1';</pre>

2.48.21. Ticket::Frontend::AgentTicketResponsible###HistoryType

Description	Value
Description:	Defines the history type for the ticket responsible screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'HistoryType'} = 'AddNote';</pre>

2.48.22. Ticket::Frontend::AgentTicketResponsible###HistoryComment

Description	Value
Description:	Defines the history comment for the ticket responsible screen action, which gets used for ticket history in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'HistoryComment'} = '%Responsible';</pre>

2.48.23. Ticket::Frontend::AgentTicketResponsible###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket responsible screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled, 2 = Enabled and required.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewResponsible
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketResponsible'}->{'DynamicField'} = {};</pre>

2.49. Frontend::Agent::Ticket::ViewSearch

2.49.1. Ticket::Frontend::AgentTicketSearch###ExtendedSearchCondition

Description	Value
Description:	Allows extended search conditions in ticket search of the agent interface. With this feature you can search e. g. with this kind of conditions like "(key1&&key2)" or "(key1 key2)".
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'ExtendedSearchCondition'} = '1';</pre>

2.49.2. Ticket::Frontend::AgentTicketSearch###SearchLimit

Description	Value
Description:	Maximum number of tickets to be displayed in the result of a search in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1

Description	Value
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'SearchLimit'} = '2000';</code>

2.49.3. Ticket::Frontend::AgentTicketSearch###SearchPageShown

Description	Value
Description:	Number of tickets to be displayed in each page of a search result in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'SearchPageShown'} = '40';</code>

2.49.4. Ticket::Frontend::AgentTicketSearch###SearchViewableTicketLines

Description	Value
Description:	Number of lines (per ticket) that are shown by the search utility in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'SearchViewableTicketLines'} = '10';</code>

2.49.5. Ticket::Frontend::AgentTicketSearch###SortBy::Default

Description	Value
Description:	Defines the default ticket attribute for ticket sorting of the ticket search result of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'SortBy::Default'} = 'Age';</code>

2.49.6. Ticket::Frontend::AgentTicketSearch###Order::Default

Description	Value
Description:	Defines the default ticket order in the ticket search result of the agent interface. Up: oldest on top. Down: latest on top.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Order::Default'} = 'Down';</code>

2.49.7. Ticket::Frontend::AgentTicketSearch###SearchArticleCSVTree

Description	Value
Description:	Exports the whole article tree in search result (it can affect the system performance).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'SearchArticleCSVTree'} = '0';</code>

2.49.8. Ticket::Frontend::AgentTicketSearch###SearchCSVData

Description	Value
Description:	Data used to export the search result in CSV format.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	<code>\$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'];</code>

2.49.9. Ticket::Frontend::AgentTicketSearch###ArticleCreateTime

Description	Value
Description:	Includes article create times in the ticket search of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'ArticleCreateTime'} = '0';</pre>

2.49.10. Ticket::Frontend::AgentTicketSearch###Defaults###Fulltext

Description	Value
Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'Fulltext'} = '';</pre>

2.49.11. Ticket::Frontend::AgentTicketSearch###Defaults###TicketNumber

Description	Value
Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'TicketNumber'} = '';</pre>

2.49.12. Ticket::Frontend::AgentTicketSearch###Defaults###Title

Description	Value
Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'Title'} = '';</pre>

2.49.13. Ticket::Frontend::AgentTicketSearch###Defaults###From

Description	Value
Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'From'} = '';</pre>

2.49.14. Ticket::Frontend::AgentTicketSearch###Defaults###To

Description	Value
Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'To'} = '';</pre>

2.49.15. Ticket::Frontend::AgentTicketSearch###Defaults###Cc

Description	Value
Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'Cc'} = '';</pre>

2.49.16. Ticket::Frontend::AgentTicketSearch###Defaults###Subject

Description	Value
Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'Subject'} = '';</pre>

2.49.17. Ticket::Frontend::AgentTicketSearch###Defaults###Body

Description	Value
Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'Body'} = '';</pre>

2.49.18. Ticket::Frontend::AgentTicketSearch###Defaults###CustomerID

Description	Value
Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'CustomerID'} = '';</pre>

2.49.19. Ticket::Frontend::AgentTicketSearch###Defaults###CustomerUserLogin

Description	Value
Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'CustomerUserLogin'} = '';</pre>

2.49.20. Ticket::Frontend::AgentTicketSearch###Defaults###StateIDs

Description	Value
Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	

Description	Value
	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'StateIDs'} = [];</code>

2.49.21. Ticket::Frontend::AgentTicketSearch###Defaults###QueueIDs

Description	Value
Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'QueueIDs'} = [];</code>

2.49.22. Ticket::Frontend::AgentTicketSearch###Defaults###TicketCreateTimePoint

Description	Value
Description:	Default data to use on attribute for ticket search screen. Example: "TicketCreateTimePointFormat=year;TicketCreateTimePointStart=Last;TicketCre
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'TicketCreateTimePoint'} = '';</code>

2.49.23. Ticket::Frontend::AgentTicketSearch###Defaults###TicketCreateTimeSlot

Description	Value
Description:	Default data to use on attribute for ticket search screen. Example: "TicketCreateTimeStartYear=2010;TicketCreateTimeStartMonth=10;TicketCre
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'TicketCreateTimeSlot'} = '';</code>

2.49.24. Ticket::Frontend::AgentTicketSearch###Defaults###TicketChangeTimePoint

Description	Value
Description:	Defines the default shown ticket search attribute for ticket search screen.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'TicketChangeTimePoint'} = '';</pre>

2.49.25. Ticket::Frontend::AgentTicketSearch###Defaults###TicketChangeTimeSlot

Description	Value
Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'TicketChangeTimeSlot'} = '';</pre>

2.49.26. Ticket::Frontend::AgentTicketSearch###Defaults###TicketCloseTimePoint

Description	Value
Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'TicketCloseTimePoint'} = '';</pre>

2.49.27. Ticket::Frontend::AgentTicketSearch###Defaults###TicketCloseTimeSlot

Description	Value
Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'TicketCloseTimeSlot'} = '';</pre>

2.49.28.
Ticket::Frontend::AgentTicketSearch###Defaults###ArticleCreateTimePoint

Description	Value
Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'ArticleCreateTimePoint'} = '';</pre>

