

ConSol Software GmbH

Release Notes ConSol CM 6.11.1

Author: Technical documentation team

Phone: Phone: +49 (0)89-458-41-100

Mail: tecdoc-l@consol.de

Version: 19.0

Date: 10.03.2022

Status: Released

Contents

Contents	2
General update instructions	8
Required scene changes	8
1. Version 6.11.1.0 (24.10.2017)	9
1.1 Update and installation instructions	10
1.1.1 System upgrade from 6.11.0 and earlier versions	10
1.1.2 WebLogic application server support restored (#630596)	10
1.1.3 Script modification requirements	10
1.1.4 Configuration modification requirements	11
1.2 New features	12
1.2.1 Relation visualization graphs (#630556)	12
1.2.2 Dashboard widget showing recent changes (#630466)	20
1.2.3 Dashboard widget listing the last viewed elements (#630465)	23
1.2.4 Webhook integration interface (#631699)	25
1.2.5 Forms for actions beyond workflow (#629429, #629430)	30
1.2.6 Unified interface for action-related operations (#629296)	36
1.2.7 Parallel editing handling for customers and resources (#630442)	39
1.2.8 Colored ticket symbols in favorites and workspace (#626079, #630889)	40
1.2.9 Specific page titles (#625806)	40
1.2.10 Workflow activity execution not listed in history (#631304)	41
1.2.11 Custom annotation usage in Web Client custom projects (#624043)	42
1.2.12 Widget style customization with CSS (#630386)	42
1.2.13 New Admin Tool template function for abbreviating strings (#607391)	42
1.2.14 Admin Tool script templates for actions unified (#630726)	43
1.2.15 Web server adapter view object cache (#631275)	43
1.2.16 Data warehouse monitor in Admin Tool (#630811)	43
1.2.17 Configuration import transaction encapsulation (#630480)	45
1.2.18 CM/Track user switch history entry (#627732)	45
1.2.19 CM/Track user definition handling (#625328)	45
1.2.20 Database performance logging (#631213)	48
1.3 Changes	50
1.3.1 Layout improvements	50
1.3.2 Performance improvements	54
1.3.3 CM/Track V2 improvements	54
1.3.4 WebLogic application server support restored (#630596)	54
1.3.5 New splashscreen design and application icons for Admin Tool and Process Designer (#630919)	55
1.3.6 User login session handling enhancements (#623625, #630644)	55
1.3.7 Undoing delete/remove operations replaced by confirmations (#627311)	55
1.3.8 Concurrent editing handling optimization for customers and resources (#631572, #631739)	56
1.3.9 Relation targets filtered for existing relations (#631302)	56
1.3.10 Usage of disabled ENUM values in grid view (#630995)	57

1.3.11	History optimization for localized labels (#631218)	58
1.3.12	Representation notification enhancement (#624305)	58
1.3.13	Copy/paste of fields in Admin Tool ignores annotation contains contact (#630820)	59
1.3.14	Clarification of information messages when deleting customers (#627639)	59
1.3.15	REST API returning FAQ tickets streamlined (#630997)	60
1.3.16	Resource relation output ETL plugin enhancement (#630633)	60
1.3.17	Customer data model switch restriction added (#632133)	61
1.3.18	Process Designer script validation disabled when offline (#631149)	61
1.3.19	Activity availability in workflow scopes with triggers and scope activities (#631072)	61
1.3.20	Class imports for scripts unified (#630718)	61
1.3.21	Identification of automatic activities for data warehouse reporting extended (#625781)	62
1.3.22	ENUM changing operations written directly to the database (#630330)	62
1.3.23	Trimming long file names of email attachments (#625239)	63
1.3.24	WorkflowEngineStatus MBean output extended (#631175)	63
1.3.25	JBoss cache item life span configurable (#630565)	64
1.3.26	Explicit deployment timeout setting for JBoss EAP (#630754)	64
1.3.27	JBoss setting for HTTP parameter limit provided (#631199)	64
1.3.28	Node clock offset tolerance increased for index synchronization in a cluster (#630525)	65
1.3.29	Dedicated category labels for resource operations in operationtimes.log (#630787)	65
1.3.30	Log file for the deletion of customers reintroduced (#631158)	65
1.3.31	REST API logging default setting and content changed (#621191)	65
1.3.32	Scene export error logging improvement (#623575)	66
1.3.33	Irrelevant log messages from server start filtered out (#629645)	66
1.3.34	Unused system properties removed (#631640)	66
1.3.35	Mail classes added to the API documentation package (#631310)	67
1.4	Bugs fixed	68
2.	Version 6.11.1.1 (31.01.2018)	73
2.1	Update and installation instructions	74
2.1.1	System upgrade from 6.11.0 and earlier versions	74
2.1.2	REST API client modification requirements	74
2.1.3	Script modification requirements	74
2.1.4	Configuration modification requirements	74
2.2	New features	75
2.2.1	New handling of scripted autocomplete fields (#632555)	75
2.3	Changes	80
2.3.1	Layout improvements	80
2.3.2	CM/Track V2 improvements	80
2.3.3	Localized title for Recent changes and Recently visited widgets (#632405)	81
2.3.4	Synchronized XML configuration files (#632410)	81
2.3.5	Webhooks integration for WebLogic (#632412)	81
2.3.6	Layout of the relation visualization graph improved (#632454)	81

2.3.7	Recent changes widgets improved (#632466, #632471, #632472, #632549, #632550)	82
2.3.8	Unneeded error log for non-existing unit (#632469)	82
2.3.9	New caching mechanism enabled (#632523)	83
2.3.10	Relation graph subscope renamed in page customization (#632569)	83
2.3.11	Support for list fields in control forms (#632667)	83
2.3.12	Support for HTTP status / error codes for the Webhook (#632695)	83
2.3.13	Restore triggers after execution of scope activity (#632698)	85
2.3.14	Action type available in relation action scripts (#632812)	85
2.3.15	Data version updated (#632834)	86
2.3.16	Default value of the property unit.description.mode updated (#632882)	86
2.3.17	Class SecurityHelper deprecated (#632911, #632937)	86
2.4	Bugs fixed	87
3.	Version 6.11.1.2 (14.02.2018)	88
3.1	Update and installation instructions	89
3.1.1	System upgrade from 6.11.0 and earlier versions	89
3.2	Bugs fixed	90
4.	Version 6.11.1.3 (05.03.2018)	91
4.1	Update and installation instructions	92
4.1.1	System upgrade from 6.11.0 and earlier versions	92
4.2	Changes	93
4.2.1	Performance improvements	93
4.2.2	Layout improvements	93
4.3	Bugs fixed	94
5.	Version 6.11.1.4 (30.03.2018)	95
5.1	Update and installation instructions	96
5.1.1	System upgrade from 6.11.0 and earlier versions	96
5.1.2	Script modification requirements	96
5.1.3	Configuration modification requirements	96
5.2	New features	97
5.2.1	Printing emails and comments (#633142)	97
5.3	Changes	99
5.3.1	Layout improvements	99
5.3.2	CM/Track V2 improvements	99
5.3.3	API for scripted autocomplete fields improved (#632870)	100
5.3.4	Session upgrade and downgrade (#632899)	101
5.3.5	Calendar section expanded for adding appointment (#632906)	102
5.3.6	Option to hide empty groups in the ticket list (#633106)	102
5.3.7	Method client.goToCreateUnit(unit) (#633156)	103
5.4	Bugs fixed	104
6.	Version 6.11.1.5 (26.04.2018)	105
6.1	Update and installation instructions	106
6.1.1	System upgrade from 6.11.0 and earlier versions	106