2.49.29.
Ticket::Frontend::AgentTicketSearch###Defaults###ArticleCreateTimeSlot

Description	Value
Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'ArticleCreateTimeSlot'} = '';</pre>

2.49.30. Ticket::Frontend::AgentTicketSearch###Defaults###SearchInArchive

Description	Value
Description:	Defines the default shown ticket search attribute for ticket search screen.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'SearchInArchive'} = '';</pre>

2.49.31. Ticket::Frontend::CustomerTicketSearch###SearchArticleCSVTree

Description	Value
Description:	Exports the whole article tree in search result (it can affect the system performance).
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<code>\$Self->{'Ticket::Frontend::CustomerTicketSearch'}->{'SearchArticleCSVTree'} = '0';</code>

2.49.32. Ticket::Frontend::AgentTicketSearch###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket search screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'DynamicField'} = {};</code>

2.49.33. Ticket::Frontend::AgentTicketSearch###Defaults###DynamicField

Description	Value
Description:	Defines the default shown ticket search attribute for ticket search screen. Example: a text, 1, Search_DynamicField_Field1StartYear=2002; Search_DynamicField_Field1StartMonth=12; Search_DynamicField_Field1StartDay=12; Search_DynamicField_Field1StartHour=00; Search_DynamicField_Field1StartMinute=00; Search_DynamicField_Field1StartSecond=00; Search_DynamicField_Field1StopYear=2009; Search_DynamicField_Field1StopMonth=02; Search_DynamicField_Field1StopDay=10; Search_DynamicField_Field1StopHour=23; Search_DynamicField_Field1StopMinute=59; Search_DynamicField_Field1StopSecond=59;.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'Defaults'}->{'DynamicField'} = {};</code>

2.49.34. Ticket::Frontend::AgentTicketSearch###SearchCSVDynamicField

Description	Value
Description:	Dynamic Fields used to export the search result in CSV format.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewSearch
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::AgentTicketSearch'}->{'SearchCSVDynamicField'} = {};</code>

2.50. Frontend::Agent::Ticket::ViewStatus

2.50.1. Ticket::Frontend::AgentTicketStatusView###ViewableTicketsPage

Description	Value
Description:	Shows all open tickets (even if they are locked) in the status view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewStatus
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketStatusView'}->{'ViewableTicketsPage'} = '50';</pre>

2.50.2. Ticket::Frontend::AgentTicketStatusView###SortBy::Default

Description	Value
Description:	Defines the default ticket attribute for ticket sorting in the status view of the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewStatus
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketStatusView'}->{'SortBy::Default'} = 'Age';</pre>

2.50.3. Ticket::Frontend::AgentTicketStatusView###Order::Default

Description	Value
Description:	Defines the default ticket order (after priority sort) in the status view of the agent interface. Up: oldest on top. Down: latest on top.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewStatus
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketStatusView'}->{'Order::Default'} = 'Down';</pre>

2.51. Frontend::Agent::Ticket::ViewZoom

2.51.1. Ticket::Frontend::PlainView

Description	Value
Description:	Shows a link to see a zoomed email ticket in plain text.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewZoom
Valid:	1
Required:	1

Description	Value
Config-Setting:	<code>\$Self->{'Ticket::Frontend::PlainView'} = '0';</code>

2.51.2. Ticket::Frontend::ZoomExpand

Description	Value
Description:	Shows all the articles of the ticket (expanded) in the zoom view.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::ZoomExpand'} = '0';</code>

2.51.3. Ticket::Frontend::ZoomExpandSort

Description	Value
Description:	Shows the articles sorted normally or in reverse, under ticket zoom in the agent interface.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::ZoomExpandSort'} = 'normal';</code>

2.51.4. Ticket::ZoomAttachmentDisplayCount

Description	Value
Description:	Shows a count of icons in the ticket zoom, if the article has attachments.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::ZoomAttachmentDisplayCount'} = '20';</code>

2.51.5. Ticket::ZoomTimeDisplay

Description	Value
Description:	Displays the accounted time for an article in the ticket zoom view.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewZoom
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::ZoomTimeDisplay'} = '0';</code>

2.51.6. Ticket::UseArticleColors

Description	Value
Description:	Shows colors for different article types in the article table.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewZoom
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::UseArticleColors'} = '0';</code>

2.51.7. Ticket::Frontend::TicketArticleFilter

Description	Value
Description:	Activates the article filter in the zoom view to specify which articles should be shown.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::TicketArticleFilter'} = '0';</code>

2.51.8. Ticket::Frontend::HTMLArticleHeightDefault

Description	Value
Description:	Set the default height (in pixels) of inline HTML articles in AgentTicketZoom.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::HTMLArticleHeightDefault'} = '100';</code>

2.51.9. Ticket::Frontend::HTMLArticleHeightMax

Description	Value
Description:	Set the maximum height (in pixels) of inline HTML articles in AgentTicketZoom.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::HTMLArticleHeightMax'} = '2500';</code>

2.51.10. Ticket::Frontend::ZoomRichTextForce

Description	Value
Description:	Show article as rich text even if rich text writing is disabled.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::ZoomRichTextForce'} = '0';</pre>

2.51.11. Ticket::Frontend::AgentTicketZoom###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket zoom screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled.
Group:	Ticket
SubGroup:	Frontend::Agent::Ticket::ViewZoom
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::AgentTicketZoom'}->{'DynamicField'} = {};</pre>

2.52. Frontend::Agent::TicketOverview

2.52.1. Ticket::Frontend::Overview###Small

Description	Value
Description:	Allows having a small format ticket overview (CustomerInfo => 1 - shows also the customer information).
Group:	Ticket
SubGroup:	Frontend::Agent::TicketOverview
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::Overview'}->{'Small'} = { 'CustomerInfo' => '1', 'Module' => 'Kernel::Output::HTML::TicketOverviewSmall', 'Name' => 'Small', 'NameShort' => 'S' };</pre>

2.52.2. Ticket::Frontend::OverviewSmall###ColumnHeader

Description	Value
Description:	Shows either the last customer article's subject or the ticket title in the small format overview.
Group:	Ticket
SubGroup:	Frontend::Agent::TicketOverview
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::OverviewSmall'}->{'ColumnHeader'} = 'LastCustomerSubject';</pre>

2.52.3. Ticket::Frontend::Overview###Medium

Description	Value
Description:	Allows having a medium format ticket overview (CustomerInfo => 1 - shows also the customer information).
Group:	Ticket
SubGroup:	Frontend::Agent::TicketOverview
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::Overview'}->{'Medium'} = { 'CustomerInfo' => '0', 'Module' => 'Kernel::Output::HTML::TicketOverviewMedium', 'Name' => 'Medium', 'NameShort' => 'M', 'TicketActionsPerTicket' => '0' };</pre>

2.52.4. Ticket::Frontend::Overview###Preview

Description	Value
Description:	Shows a preview of the ticket overview (CustomerInfo => 1 - shows also Customer-Info, CustomerInfoMaxSize max. size in characters of Customer-Info).
Group:	Ticket
SubGroup:	Frontend::Agent::TicketOverview
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::Overview'}->{'Preview'} = { 'CustomerInfo' => '0', 'CustomerInfoMaxSize' => '18', 'DefaultPreViewLines' => '25', 'DefaultViewNewLine' => '90', 'Module' => 'Kernel::Output::HTML::TicketOverviewPreview', 'Name' => 'Preview', 'NameShort' => 'L', 'StripEmptyLines' => '0', 'TicketActionsPerTicket' => '0' };</pre>

2.52.5. Ticket::Frontend::Overview::PreviewArticleSenderTypes

Description	Value
Description:	Defines which article sender types should be shown in the preview of a ticket.
Group:	Ticket
SubGroup:	Frontend::Agent::TicketOverview
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::Overview::PreviewArticleSenderTypes'} = { 'agent' => '1', 'customer' => '1', 'system' => '1' };</pre>

2.52.6. Ticket::Frontend::Overview::PreviewArticleTypeExpanded

Description	Value
Description:	Defines wich article type should be expanded when entering the overview. If nothing defined, latest article will be expanded.
Group:	Ticket
SubGroup:	Frontend::Agent::TicketOverview
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::Overview::PreviewArticleTypeExpanded'} = '';</pre>

2.52.7. Ticket::Frontend::OverviewSmall###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket small format overview screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled.
Group:	Ticket
SubGroup:	Frontend::Agent::TicketOverview
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::OverviewSmall'}->{'DynamicField'} = {};</pre>

2.52.8. Ticket::Frontend::OverviewMedium###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket medium format overview screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled.
Group:	Ticket
SubGroup:	Frontend::Agent::TicketOverview
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::OverviewMedium'}->{'DynamicField'} = {};</pre>

2.52.9. Ticket::Frontend::OverviewPreview###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket preview format overview screen of the agent interface. Possible settings: 0 = Disabled, 1 = Enabled.
Group:	Ticket
SubGroup:	Frontend::Agent::TicketOverview
Valid:	1
Required:	0
Config-Setting:	

Description	Value
	<code>\$Self->{'Ticket::Frontend::OverviewPreview'}->{'DynamicField'} = {};</code>

2.53. Frontend::Agent::ToolBarModule

2.53.1. Frontend::ToolBarModule###1-Ticket::AgentTicketQueue

Description	Value
Description:	Toolbar Item for a shortcut.
Group:	Ticket
SubGroup:	Frontend::Agent::ToolBarModule
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::ToolBarModule'}->{'1-Ticket::AgentTicketQueue'} = { 'AccessKey' => 'q', 'Action' => 'AgentTicketQueue', 'CssClass' => 'QueueView', 'Link' => 'Action=AgentTicketQueue', 'Module' => 'Kernel::Output::HTML::ToolBarLink', 'Name' => 'Queue view', 'Priority' => '1010010' }; </pre>

2.53.2. Frontend::ToolBarModule###2-Ticket::AgentTicketStatus

Description	Value
Description:	Toolbar Item for a shortcut.
Group:	Ticket
SubGroup:	Frontend::Agent::ToolBarModule
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::ToolBarModule'}->{'2-Ticket::AgentTicketStatus'} = { 'AccessKey' => 'o', 'Action' => 'AgentTicketStatusView', 'CssClass' => 'StatusView', 'Link' => 'Action=AgentTicketStatusView', 'Module' => 'Kernel::Output::HTML::ToolBarLink', 'Name' => 'Status view', 'Priority' => '1010020' }; </pre>

2.53.3. Frontend::ToolBarModule###3-Ticket::AgentTicketEscalation

Description	Value
Description:	Toolbar Item for a shortcut.
Group:	Ticket
SubGroup:	Frontend::Agent::ToolBarModule
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::ToolBarModule'}->{'3- Ticket::AgentTicketEscalation'} = { </pre>

Description	Value
	<pre>'AccessKey' => 'w', 'Action' => 'AgentTicketEscalationView', 'CssClass' => 'EscalationView', 'Link' => 'Action=AgentTicketEscalationView', 'Module' => 'Kernel::Output::HTML::ToolBarLink', 'Name' => 'Escalation view', 'Priority' => '1010030' };</pre>

2.53.4. Frontend::ToolBarModule###4-Ticket::AgentTicketPhone

Description	Value
Description:	Toolbar Item for a shortcut.
Group:	Ticket
SubGroup:	Frontend::Agent::ToolBarModule
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::ToolBarModule'}->{'4-Ticket::AgentTicketPhone'} = { 'AccessKey' => 'l', 'Action' => 'AgentTicketPhone', 'CssClass' => 'PhoneTicket', 'Link' => 'Action=AgentTicketPhone', 'Module' => 'Kernel::Output::HTML::ToolBarLink', 'Name' => 'New phone ticket', 'Priority' => '1020010' };</pre>

2.53.5. Frontend::ToolBarModule###5-Ticket::AgentTicketEmail

Description	Value
Description:	Toolbar Item for a shortcut.
Group:	Ticket
SubGroup:	Frontend::Agent::ToolBarModule
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::ToolBarModule'}->{'5-Ticket::AgentTicketEmail'} = { 'AccessKey' => 'l', 'Action' => 'AgentTicketEmail', 'CssClass' => 'EmailTicket', 'Link' => 'Action=AgentTicketEmail', 'Module' => 'Kernel::Output::HTML::ToolBarLink', 'Name' => 'New email ticket', 'Priority' => '1020020' };</pre>