6.2 Changes	107
6.2.1 Usability improvements	107
6.2.2 CM/Track V2 improvements	107
6.2.3 Method to open action forms for customers and resources (#633102)	107
6.2.4 Duplicate entries removed from domain.xml (#633285)	108
6.2.5 VOCaching disabled in cluster mode (#631155)	108
6.3 Bugs fixed	109
7. Version 6.11.1.6 (29.05.2018)	110
7.1 Update and installation instructions	111
7.1.1 System upgrade from 6.11.0 and earlier versions	111
7.1.2 Data warehouse update	111
7.1.3 Script modification requirements	111
7.1.4 Configuration modification requirements	111
7.2 New features	112
7.2.1 Contact deletion for GDPR (#633392)	112
7.3 Changes	116
7.3.1 Usability improvements	116
7.3.2 Rendering of empty scripted visualization fields introduced (#632892)	117
7.3.3 Usage of wildcards in the Quick Search improved (#633086)	118
7.3.4 API method RequestLocale.get() improved to retrieve locale in CM/Track V2 (#633249)	118
7.3.5 Support for secure SMTP introduced (#633298)	118
7.3.6 API and logs for clusters improved (#633357)	118
7.4 Bugs fixed	120
8. Version 6.11.1.7 (11.07.2018)	121
8.1 Update and installation instructions	122
8.1.1 System upgrade from 6.11.0 and earlier versions	122
8.1.2 Data warehouse update	122
8.1.3 Script modification requirements	122
8.1.4 Configuration modification requirements	122
8.2 New features	123
8.2.1 Contact anonymization for GDPR (#633502)	123
8.3 Changes	125
8.3.1 Usability improvements	125
8.3.2 New API methods related to ticket closing date (#627989)	125
8.3.3 Whitelist for links in incoming emails (#632591)	125
8.3.4 Logging of optimistic locking exceptions improved (#633415)	126
8.3.5 Content of pContext variable after queue change (#633441)	126
8.3.6 Automatic ticket update for Webhooks (#633444)	126
8.3.7 Content of pContext variable in actions changed (#633475)	126
8.3.8 Forwarding of manual emails configurable (#633499)	126
8.4 Bugs fixed	127
9. Version 6.11.1.8 (06.09.2018)	129
9.1 Update and installation instructions	130
9.1.1 System upgrade from 6.11.0 and earlier versions	130

9.2 Changes	131
9.2.1 Usability improvements	131
9.2.2 Rendering for empty scripted visualization fields introduced for CM/Track V2 (#633443)	131
9.3 Bugs fixed	132
10. Version 6.11.1.9 (12.10.2018)	133
10.1 Update and installation instructions	134
10.1.1 System upgrade from 6.11.0 and earlier versions	134
10.1.2 Configuration modification requirements	134
10.2 Changes	135
10.2.1 Usability improvements	135
10.2.2 Performance improvements	135
10.2.3 Configurable date format in CM/Track V2 (#633393)	135
10.2.4 Shortcut to jump to the Quick Search (#633819)	135
10.3 Bugs fixed	136
11. Version 6.11.1.10 (17.10.2018)	138
11.1 Update and installation instructions	139
11.1.1 System upgrade from 6.11.0 and earlier versions	139
11.2 Bugs fixed	140
12. Version 6.11.1.11 (26.10.2018)	141
12.1 Update and installation instructions	142
12.1.1 System upgrade from 6.11.0 and earlier versions	142
12.2 Bugs fixed	143
13. Version 6.11.1.12 (21.12.2018)	144
13.1 Update and installation instructions	145
13.1.1 System upgrade from 6.11.0 and earlier versions	145
13.1.2 REST API client modification requirements	145
13.1.3 Script modification requirements	145
13.1.4 Configuration modification requirements	145
13.2 New features	146
13.2.1 Development mode for dashboard widgets (#633970)	146
13.3 Changes	147
13.3.1 Unit ID available in autocomplete scripts in customer forms (#633820)	147
13.3.2 Precedence of visibility annotations changed (#633871)	147
13.3.3 Home screen of the Admin Tool adapted (#634166)	147
13.3.4 Localized description of ACFs added to the REST API and CM/Track (#634170)	147
13.3.5 Support for sending emails to many recipients (#634174)	147
13.4 Bugs fixed	148
14. Version 6.11.1.13 (18.01.2019)	149
14.1 Update and installation instructions	150
14.1.1 System upgrade from 6.11.0 and earlier versions	150

14.2 Bugs fixed	151
15. Version 6.11.1.14 (11.03.2019)	152
15.1 Update and installation instructions	153
15.1.1 System upgrade from 6.11.0 and earlier versions	153
15.1.2 Configuration modification requirements	153
15.2 Changes	154
15.2.1 Usability improvements	154
15.2.2 Remember filter settings on customer pages (#633523)	154
15.2.3 Error handling regarding password change in CM/Track V2 improved (#634305)	154
15.2.4 Script cache invalidation enhanced (#634409)	155
15.2.5 Logging for authentication exceptions added for the REST API (#634701)	155
15.3 Bugs fixed	156
16. Version 6.11.1.15 (19.07.2019)	157
16.1 Update and installation instructions	158
16.1.1 System upgrade from 6.11.0 and earlier versions	158
16.2 Changes	159
16.2.1 Usability improvements	159
16.2.2 Wildcard behavior changed for customer and engineer autocomplete search (#634405)	159
16.2.3 Handling of required fields in CM/Track V2 improved (#634751)	159
16.3 Bugs fixed	160
17. Version 6.11.1.16 (27.09.2019)	161
17.1 Update and installation instructions	162
17.1.1 System upgrade from 6.11.0 and earlier versions	162
17.1.2 Data Warehouse update	162
17.2 Changes	163
17.2.1 DWH adapted to support dates until 01.01.2100 (#641871)	163
18. Version 6.11.1.17 (23.11.2021)	164
18.1 Update and installation instructions	165
18.1.1 System upgrade from 6.11.0 and earlier versions	165
19. Version 6.11.1.18 (10.03.2022)	166
19.1 Update and installation instructions	167
19.1.1 System upgrade from 6.11.0 and earlier versions	167
20. Web applications expiry	168

General update instructions

There are two ways for updating **ConSol CM** from an old version:

- **Distribution installation**

The distribution is installed on the application server. You need to back up the configuration files, as for example `cm6.xml`, before performing the update. Then you need to apply your settings to the new configuration files.

This type of update ensures that every change between the versions is installed. It is recommended for updates of the major or minor version, e.g. for an update from 6.10.8 to 6.11.0.