2.53.6. Frontend::ToolBarModule###6-Ticket::TicketResponsible

Description	Value
Description:	Agent interface notification module to see the number of tickets an agent is responsible for.
Group:	Ticket
SubGroup:	Frontend::Agent::ToolBarModule

Description	Value
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::ToolBarModule'}->{'6-Ticket::TicketResponsible'} = { 'CssClass' => 'Responsible', 'CssClassNew' => 'Responsible New', 'CssClassReached' => 'Responsible Reached', 'Module' => 'Kernel::Output::HTML::ToolBarTicketResponsible', 'Priority' => '1030010' };</pre>

2.53.7. Frontend::ToolBarModule###7-Ticket::TicketWatcher

Description	Value
Description:	Agent interface notification module to see the number of watched tickets.
Group:	Ticket
SubGroup:	Frontend::Agent::ToolBarModule
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::ToolBarModule'}->{'7-Ticket::TicketWatcher'} = { 'CssClass' => 'Watcher', 'CssClassNew' => 'Watcher New', 'CssClassReached' => 'Watcher Reached', 'Module' => 'Kernel::Output::HTML::ToolBarTicketWatcher', 'Priority' => '1030020' };</pre>

2.53.8. Frontend::ToolBarModule###8-Ticket::TicketLocked

Description	Value
Description:	Agent interface notification module to check the used charset.
Group:	Ticket
SubGroup:	Frontend::Agent::ToolBarModule
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::ToolBarModule'}->{'8-Ticket::TicketLocked'} = { 'CssClass' => 'Locked', 'CssClassNew' => 'Locked New', 'CssClassReached' => 'Locked Reached', 'Module' => 'Kernel::Output::HTML::ToolBarTicketLocked', 'Priority' => '1030030' };</pre>

2.53.9. Frontend::ToolBarModule###9-Ticket::TicketSearchProfile

Description	Value
Description:	Agent interface module to access search profiles via nav bar.
Group:	Ticket
SubGroup:	Frontend::Agent::ToolBarModule
Valid:	0

Description	Value
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::ToolBarModule'}->{'9-Ticket::TicketSearchProfile'} = { 'Block' => 'ToolBarSearchProfile', 'Description' => 'Search-Template', 'MaxWidth' => '40', 'Module' => 'Kernel::Output::HTML::ToolBarTicketSearchProfile', 'Name' => 'Search-Template', 'Priority' => '1990010' };</pre>

2.53.10. Frontend::ToolBarModule###10-Ticket::TicketSearchFulltext

Description	Value
Description:	Agent interface module to access fulltext search via nav bar.
Group:	Ticket
SubGroup:	Frontend::Agent::ToolBarModule
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::ToolBarModule'}->{'10-Ticket::TicketSearchFulltext'} = { 'Block' => 'ToolBarSearchFulltext', 'CSS' => 'Core.Agent.Toolbar.FulltextSearch.css', 'Description' => 'Fulltext-Search', 'Module' => 'Kernel::Output::HTML::ToolBarTicketSearchFulltext', 'Name' => 'Fulltext-Search', 'Priority' => '1990020', 'Size' => '10' };</pre>

2.54. Frontend::Customer

2.54.1. Ticket::Frontend::CustomerTicketOverviewSortable

Description	Value
Description:	Controls if customers have the ability to sort their tickets.
Group:	Ticket
SubGroup:	Frontend::Customer
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerTicketOverviewSortable'} = '';</pre>

2.54.2. Ticket::Frontend::CustomerTicketOverviewCustomEmptyText

Description	Value
Description:	Custom text for the page shown to customers that have no tickets yet.
Group:	Ticket
SubGroup:	Frontend::Customer
Valid:	0
Required:	0

Description	Value
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerTicketOverviewCustomEmptyText'} = { 'Button' => 'Create your first ticket', 'Text' => 'Please click the button below to create your first ticket.', 'Title' => 'Welcome!' };</pre>

2.54.3. Ticket::Frontend::CustomerTicketOverview###ColumnHeader

Description	Value
Description:	Shows either the last customer article's subject or the ticket title in the small format overview.
Group:	Ticket
SubGroup:	Frontend::Customer
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerTicketOverview'}->{'ColumnHeader'} = 'TicketTitle';</pre>

2.54.4. Frontend::CustomerUser::Item###15-OpenTickets

Description	Value
Description:	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.
Group:	Ticket
SubGroup:	Frontend::Customer
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Frontend::CustomerUser::Item'}->{'15-OpenTickets'} = { 'Action' => 'AgentTicketSearch', 'Attributes' => 'StateType=Open;', 'CSS' => 'Core.Agent.CustomerUser.OpenTicket.css', 'CSSClassNoOpenTicket' => 'NoOpenTicket', 'CSSClassOpenTicket' => 'OpenTicket', 'CustomerUserLogin' => '0', 'Module' => 'Kernel::Output::HTML::CustomerUserGenericTicket', 'Subaction' => 'Search', 'Target' => '_blank', 'Text' => 'Open tickets' };</pre>

2.54.5. Frontend::CustomerUser::Item###16-OpenTicketsForCustomerUserLogin

Description	Value
Description:	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.
Group:	Ticket
SubGroup:	Frontend::Customer

Description	Value
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::CustomerUser::Item'}->{'16-OpenTicketsForCustomerUserLogin'} = { 'Action' => 'AgentTicketSearch', 'Attributes' => 'StateType=Open;', 'CSS' => 'Core.Agent.CustomerUser.OpenTicket.css', 'CSSClassNoOpenTicket' => 'NoOpenTicket', 'CSSClassOpenTicket' => 'OpenTicket', 'CustomerUserLogin' => '1', 'Module' => 'Kernel::Output::HTML::CustomerUserGenericTicket', 'Subaction' => 'Search', 'Target' => '_blank', 'Text' => 'Open tickets of customer' }; </pre>