- **EAR / WAR update**

The EAR (`cm6.ear`, `cmrf.ear`) and WAR (`cm-track.war`) files of the new version are installed on the application server. The configuration files, as for example `cm6.xml`, are not changed during the update process. Therefore, you have to manually apply any changes made to the configuration files between the old and the new versions.

This type of update is only recommended for updates within a minor version, e.g. from 6.11.1.0 to 6.11.1.5.



Please check the *System Requirements* of the new version to ensure that your operating system, application server and database version are still supported.

Required scene changes

Some changes to your **ConSol CM** scene might be required. Please check the *Update and installation instructions* sections of all **ConSol CM** versions between the old version and the new version. For example, for an update from 6.11.1.0 to 6.11.1.5, the instructions of the versions 6.11.1.1, 6.11.1.2, 6.11.1.3, 6.11.1.4 and 6.11.1.5 have to be checked. In addition, you need to check the instructions of older **ConSol CM** versions which were merged between both versions. The merged versions are stated on the introduction page of each version.

1. Version 6.11.1.0 (24.10.2017)

Version 6.11.1.0 includes all **ConSol CM** changes and additions of

- 6.11.0 versions up to 6.11.0.5
- 6.10 versions up to 6.10.7.0
- 6.9 versions up to 6.9.4.7
- 6.8 versions up to 6.8.5.8

Newer versions and the respective changes are not included in this **ConSol CM** version.

1.1 Update and installation instructions

Please read all information on update and installation of **ConSol CM** specific for version 6.11.1.0 in this section before performing a system update or installation. Please consult the general product documentation on update and installation procedures as well as earlier version Release Notes, if applicable.

1.1.1 System upgrade from 6.11.0 and earlier versions

This **ConSol CM** release needs specific planning for an update from earlier releases.



If you update from a version below 6.11.0, you need to carry out the migration steps described in the *Release Notes* for **ConSol CM** version 6.11.0. Please see [System upgrade from 6.10 and earlier versions \(Release Notes 6.11.0\)](#)



Please check your database indexes for fragmentation. If necessary, reorganize or rebuild them to ensure that the performance of **ConSol CM** is not impaired by a fragmented index after the update. This is especially recommended on Microsoft SQL Server.

You can find more information in the [Microsoft SQL Server documentation](#).

Optional pruning of localized label history

The change described in section [History optimization for localized labels \(#631218\)](#) includes a task script named `translations_cleanup` which will be added during the update. This script can be run as a task and removes orphaned history entries of localized labels. Other unneeded entries are automatically removed during the update. It is recommended to run this script after the update in order to remove unneeded data.

1.1.2 WebLogic application server support restored (#630596)

The support for the **WebLogic** application server product suspended since the release of **ConSol CM** version 6.11.0 is restored with this release.

1.1.3 Script modification requirements

This release features some interface changes in API methods. **Please be aware that scripts using the respective methods may need adjustments!** The affected methods are listed in the following sections of this document:

- [New **Admin Tool** template function for abbreviating strings \(#607391\)](#)
- [ENUM changing operations written directly to the database \(#630330\)](#)
- [Class imports for scripts unified \(#630718\)](#)



Please be aware that a few classes were removed completely and from multiple other classes methods were removed. Please ensure before an update that the scripts in your installation are not affected by these removals!

1.1.4 Configuration modification requirements

The changes in this release require several configuration items to be reviewed and possibly adjusted. Please see the following sections for details:

- [**JBoss** cache item life span configurable \(#630565\)](#)
- [Explicit deployment timeout setting for **JBoss EAP** \(#630754\)](#)
- [REST API logging default setting and content changed \(#621191\)](#)
- [Configuration import transaction encapsulation \(#630480\)](#)
- [Database performance logging \(#631213\)](#)
- [Representation notification enhancement \(#624305\)](#)
- [Unused system properties removed \(#631640\)](#)

No further instructions available.

1.2 New features

The features newly introduced in this **ConSol CM** version 6.11.1.0 are described in detail in this section.

1.2.1 Relation visualization graphs (#630556)

This release of **ConSol CM** adds a whole new visual capability to the system, relation graphs visualizing the explicit association between the objects in the system. This means that there is now functionality to visually present the relations between tickets, customers, and resources in a graph diagram for each single object.

The relation visualization graph can be displayed in the existing relations sections of the **Web Client** or in the newly added *Custom Relation Graph* section. No further configuration or customization is needed to use the relation graphs with their default settings. The configuration of the *Custom Relation Graph* section is described further below.

In the regular relation sections, the user can switch between the list view and the graph view using the *List* and *Graph* links next to the section heading. The *Graph* link is only displayed if the list includes at least one relation. In the graph view, the objects are represented by their icons. Two icons representing linked objects are connected with an arrow. When the graph view is selected, there is no counter stating the number of relations in the section header.

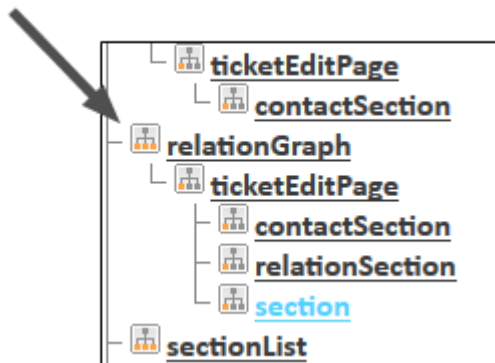
The lower right corner of the graph includes buttons to zoom in and out and restore the default zoom. A tooltip is displayed when the user hovers over an icon or clicks an icon. Double-clicking an icon opens the page of the respective object.

The level of detail displayed in the graph depends on the number of objects and the zoom factor. The graph always shows the icons of the objects and the arrows between related objects. If there is enough space on the canvas, it also includes the description of each object and the type of relation between two objects.

Custom configuration of the relationGraph (standard sections)

The page customization in the **Web Client** has the new type *relationGraph*. It can be used to configure the relation visualization graphs in the standard relation sections of the **Web Client**. The type *relationGraph* has three attributes:

- *graph*
Determines whether a graph should be shown. Possible values are:
 - “Default” - The graph is activated and shown by default.
 - “On” - Default value. The graph is activated and the user can click the *Graph* link to display it.
 - “Off” - The graph is deactivated.
- *graphConfiguration*
Allows to provide a custom configuration for the graph.
- *graphScriptName*
Allows to enter the name of the script holding the custom configuration for the graph. The script has the type *Relation graph* in the **Admin Tool**.



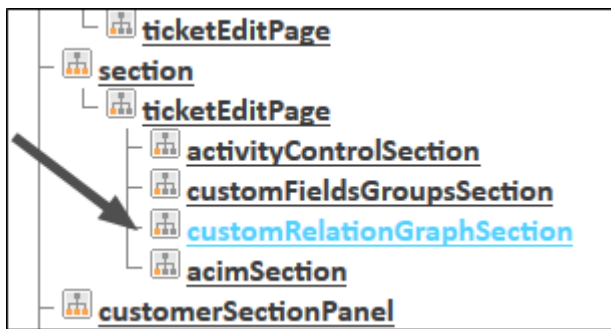
relationGraph	/ticketEditPage/section	ExtendedRelationGraphCustomization	configuration script: <input type="text"/> run with admin privileges: <input checked="" type="checkbox"/>
type	scope	className	
Customization of relation graph, context={ticketId,unitId,resourceId}			
Attribute name	Description	Value	
graph	Possible values: <ul style="list-style-type: none"> • Default - the graph visualization is active and replace the standard view by default • On - (default value) the graph visualization is active, user is able to see the data in the graph • Off - No graph is visible 	<input type="text" value="On"/> (java.lang.String) (+) (-)	
graphConfiguration	Graph configuration which is passed to the service directly example: { 'ticketTickets': { 'maximumStepsFromRoot': 2, 'relations': ['reference', 'parent', 'child', 'master', 'slave'], 'includedQueues': ['queue1', 'queue2'], 'excludedQueues': ['queue1', 'queue2'] } }	<input type="text"/> (java.lang.String) (+) (-)	
graphScriptName	Relation graph script used to obtain the graph details from server side	<input type="text"/> (java.lang.String) (+) (-)	

This configuration is done in the subscope of the section where the graph should be shown. If it is done in the type or scope, it is applied to the first subscope automatically.

Custom configuration of the customRelationGraph (new section)

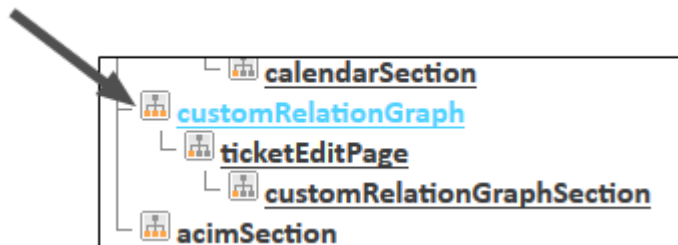
A new section *customRelationGraphSection* was added to the page customization in the type *section* on the ticket edit page, company edit page, contact edit page, and resource edit page. It can be used to add a customized relation visualization graph. When using this custom section, the complete

configuration of the graph needs to be provided in a script.



The section is hidden by default. It can be enabled by setting the attribute *state* to “expanded” or “collapsed”. The default header for the section is *Custom Relation Graph*.

Once the section has been made visible, the type *customRelationGraph* is available in the page customization.



It has two attributes which can be used to configure the section:

- *graphScriptName*
Allows to enter the name of the script holding the custom configuration for the graph. The script has the type *Relation graph* in the **Admin Tool**. It is mandatory to provide a script, otherwise an error is displayed when opening the section in the **Web Client**.
- *sectionHeader*
Allows to edit the label displayed in the section header.

Attribute name	Description	Value
graphScriptName	Relation graph script used to obtain the graph details from server side	<input type="text"/> (java.lang.String) (+) (-)
sectionHeader	Section header	Custom relation graph (java.lang.String) (+) (-)

Create Reset

Configuration options for relation visualization graphs

The configuration of a standard relation visualization graph (existing relations section) is optional. It is done in the page customization of the respective section by entering a JSON string in the *graphConfiguration* attribute or entering the name of the script which configures the graph in the *graphScriptName* attribute.

The custom relation visualization graph can only be configured using a script. The custom configuration is mandatory in this case.

The available configuration options are the same for both types of relation visualization graphs. There are several aspects which can be customized:

- Scope of the graph: relations (object types, relation definitions), depth, inclusions/exclusions
- Layout of the graph elements: colors, fonts, appearance of edges and nodes, labels, etc. See the **vis.js** documentation (<http://visjs.org/docs/network/>) for a complete list of options.

The relation visualization graph can show all kinds of relations of the object type of the current page, independently from the purpose of the section where it is located. The available relations are:

- For ticket pages:
 - **ticketTickets**: related tickets of a ticket
 - **ticketCustomers**: main and additional customers of a ticket
 - **ticketResources**: related resources of a ticket
- For customer pages:
 - **unitUnits**: related customers of a customer
 - **companyContacts**: contacts belonging to a company
 - **contactCompany**: company of a contact
 - **customerTickets**: tickets of a customer (as main or additional customer)
 - **unitResources**: related resources of a customer
- For resource pages:
 - **resourceResources**: related resources of a resource
 - **resourceTickets**: related tickets of a resource
 - **resourceUnits**: related customers of a resource

The following example shows a list of all available definitions regarding the scope of the graph.

```
// one can set either included or excluded elements, not both
// default maximumStepsFromRoot == 2
{
  "ticketTickets" : {
    "maximumStepsFromRoot" : 2,
    "relations" : [ "reference", "parent", "child", "master", "slave" ],
    "includedQueues" : [ "queue1", "queue2" ],
    "excludedQueues" : [ "queue1", "queue2" ]
  },
  "unitUnits" : {
    "maximumStepsFromRoot" : 2,
    "relations" : [ "unitRelationDefinition1", "unitRelationDefinition2" ],
    "includedCustomerGroups" : [ "customerGroup1", "customerGroup2" ],
    "excludedCustomerGroups" : [ "customerGroup1", "customerGroup2" ]
  },
  "ticketCustomers" : {
    "maximumStepsFromRoot" : 2,
    "relations" : [ "main", "default", "contactTicketRoleName" ],
    "includedCustomerGroups" : [ "customerGroup1", "customerGroup2" ],
    "excludedCustomerGroups" : [ "customerGroup1", "customerGroup2" ]
  },
  "customerTickets" : {
```



```

    "maximumStepsFromRoot" : 2,
    "relations" : [ "main", "default", "contactTicketRoleName" ],
    "includedQueues" : [ "queue1", "queue2" ],
    "excludedQueues" : [ "queue1", "queue2" ]
  },
  "contactCompany" : {
    "maximumStepsFromRoot" : 2
  },
  "companyContacts" : {
    "maximumStepsFromRoot" : 2
  },
  "resourceResources" : {
    "maximumStepsFromRoot" : 2,
    "relations" : [ "related", "resourceRelationDefinition" ],
    "includedResourceTypes" : [ "resourceType1", "resourceType2" ],
    "excludedResourceTypes" : [ "resourceType1", "resourceType2" ]
  },
  "ticketResources" : {
    "maximumStepsFromRoot" : 2,
    "relations" : [ "related", "resourceRelationDefinition" ],
    "includedResourceTypes" : [ "resourceType1", "resourceType2" ],
    "excludedResourceTypes" : [ "resourceType1", "resourceType2" ]
  },
  "resourceTickets" : {
    "maximumStepsFromRoot" : 2,
    "relations" : [ "related", "resourceRelationDefinition" ],
    "includedQueues" : [ "queue1", "queue2" ],
    "excludedQueues" : [ "queue1", "queue2" ]
  },
  "unitResources" : {
    "maximumStepsFromRoot" : 2,
    "relations" : [ "related", "resourceRelationDefinition" ],
    "includedResourceTypes" : [ "resourceType1", "resourceType2" ],
    "excludedResourceTypes" : [ "resourceType1", "resourceType2" ]
  },
  "resourceUnits" : {
    "maximumStepsFromRoot" : 2,
    "relations" : [ "related", "resourceRelationDefinition" ],
    "includedCustomerGroups" : [ "customerGroup1", "customerGroup2" ],
    "excludedCustomerGroups" : [ "customerGroup1", "customerGroup2" ]
  }
}