2.54.6. Frontend::CustomerUser::Item###17-ClosedTickets

Description	Value
Description:	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.
Group:	Ticket
SubGroup:	Frontend::Customer
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::CustomerUser::Item'}->{'17-ClosedTickets'} = { 'Action' => 'AgentTicketSearch', 'Attributes' => 'StateType=Closed;', 'CSS' => 'Core.Agent.CustomerUser.OpenTicket.css', 'CSSClassNoOpenTicket' => 'NoOpenTicket', 'CSSClassOpenTicket' => 'OpenTicket', 'CustomerUserLogin' => '0', 'Module' => 'Kernel::Output::HTML::CustomerUserGenericTicket', 'Subaction' => 'Search', 'Target' => '_blank', 'Text' => 'Closed tickets' }; </pre>

2.54.7. Frontend::CustomerUser::Item###18-ClosedTicketsForCustomerUserLogin

Description	Value
Description:	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.
Group:	Ticket
SubGroup:	Frontend::Customer
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'Frontend::CustomerUser::Item'}->{'18-ClosedTicketsForCustomerUserLogin'} = { 'Action' => 'AgentTicketSearch', 'Attributes' => 'StateType=Closed;', </pre>

Description	Value
	<pre>'CSS' => 'Core.Agent.CustomerUser.OpenTicket.css', 'CSSClassNoOpenTicket' => 'NoOpenTicket', 'CSSClassOpenTicket' => 'OpenTicket', 'CustomerUserLogin' => '1', 'Module' => 'Kernel::Output::HTML::CustomerUserGenericTicket', 'Subaction' => 'Search', 'Target' => '_blank', 'Text' => 'Closed tickets of customer' };</pre>

2.54.8. CustomerFrontend::CommonObject###QueueObject

Description	Value
Description:	Path of the file that stores all the settings for the QueueObject object for the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'CustomerFrontend::CommonObject'}->{'QueueObject'} = 'Kernel::System::Queue';</pre>

2.54.9. CustomerFrontend::CommonObject###TicketObject

Description	Value
Description:	Path of the file that stores all the settings for the TicketObject for the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'CustomerFrontend::CommonObject'}->{'TicketObject'} = 'Kernel::System::Ticket';</pre>

2.54.10. CustomerFrontend::CommonParam###Action

Description	Value
Description:	Defines the default used Frontend-Module if no Action parameter given in the url on the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'CustomerFrontend::CommonParam'}->{'Action'} = 'CustomerTicketOverview';</pre>

2.54.11. CustomerFrontend::CommonParam###TicketID

Description	Value
Description:	Default ticket ID used by the system in the customer interface.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Customer
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'CustomerFrontend::CommonParam'}->{'TicketID'} = '';</pre>

2.55. Frontend::Customer::ModuleMetaHead

2.55.1. CustomerFrontend::HeaderMetaModule###2-TicketSearch

Description	Value
Description:	Module to generate html OpenSearch profile for short ticket search in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::ModuleMetaHead
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerFrontend::HeaderMetaModule'}->{'2-TicketSearch'} = { 'Action' => 'CustomerTicketSearch', 'Module' => 'Kernel::Output::HTML::CustomerHeaderMetaTicketSearch' };</pre>

2.56. Frontend::Customer::ModuleRegistration

2.56.1. CustomerFrontend::Module###CustomerTicketOverview

Description	Value
Description:	Frontend module registration for the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerFrontend::Module'}->{'CustomerTicketOverview'} = { 'Description' => 'Overview of customer tickets', 'NavBar' => [{ 'AccessKey' => 'm', 'Block' => '', 'Description' => 'My Tickets', 'Link' => 'Action=CustomerTicketOverview;Subaction=MyTickets', 'LinkOption' => '', 'Name' => 'My Tickets', 'NavBar' => '', 'Prio' => '110', 'Type' => '' }, { 'AccessKey' => 'c', 'Block' => '', 'Description' => 'Company Tickets', 'Link' => 'Action=CustomerTicketOverview;Subaction=CompanyTickets',</pre>

Description	Value
	<pre> 'LinkOption' => '', 'Name' => 'Company Tickets', 'NavBar' => '', 'Prio' => '120', 'Type' => '' }], 'NavBarName' => 'Ticket', 'Title' => 'Overview' }; </pre>

2.56.2. CustomerFrontend::Module###CustomerTicketMessage

Description	Value
Description:	Frontend module registration for the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'CustomerFrontend::Module'}->{'CustomerTicketMessage'} = { 'Description' => 'Create tickets', 'NavBar' => [{ 'AccessKey' => 'n', 'Block' => '', 'Description' => 'Create new Ticket', 'Link' => 'Action=CustomerTicketMessage', 'LinkOption' => '', 'Name' => 'New Ticket', 'NavBar' => '', 'Prio' => '100', 'Type' => '' }], 'NavBarName' => 'Ticket', 'Title' => 'New Ticket' }; </pre>

2.56.3. CustomerFrontend::Module###CustomerTicketZoom

Description	Value
Description:	Frontend module registration for the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre> \$self->{'CustomerFrontend::Module'}->{'CustomerTicketZoom'} = { 'Description' => 'Ticket zoom view', 'Loader' => { 'JavaScript' => ['Core.Customer.TicketZoom.js', 'Core.UI.Popup.js'] }, 'NavBarName' => 'Ticket', 'Title' => 'Zoom' }; </pre>

2.56.4. CustomerFrontend::Module###CustomerTicketPrint

Description	Value
Description:	Frontend module registration for the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerFrontend::Module'}->{'CustomerTicketPrint'} = { 'Description' => 'Customer Ticket Print Module', 'NavBarName' => '', 'Title' => 'Print' };</pre>

2.56.5. CustomerFrontend::Module###CustomerTicketAttachment

Description	Value
Description:	Frontend module registration for the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerFrontend::Module'}->{'CustomerTicketAttachment'} = { 'Description' => 'To download attachments', 'NavBarName' => '', 'Title' => '' };</pre>

2.56.6. CustomerFrontend::Module###CustomerTicketSearch

Description	Value
Description:	Frontend module registration for the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::ModuleRegistration
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerFrontend::Module'}->{'CustomerTicketSearch'} = { 'Description' => 'Customer ticket search', 'NavBar' => [{ 'AccessKey' => 's', 'Block' => '', 'Description' => 'Search', 'Link' => 'Action=CustomerTicketSearch', 'LinkOption' => '', 'Name' => 'Search', 'NavBar' => '', 'Prio' => '300', 'Type' => '' }], 'NavBarName' => 'Ticket', 'Title' => 'Search' };</pre>

Description	Value
	};

2.57. Frontend::Customer::Preferences

2.57.1. CustomerPreferencesGroups###ShownTickets

Description	Value
Description:	Defines all the parameters for the ShownTickets object in the customer preferences of the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerPreferencesGroups'}->{'ShownTickets'} = { 'Active' => '1', 'Column' => 'User Profile', 'Data' => { '15' => '15', '20' => '20', '25' => '25', '30' => '30' }, 'DataSelected' => '25', 'Key' => 'Tickets per page', 'Label' => 'Number of displayed tickets', 'Module' => 'Kernel::Output::HTML::PreferencesGeneric', 'PrefKey' => 'UserShowTickets', 'Prio' => '4000' };</pre>

2.57.2. CustomerPreferencesGroups###RefreshTime

Description	Value
Description:	Defines all the parameters for the RefreshTime object in the customer preferences of the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Preferences
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'CustomerPreferencesGroups'}->{'RefreshTime'} = { 'Active' => '1', 'Column' => 'User Profile', 'Data' => { '' => 'off', '10' => '10 minutes', '15' => '15 minutes', '2' => ' 2 minutes', '5' => ' 5 minutes', '7' => ' 7 minutes' }, 'DataSelected' => '', 'Key' => 'Refresh interval', 'Label' => 'Ticket overview', 'Module' => 'Kernel::Output::HTML::PreferencesGeneric', 'PrefKey' => 'UserRefreshTime', 'Prio' => '4000' };</pre>

2.58. Frontend::Customer::Ticket::ViewNew

2.58.1.

Ticket::Frontend::CustomerTicketMessage###NextScreenAfterNewTicket

Description	Value
Description:	Determines the next screen after new customer ticket in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'NextScreenAfterNewTicket'} = 'CustomerTicketOverview';</pre>

2.58.2. Ticket::Frontend::CustomerTicketMessage###Priority

Description	Value
Description:	Allows customers to set the ticket priority in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'Priority'} = '1';</pre>

2.58.3. Ticket::Frontend::CustomerTicketMessage###PriorityDefault

Description	Value
Description:	Defines the default priority of new customer tickets in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'PriorityDefault'} = '3 normal';</pre>

2.58.4. Ticket::Frontend::CustomerTicketMessage###Queue

Description	Value
Description:	Allows customers to set the ticket queue in the customer interface. If this is set to 'No', QueueDefault should be configured.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	1

Description	Value
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'Queue'} = '1';</code>

2.58.5. Ticket::Frontend::CustomerTicketMessage###QueueDefault

Description	Value
Description:	Defines the default queue for new customer tickets in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'QueueDefault'} = 'Postmaster';</code>

2.58.6. Ticket::Frontend::CustomerTicketMessage###Service

Description	Value
Description:	Allows customers to set the ticket service in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'Service'} = '1';</code>

2.58.7. Ticket::Frontend::CustomerTicketMessage###SLA

Description	Value
Description:	Allows customers to set the ticket SLA in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'SLA'} = '1';</code>

2.58.8. Ticket::Frontend::CustomerTicketMessage###StateDefault

Description	Value
Description:	Defines the default state of new customer tickets in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	1

Description	Value
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'StateDefault'} = 'new';</pre>

2.58.9. Ticket::Frontend::CustomerTicketMessage###ArticleType

Description	Value
Description:	Defines the default type for article in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'ArticleType'} = 'webrequest';</pre>

2.58.10. Ticket::Frontend::CustomerTicketMessage###SenderType

Description	Value
Description:	Sender type for new tickets from the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'SenderType'} = 'customer';</pre>

2.58.11. Ticket::Frontend::CustomerTicketMessage###HistoryType

Description	Value
Description:	Defines the default history type in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'HistoryType'} = 'WebRequestCustomer';</pre>

2.58.12. Ticket::Frontend::CustomerTicketMessage###HistoryComment

Description	Value
Description:	Comment for new history entries in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<code>\$Self->{'Ticket::Frontend::CustomerTicketMessage'}->{'HistoryComment'} = '';</code>

2.58.13. CustomerPanelSelectionType

Description	Value
Description:	Defines the recipient target of the tickets ("Queue" shows all queues, "SystemAddress" displays all system addresses) in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'CustomerPanelSelectionType'} = 'Queue';</code>

2.58.14. CustomerPanelSelectionString

Description	Value
Description:	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.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'CustomerPanelSelectionString'} = '<Queue>';</code>

2.58.15. CustomerPanelOwnSelection

Description	Value
Description:	Determines which queues will be valid for ticket's recipients in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	0
Required:	0
Config-Setting:	<code>\$Self->{'CustomerPanelOwnSelection'} = { 'Junk' => 'First Queue', 'Misc' => 'Second Queue' };</code>

2.58.16. CustomerPanel::NewTicketQueueSelectionModule

Description	Value
Description:	Module for To-selection in new ticket screen in the customer interface.

Description	Value
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'CustomerPanel::NewTicketQueueSelectionModule'} = 'Kernel::Output::HTML::CustomerNewTicketQueueSelectionGeneric';</pre>

2.58.17. Ticket::Frontend::CustomerTicketMessage###DynamicField

Description	Value
Description:	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.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewNew
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerTicketMessage'}-> {'DynamicField'} = {};</pre>

2.59. Frontend::Customer::Ticket::ViewPrint

2.59.1. Ticket::Frontend::CustomerTicketPrint###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket print screen of the customer interface. Possible settings: 0 = Disabled, 1 = Enabled.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewPrint
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerTicketPrint'}->{'DynamicField'} = {};</pre>

2.60. Frontend::Customer::Ticket::ViewSearch

2.60.1. Ticket::CustomerTicketSearch::SearchLimit

Description	Value
Description:	Maximum number of tickets to be displayed in the result of a search in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<code>\$Self->{'Ticket::CustomerTicketSearch::SearchLimit'} = '5000';</code>

2.60.2. Ticket::CustomerTicketSearch::SearchPageShown

Description	Value
Description:	Number of tickets to be displayed in each page of a search result in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::CustomerTicketSearch::SearchPageShown'} = '40';</code>