```

The following keywords can be used for **relations**:

- **ticketTickets**: "reference" (ticket references the current ticket), "parent" (ticket is parent ticket of the current ticket), "child" (ticket is child ticket of the current ticket), "master" (ticket is parent ticket of the master ticket), "slave" (ticket is slave ticket of the current ticket)
- **ticketCustomers** / **customerTickets**: "main" (customer is the main customer of the ticket), "default" (customer is an additional customer without role)
- **resourceResources** / **ticketResources** / **resourceTickets** / **unitResources** / **resourceUnits**: "related" (anonymous relation)

For directional relations between customers and resources (`unitUnits` and `resourceResources`), it is possible to specify the direction, i.e., whether the objects linked in the graph should be source or target of the relation. The syntax is as follows:

- `"relations" : ["unitRelationDefinition:source"]`

The linked customers are the source of the relation and the root node is the target.

- `"relations" : ["unitRelationDefinition:target"]`

The linked customers are the target of the relation and the root node is the source.

- `"relations" : ["resourceRelationDefinition:source"]`

The linked resources are the source of the relation and the root node is the target.

- `"relations" : ["resourceRelationDefinition:target"]`

The linked resources are the target of the relation and the root node is the source.

The following code example is a script for a custom relation graph which shows all related objects of a resource. No layout customizations are made, so the standard layout is used.

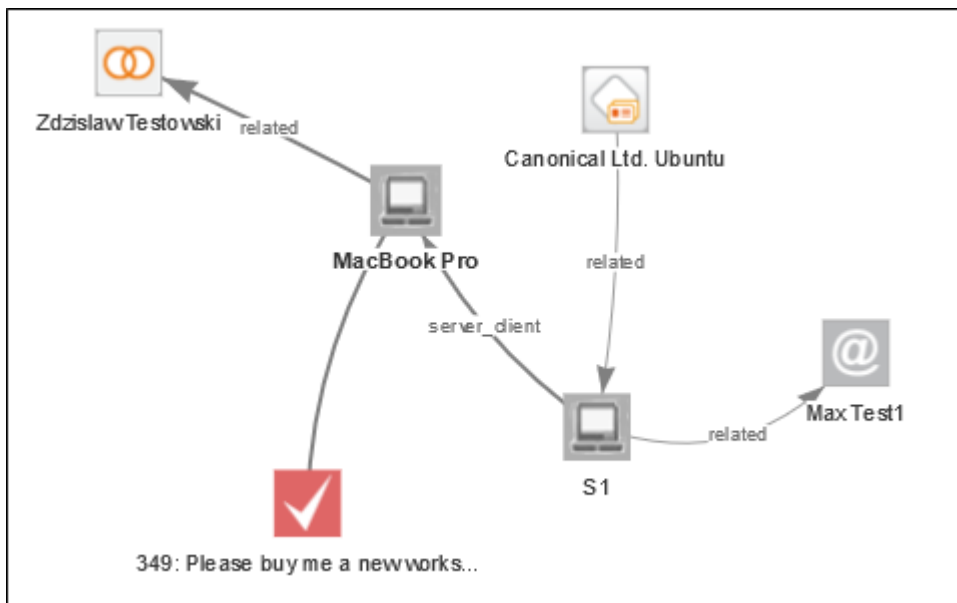
```
import com.consol.cmas.common.model.relation.*

//create criteria using java api

def relationCriteria = new RelationCriteria()
  .withResourceTickets(new RelationCriteria.ResourceTickets()
    .withMaximumStepsFromRoot(2))
  .withResourceUnits(new RelationCriteria.ResourceUnits()
    .withMaximumStepsFromRoot(2))
  .withResourceResources(new RelationCriteria.ResourceResources()
    .withMaximumStepsFromRoot(2));
def resource = resourceService.getById(resourceId)
def relationGraph = relationService.load(resource, relationCriteria);

return relationGraph;
```

The following figure shows the graph as rendered by the above script. The object's icons are used in the standard layout.



The following code example is a script for a custom relation graph which shows all related objects of a resource. The layout is customized using some options from *vis.js*.

```
import com.consol.cmas.common.model.relation.*

//create criteria using java api

def relationCriteria = new RelationCriteria()
    .withResourceTickets(new RelationCriteria.ResourceTickets()
        .withMaximumStepsFromRoot(2))
    .withResourceUnits(new RelationCriteria.ResourceUnits()
        .withMaximumStepsFromRoot(2))
    .withResourceResources(new RelationCriteria.ResourceResources()
        .withMaximumStepsFromRoot(2));

//variables available in script: ticketId, unitId, resourceId (depending on
page)

def resource = resourceService.getById(resourceId)
def relationGraph = relationService.load(resource, relationCriteria);

//modify graph nodes, edges and visjs property

for (RelationNode node : relationGraph.getNodes()) {
    node.withProperty("font", [size:20,color:"blue"]) //change default property
    .withProperty("image", null) //remove default property
    .withProperty("shape", "diamond") //remove default property
}

for (RelationEdge edge : relationGraph.getEdges()) {
    edge.withProperty("arrows", null) //remove default property
    .withProperty("font", [size:16, color:"red"]); //change default property
```

```

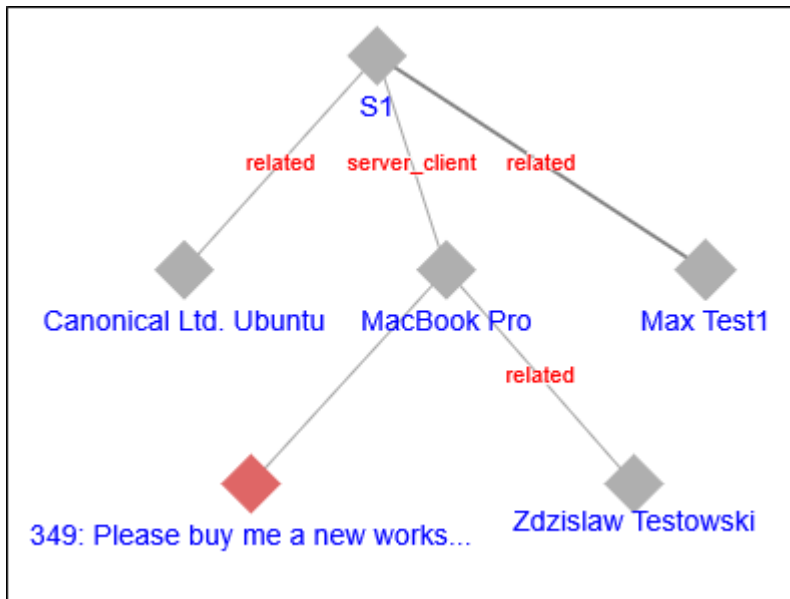
}

relationGraph.withProperty("layout", [hierarchical:[direction:"UD"]]) //change
default property

return relationGraph;

```

The following figure shows the graph as rendered by the above script. Instead of the object's icons, the diamond shape from **vis.js** is used. In addition, the labels have red or blue color.