2.60.3. Ticket::CustomerTicketSearch::SortBy::Default

Description	Value
Description:	Defines the default ticket attribute for ticket sorting in a ticket search of the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::CustomerTicketSearch::SortBy::Default'} = 'Age';</code>

2.60.4. Ticket::CustomerTicketSearch::Order::Default

Description	Value
Description:	Defines the default ticket order of a search result in the customer interface. Up: oldest on top. Down: latest on top.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::CustomerTicketSearch::Order::Default'} = 'Down';</code>

2.60.5. Ticket::Frontend::CustomerTicketSearch###ExtendedSearchCondition

Description	Value
Description:	Allows extended search conditions in ticket search of the customer interface. With this feature you can search w. g. with this kind of conditions like "(key1&&key2)" or "(key1 key2)".
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<code>\$Self->{'Ticket::Frontend::CustomerTicketSearch'}->{'ExtendedSearchCondition'} = '1';</code>

2.60.6. Ticket::Frontend::CustomerTicketSearch###SearchCSVData

Description	Value
Description:	Data used to export the search result in CSV format.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewSearch
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketSearch'}->{'SearchCSVData'} = ['TicketNumber', 'Age', 'Created', 'Closed', 'State', 'Priority', 'Lock', 'CustomerID', 'CustomerName', 'From', 'Subject'];</code>

2.60.7. Ticket::Frontend::CustomerTicketSearch###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket search screen of the customer interface. Possible settings: 0 = Disabled, 1 = Enabled.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewSearch
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketSearch'}->{'DynamicField'} = {};</code>

2.60.8.

Ticket::Frontend::CustomerTicketSearch###SearchOverviewDynamicField

Description	Value
Description:	Dynamic fields shown in the ticket search overview results screen of the customer interface. Possible settings: 0 = Disabled, 1 = Enabled.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewSearch
Valid:	1
Required:	0
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketSearch'}->{'SearchOverviewDynamicField'} = {};</code>

2.60.9. Ticket::Frontend::CustomerTicketSearch###SearchCSVDynamicField

Description	Value
Description:	Dynamic Fields used to export the search result in CSV format.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewSearch
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerTicketSearch'}->{'SearchCSVDynamicField'} = {};</pre>

2.61. Frontend::Customer::Ticket::ViewZoom

2.61.1. Ticket::Frontend::CustomerTicketZoom###NextScreenAfterFollowUp

Description	Value
Description:	Determines the next screen after the follow up screen of a zoomed ticket in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'NextScreenAfterFollowUp'} = 'CustomerTicketOverview';</pre>

2.61.2. Ticket::Frontend::CustomerTicketZoom###ArticleType

Description	Value
Description:	Defines the default type of the note in the ticket zoom screen of the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'ArticleType'} = 'webrequest';</pre>

2.61.3. Ticket::Frontend::CustomerTicketZoom###SenderType

Description	Value
Description:	Defines the default sender type for tickets in the ticket zoom screen of the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	

Description	Value
	<code>\$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'SenderType'} = 'customer';</code>

2.61.4. Ticket::Frontend::CustomerTicketZoom###HistoryType

Description	Value
Description:	Defines the history type for the ticket zoom action, which gets used for ticket history in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'HistoryType'} = 'FollowUp';</code>

2.61.5. Ticket::Frontend::CustomerTicketZoom###HistoryComment

Description	Value
Description:	Defines the history comment for the ticket zoom action, which gets used for ticket history in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'HistoryComment'} = '';</code>

2.61.6. Ticket::Frontend::CustomerTicketZoom###Priority

Description	Value
Description:	Allows customers to change the ticket priority in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<code>\$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'Priority'} = '1';</code>

2.61.7. Ticket::Frontend::CustomerTicketZoom###PriorityDefault

Description	Value
Description:	Defines the default priority of follow up customer tickets in the ticket zoom screen in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewZoom
Valid:	1

Description	Value
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'PriorityDefault'} = '3 normal';</pre>

2.61.8. Ticket::Frontend::CustomerTicketZoom###State

Description	Value
Description:	Allows choosing the next compose state for customer tickets in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'State'} = '1';</pre>

2.61.9. Ticket::Frontend::CustomerTicketZoom###StateDefault

Description	Value
Description:	Defines the default next state for a ticket after customer follow up in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'StateDefault'} = 'open';</pre>

2.61.10. Ticket::Frontend::CustomerTicketZoom###StateType

Description	Value
Description:	Defines the next possible states for customer tickets in the customer interface.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'StateType'} = ['open', 'closed'];</pre>

2.61.11. Ticket::Frontend::CustomerTicketZoom###AttributesView

Description	Value
Description:	Shows the activated ticket attributes in the customer interface (0 = Disabled and 1 = Enabled).
Group:	Ticket

Description	Value
SubGroup:	Frontend::Customer::Ticket::ViewZoom
Valid:	1
Required:	1
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'AttributesView'} = { 'Owner' => '0', 'Priority' => '1', 'Queue' => '1', 'Responsible' => '0', 'SLA' => '0', 'Service' => '0', 'State' => '1', 'Type' => '0' };</pre>

2.61.12. Ticket::Frontend::CustomerTicketZoom###DynamicField

Description	Value
Description:	Dynamic fields shown in the ticket zoom screen of the customer interface. Possible settings: 0 = Disabled, 1 = Enabled.
Group:	Ticket
SubGroup:	Frontend::Customer::Ticket::ViewZoom
Valid:	1
Required:	0
Config-Setting:	<pre>\$Self->{'Ticket::Frontend::CustomerTicketZoom'}->{'DynamicField'} = {};</pre>

2.62. Frontend::Queue::Preferences

2.62.1. QueuePreferences###Comment2

Description	Value
Description:	Parameters of the example queue attribute Comment2.
Group:	Ticket
SubGroup:	Frontend::Queue::Preferences
Valid:	0
Required:	0
Config-Setting:	<pre>\$Self->{'QueuePreferences'}->{'Comment2'} = { 'Block' => 'TextArea', 'Cols' => '50', 'Desc' => 'Define the queue comment 2.', 'Label' => 'Comment2', 'Module' => 'Kernel::Output::HTML::QueuePreferencesGeneric', 'PrefKey' => 'Comment2', 'Rows' => '5' };</pre>

2.63. Frontend::SLA::Preferences

2.63.1. SLAPreferences###Comment2

Description	Value
Description:	Parameters of the example SLA attribute Comment2.