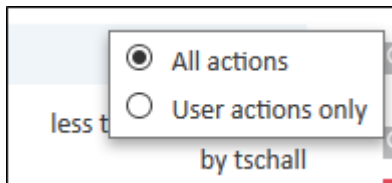
1.2.2 Dashboard widget showing recent changes (#630466)

This **ConSol CM** release features a new standard widget for the dashboard. This new list shows the object changes in the system done most recently.

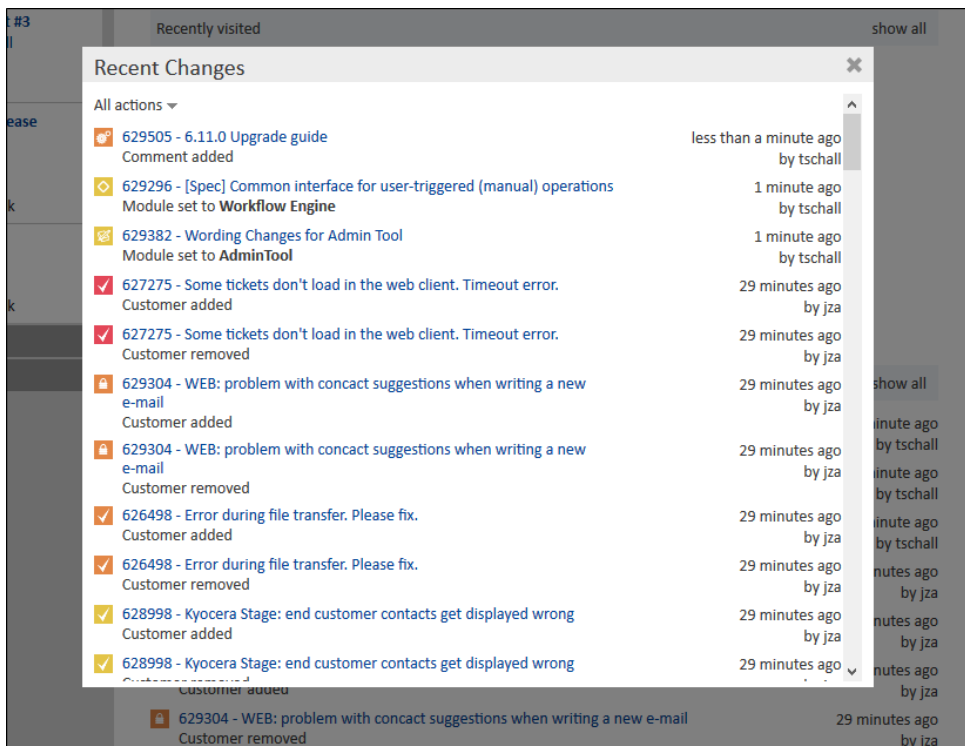
Status	Ticket ID	Description	Time	Engineer
✓	627275	Some tickets don't load in the web client. Timeout error. Customer added	9 minutes ago	jza
✓	627275	Some tickets don't load in the web client. Timeout error. Customer removed	9 minutes ago	jza
✗	629304	WEB: problem with contact suggestions when writing a new e-mail. Customer added	9 minutes ago	jza
✗	629304	WEB: problem with contact suggestions when writing a new e-mail. Customer removed	9 minutes ago	jza
✓	626498	Error during file transfer. Please fix. Customer added	9 minutes ago	jza

This list widget shows the latest changes made to tickets to which the engineer has access. It shows the colored status icon with the standard ticket identification label, the time of the change as well as which engineer made which kind of change. The widget uses speaking category names for the time and change as it can be observed in the screenshot above.

In the widget header bar on the right side there is a small cog wheel menu for applying a filter to the entry list. The menu allows to either show all changes including those done automatically by background workflow activities and incoming emails (*All actions*) or limit the entries to changes actively made by an engineer (*User actions only*).



Next to this menu there is the command link *show all* which opens the same list in an overlay over the grayed out page. The overlay list is longer and shows (up to) the last 100 changes whereas the main widget view by configuration is shorter as is explained below.



The widget does not require a script to provide the data and can be configured by page customizations completely.

The layout attribute on the welcome page (type *widgetsGrid* and scope */welcomePage*) defines its presence and position just like for any other widget. The following value shows it in the first line above another widget for listing recently viewed objects:

```
[recentDataChanges:RecentChanges][recentlyVisitedObjects:RecentlyVisited]
```

The first element names this specific widget instance as **recentDataChanges** and identifies its type as **RecentChanges** for the kind of widget described here.

maxItems	Maximum number of items that are to be displayed.	10 (int) (+) (-)	
title	Recent changes widget title	Recent Changes (java.lang.String) (+) (-)	
visible	Indicates whether the widget is shown	true (boolean) (+) (-)	

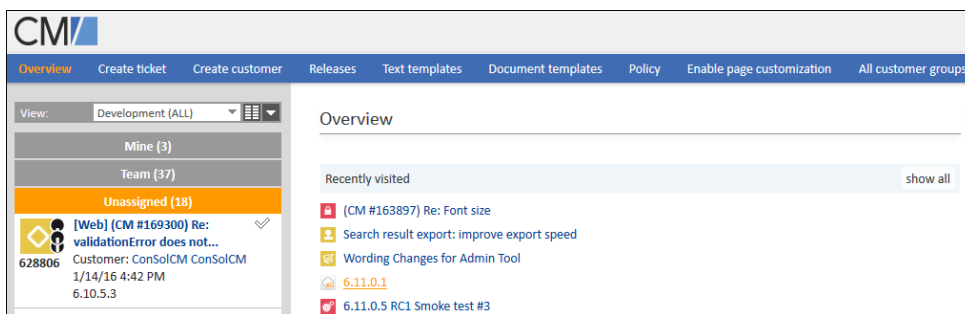
Other configuration for this example is then fully made in the scope `/welcomePage/recentDataChanges` of the type `recentChangesWidget`. This scope lists the following attributes for widget configuration allowing the values described:

- `excludedHistoryTypes`: History entry types which should be excluded from the listing as a change. The value is entered as a comma-separated list of the following entry type identifiers:
 - `"OPEN"`: Ticket has been created.
 - `"REOPEN"`: Ticket has been re-opened.
 - `"UPDATE"`: Ticket has been updated.
 - `"SUBJECT_CHANGE"`: Ticket subject has been updated.
 - `"ENGINEER_CHANGE"`: Engineer associated has been changed.
 - `"TICKET_USER_ADDED"`: An additional engineer has been added.
 - `"TICKET_USER_REMOVED"`: An additional engineer has been removed.
 - `"TICKET_USER_MOVED"`: An additional engineer function has been changed from one function to another.
 - `"QUEUE_CHANGE"`: Ticket's queue has been changed.
 - `"ACTIVITY_CHANGE"`: Ticket status has changed.
 - `"TRANSFERRED_TO_ACTIVITY"`: Ticket has been transferred to another activity.
 - `"CONTENT_ADD"`: New ticket content has been added.
 - `"ATTACHMENT_DELETE"`: Ticket attachment has been deleted.
 - `"ATTACHMENT_UNDO_DELETION"`: Ticket attachment deletion has been made undone.
 - `"EMAIL_ATTACHMENT_DELETE"`: An attachment was deleted from an email.
 - `"EMAIL_ATTACHMENT_UNDO_DELETION"`: An email attachment deletion has been made undone.
 - `"BOOKING"`: A time booking has been added.
 - `"CONTACT_ADD"`: A customer has been added to the ticket.
 - `"CONTACT_REMOVE"`: A customer has been removed from the ticket.
 - `"CONTACTROLE_SWAP"`: The customer role in the ticket has been changed.
 - `"CONTACTROLE_UPDATE"`: The customer role in the ticket has been updated.
 - `"TRANSFERRED_TO_CONTACT"`: The ticket has been transferred to a new contact.
 - `"TICKETRELATION_CREATE"`: A ticket relation has been created.
 - `"TICKETRELATION_REMOVE"`: A ticket relation has been removed.
 - `"TICKETRELATION_UPDATE"`: A ticket relation has been updated.
 - `"TICKETRELATION_MOVE"`: A ticket relation has been changed from one relation type to another.
 - `"RESOURCE_RELATION_CREATED"`: A resource relation has been created.
 - `"RESOURCE_RELATION_UPDATED"`: A resource relation has been updated.
 - `"RESOURCE_RELATION_REMOVED"`: A resource relation has been removed.

- “RESOURCE_ON_THE_FLY_RELATION_CREATED”: An on-the-fly resource relation has been created.
- “RESOURCE_ON_THE_FLY_RELATION_UPDATED”: An on-the-fly resource relation has been updated.
- “RESOURCE_ON_THE_FLY_RELATION_REMOVED”: An on-the-fly resource relation has been removed.
- *excludedQueues*: A comma-separated list of queues identified by their technical names. The named queues and their tickets will be excluded from the listing in the widget.
- *localization*: Currently not used.
- *maxItems*: The maximum number of items shown in the widget main view (integer).
- *title*: The title string shown in the header bar, localization of this label is not available yet.
- *visible*: Widget visibility, must be set to “true” to have the widget on the welcome page, setting it to “false” hides the widget and all its contents.

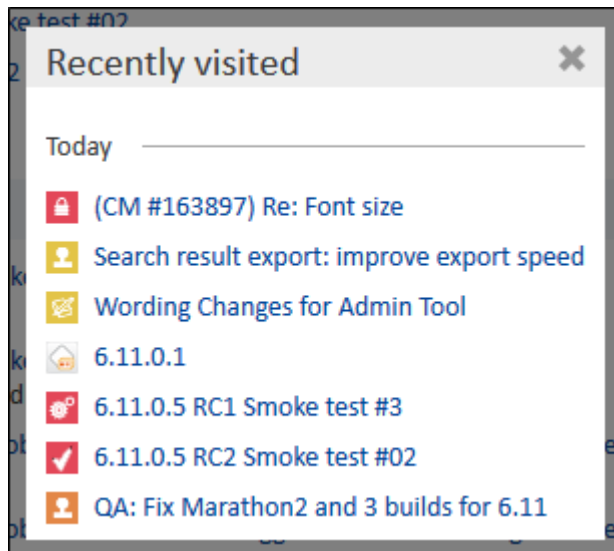
1.2.3 Dashboard widget listing the last viewed elements (#630465)

A second new dashboard widget is introduced with this **ConSol CM** version. This widget lists the data objects the engineer has opened in the **Web Client** most recently.



The widget displays a list showing the latest history of data elements visited by the engineer currently logged in. It just shows a list of the standard display identification name of the object together with its (current) symbol as shown above.

On the right side the header bar of the widget displays the command link *show all* which opens the same list in an overlay over the grayed out page. The *show all* list presents up to 50 items previously viewed.



There is no data script required and the widget configuration can be fully made by page customizations.

The layout attribute for the type *widgetsGrid* with the scope */welcomePage* on the welcome page defines its presence and position as common for widgets. The value below displays it full width on top another widget for listing recently changed tickets:

```
[recentlyVisitedObjects:RecentlyVisited][recentDataChanges:RecentChanges]
```

The other available configuration options can be set as page customization attribute values for the widget, in the example from above in the scope */welcomePage/recentlyVisitedObjects* for the type *recentlyVisitedWidget*. The available configuration attributes and their possible values are explained in the list below:

- *excludedCustomerGroups*: A comma-separated list of customer groups identified by their technical names. The named customer groups and their members will be excluded from the listing in the widget.
- *excludedItemTypes*: A comma-separated list of base object types which will be excluded. The types are identified by the following keywords:
 - TICKET: All tickets will be excluded from the listing in the widget.
 - UNIT: All customers (companies and contacts) will be excluded from the listing in the widget.
 - RESOURCE: All resources will be excluded from the listing in the widget.
- *excludedQueues*: A comma-separated list of queues identified by their technical names. The named queues and their tickets will be excluded from the listing in the widget.
- *excludedResources*: A comma-separated list of resource types identified by their technical names. The named resource types and their resources will be excluded from the listing in the widget.
- *localization*: Currently not used.
- *maxItems*: The maximum number of items shown in the widget main view (integer).
- *title*: The title string shown in the header bar, localization of this label is not available yet.

- **visible**: Widget visibility, must be set to “true” to have the widget on the welcome page, setting it to “false” hides the widget and all its contents.

maxItems	Number of maximum items which will be displayed. By default its 16.	10 (int) (+) (-)	
title	Recently visited items widget title	Recently visited (java.lang.String) (+) (-)	
visible	Indicates whether the widget is shown	true (boolean) (+) (-)	

The data gathering for this widget may show a performance impact occasionally. In such a case the data collection can be turned off by the system property `recent.items.persistence.enabled`, if this widget is not used in the system. The module `cmas-core-server` features more properties to control the discharge of obsolete data for the widget. All properties are created during setup and update. The table below shows all properties related to this widget.

Module	Property name	Property function	Default value
<code>cmas-core-server</code>	<code>recent.items.persistence.enabled</code>	Enable/disable data collection	“true”
<code>cmas-core-server</code>	<code>recent.items.max.per.engineer</code>	Maximum number of recent items stored per engineer	“500”
<code>cmas-core-server</code>	<code>recent.items.cleanup.interval.minutes</code>	Interval in minutes between two runs of the data cleanup	“60”
<code>cmas-core-server</code>	<code>recent.items.cleanup.cluster.node.id</code>	Node in a cluster setup on which to perform the data cleanup	(none)

1.2.4 Webhook integration interface (#631699)

ConSol CM introduces a new interface for integrating with third-party applications in this release, so-called webhooks. These are used by sending an HTTP POST request to **ConSol CM**, which is processed in the system in a defined way, and a tailored response, usually based on the request and the system content or status, is returned. Thus, it is an easy to use interface with other applications.