Description	Value
Group:	Ticket
SubGroup:	Frontend::SLA::Preferences
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'SLAPreferences'}->{'Comment2'} = { 'Block' => 'TextArea', 'Cols' => '50', 'Desc' => 'Define the sla comment 2.', 'Label' => 'Comment2', 'Module' => 'Kernel::Output::HTML::SLAPreferencesGeneric', 'PrefKey' => 'Comment2', 'Rows' => '5' }; </pre>

2.64. Frontend::Service::Preferences

2.64.1. ServicePreferences###Comment2

Description	Value
Description:	Parameters of the example service attribute Comment2.
Group:	Ticket
SubGroup:	Frontend::Service::Preferences
Valid:	0
Required:	0
Config-Setting:	<pre> \$Self->{'ServicePreferences'}->{'Comment2'} = { 'Block' => 'TextArea', 'Cols' => '50', 'Desc' => 'Define the service comment 2.', 'Label' => 'Comment2', 'Module' => 'Kernel::Output::HTML::ServicePreferencesGeneric', 'PrefKey' => 'Comment2', 'Rows' => '5' }; </pre>

Приложение С. Благодарности


OTRS является проектом с открытым кодом и, мы хотим поблагодарить многих людей за их помощь и поддержку. Конечно же следующий список далеко неполный и мы приносим свои извинения за это! Просто напишите нам, если не наши себя в этом списке.

Следующие люди внесли особый вклад в проект и являются его активными сторонниками:

- Robert Kehl, создавший Win32-инсталлятор для релиза 1.x. Больше тебе спасибо, Robert!
- Torsten Werner, который поддерживает Debian-installer для OTRS. Спасибо большое, Torsten, теперь OTRS также доступен для Debian-сообщества.
- Nils Jeppe (зеркало Гамбург, Германия), Bryan Fullerton (зеркало Торонто, Канада), Eberhard Mönkeberg (зеркало Геттинген, Германия), Timo Dreger (зеркало Дюссельдорф, Германия) и Netmonic (зеркало Вена, Австрия), которые поддерживают зеркальное копии нашего ftp-сервера. Большое спасибо, благодаря вашей помощи всегда есть возможность быстро скачать OTRS!
- Anja Schneider, которая помогает с переводом этого руководства и коррекции и пересмотр немецких текстов. Большое спасибо за вашу помощь и терпение, Anja!
- Через списки рассылки OTRS мы получаем множество идей, советов по усовершенствованию системы, обновления и исправления. Также здесь доступна большая поддержка для всех пользователей. Большое спасибо всем людям, которые участвуют в наших списках рассылки за вашу активную помощь!

Также мы хотели бы поблагодарить следующих людей:

- Martin Scherbaum
- Carsten Gross
- Harald Müller
- Stefan Schmidt
- Milisav Radmanic
- Uli Hecht
- Norman Walsh
- Heiko Baumann
- Atif Ghaffar
- Pablo Ruiz Garcia
- Dan Rau
- Christoph Kaulich
- Mark Jackson
- Diane Shieh
- Bernard Choppy

-
- Carl Bailey
 - Phil Davis
 - Edwin D. Vinas
 - Lars Müller
 - Vladimir Gerdjikov
 - Fred van Dijk
 - Sebastien Guilbaud
 - Wiktor Wodecki
 - Arnold Ligtoet
 - Antti Kämäräinen
 - Nicolas Goralski
 - Gilberto Cezar de Almeida
 - Jorge Becerra
 - Eddie Urenda
 - Stella Power
 - Andreas Haase
 - Reiner Keller
 - Covert Jake
 - Moshe Leibovitch
 - Björn Jacke
 - Remo Catelotti
 - Alfons Obermeyer
 - Michael Rech
 - Danie Theron
 - Richard Paradies
 - Art Powell
- 

Приложение D. GNU Free Documentation License

Version 1.1, March 2000

Copyright (C) 2000 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other written document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or non-commercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you".

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document, that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters), and contains nothing that could fall directly within that overall subject. (For example, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, whose contents can be viewed and edited directly and straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available

drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats, suitable for input to text formatters. A copy made in an otherwise Transparent file format, whose markup has been designed to thwart or discourage subsequent modification by readers is not Transparent. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include: plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML designed for human modification. Opaque formats include: PostScript, PDF, proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or non-commercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies of the Document numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a publicly-accessible computer-network location, containing a complete Transparent copy of the Document, free of added material, which the general network-using public has access to download anonymously, at no charge, using public-standard network protocols. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location, until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
- B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has less than five).
- C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
- D. Preserve all the copyright notices of the Document.
- E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
- F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
- G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
- H. Include an unaltered copy of this License.
- I. Preserve the section entitled "History", and its title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version, as stated in the previous sentence.
- J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
- K. In any section entitled "Acknowledgements" or "Dedications", preserve the section's title, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
- L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
- M. Delete any section entitled "Endorsements". Such a section may not be included in the Modified Version.

N. Do not retitle any existing section as "Endorsements" or to conflict in title with any Invariant Section.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties--for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections entitled "History" in the various original documents, forming one section entitled "History"; likewise combine any sections entitled "Acknowledgements", and any sections entitled "Dedications". You must delete all sections entitled "Endorsements."

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License, for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, does not as a whole count as a Modified Version of the Document, provided no compilation copyright is

claimed for the compilation. Such a compilation is called an "aggregate", and this License does not apply to the other self-contained works thus compiled with the Document, on account of their being thus compiled, if they are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one quarter of the entire aggregate, the Document's Cover Texts may be placed on covers that surround only the Document within the aggregate. Otherwise they must appear on covers around the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections, in addition to the original versions of these Invariant Sections. You may include a translation of this License provided that you also include the original English version of this License. In case of a disagreement between the translation and the original English version of this License, the original English version will prevail.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See <http://www.gnu.org/copyleft/>.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

. How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

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".

If you have no Invariant Sections, write "with no Invariant Sections" instead of saying which ones are invariant. If you have no Front-Cover Texts, write "no Front-Cover Texts" instead of "Front-Cover Texts being LIST"; likewise for Back-Cover Texts.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.