Please be aware that it is necessary to ensure the security of this interface in a production environment by exposing it only via proxy. Although basic security is built in, it is mandatory to secure such a simple, but powerful interface with third-party applications additionally, especially when allowing access to external third-party applications via Internet.

An instance of a webhook interface consists of two main elements:

- an **Admin Tool** script to define the behavior of the interface and
- a security (provider) configuration made via web interface

The webhook interface will only operate if both are valid. The script in the **Admin Tool** must be of the type *Integration*, which is newly introduced in the current release for this exact purpose. Choosing another script type will not enable the interface.

Once a script is available, the configuration web interface for this service is available under an URL composed as shown:

- `http://[CM-Server-Address]/intg/[Admin-Tool-Script-Name]/config`

For example: `http://10.20.3.40:8080/intg/etsearch/config`

The interface service, when configured properly, will then be available under a similar URL:

- `http://[CM-Server-Address]/intg/[Admin-Tool-Script-Name]/service`

For example: `http://10.20.3.40:8080/intg/etsearch/service`

The service can be accessed from remote computers via HTTP request to this address. The request **must** be a POST request to this address with the submitted data in JSON form. Other request types and payload formats are not supported.



Caution: The configured webhook interfaces are not part of scene exports and imports at this point!

Configuration web interface

When trying to access the configuration interface, a login page is shown:



It is strongly discouraged to expose this configuration page to external networks, since by default it can be accessed over an unencrypted connection.

For access, the credentials of a **ConSol CM** engineer with administrator privileges must be entered. Upon successful login, the configuration page is shown with a form to enter the relevant information:

- Display of the relative URL to access the webhook interface service
- Checkbox to enable/disable the service generally, must be checked for the interface to operate
- Security providers configuration:
 - IP-based filtering to limit access to defined IP ranges:
 - Checkbox to enable/disable IP-based filtering, must be checked for IP range based access limitation
 - List with IP address ranges in CIDR notation (see below)
 - Secret tokens configuration to require a shared secret for access:
 - Checkbox to enable/disable access by token, must be checked for access by token
 - Checkbox for *HMAC* to use a secure hash for message authentication (see below)

- HTML header name for the shared secret, by default “SECRET_TOKEN”
- List with access tokens

URL: /intg/etsearch/service

Logout

☒ Enabled

SECURITY PROVIDERS:

IP-BASED FILTERING

☒ Enabled

IP Addresses(CIDR Notation: X.X.X.X/X or X:X:X:X:X:X/X, one entry in line):

10.20.30.40/24

SECRET TOKENS

☐ Enabled

☐ HMAC

Header:

SECRET_TOKEN


Tokens(one entry in line):

Save

On the bottom of the page is a button to save and apply the configuration changes entered. On the top right a button to log out from the configuration page is shown.

The interface service will be **accessible only** if the checkbox *Enabled* on top of the page is checked and at least one security provider is configured. In case both are configured, the validity of the requester IP address is checked first. If the address is valid, the request is processed; if not, the request is checked for a valid shared secret. If this token is sent, the request is also processed.

The **IP-based filter**, when enabled by checking the checkbox, uses a list of IP address ranges defined in the CIDR notation (Classless Inter-Domain Routing). A valid entry in this notation is an IP address followed by a slash separating it from a number of bits defining the subnet mask. The example “10.20.30.40/24” stands for the IP address “10.20.30.40” and its subnet mask “255.255.255.0”. The address list for service configuration allows multiple IP ranges in this notation with each range entry on a separate line.


 Please consult the external documentation about CIDR (Classless Inter-Domain Routing) for more details on valid IP address range entries, if desired!

Using interface **access with secret tokens** must be enabled by checking the checkbox *Enabled* as well. Each token to be used must be added as a string on a separate line in the list *Tokens* in the last input box on the page.

The checkbox *HMAC* activates the validation of the request by a mandatory hash based on the token. The HMAC (Hash-based Message Authentication Code) value is derived from the unencrypted payload and the shared secret. This code is transferred in the header of the request. The only hashing algorithm supported by **ConSol CM** in this case is **SHA-1**, **MD5** is not supported!

A corresponding hashing of the response is currently not supported.

The input field *Header* allows to define the HTTP header field name which shall be used to transfer the shared secret token. The value defaults to "SECRET_TOKEN", but it is recommended to change this value, if possible. A non-standard name can mean a security improvement.

 Please be aware that for a webhook integration there is no other mandatory access validation beyond the security providers, specifically no login-based authentication as for engineers which imposes additional access restrictions. The interface has full system access, so securing it additionally with a proxy is mandatory for real-life usage.

Scripting examples

The script defined in the **Admin Tool** having the mandatory type *Integration* does implement the logic for dealing with the requests sent. The script name without the extension ".groovy" will also be the service name used for the URL to access the interface. So in the above example <http://10.20.3.40:8080/intg/etsearch/service> an *Integration* type script called **etsearch.groovy** is present in the **Admin Tool** for the interface logic. When the script is executed for processing a request, it runs in an administrative context and has full system access.

Adding a script of this new type puts a basic script template in the editor as a start. The template includes imports for JSON helper classes easing interaction with JSON objects. The script has the request headers and payload available in the **headers** and **payload** variables, with headers being a map and payload being a JSON object. The return value of the script must be JSON as well.

The example script below returns a simple JSON object with ticket number, subject, and engineer as response on a request with the ticket number. It returns "n/a" for invalid ticket numbers or a missing engineer assignment. Before returning the response, it logs the request and response content.

```

import groovy.json.JsonSlurper
import groovy.json.JsonOutput
import com.consol.cmas.common.service.TicketService

def jsonSlurper = new JsonSlurper()
def message = jsonSlurper.parseText(payload)

def ticketname      = "n/a"
def ticketsubject   = "n/a"
def engineername    = "n/a"

if (message.ticket) {
    ticket = ticketService.getByname(message.ticket) ?: "n/a"
    if (ticket) {
        ticketname      = message.ticket
        ticketsubject   = ticket.getSubject()
        if (ticket.engineer) {
            engineername = ticket.engineer.getFirstname() + " " + \
                ticket.engineer.getLastname()
        }
    }
}
log.info "Webhook [Search engineer for ticket] returned {ticket: " + \
    ticketname + ", subject: " + ticketsubject + ", engineer: " + \
    engineername + "} on request [" + message + "]"

return JsonOutput.toJson([status: 'OK', ticket: ticketname, \
    subject: ticketsubject, engineer: engineername]);

```

The receiving application is responsible for parsing and digesting the response sent by **ConSol CM**. The JSON response can be crafted to meet this third-party application's requirements regarding the JSON response.

A command line (test) request accessing the interface defined by the above script looks like this for example:

```

# Linux:
curl -X POST http://10.20.30.40:8080/intg/etsearch/service //
-H 'content-type: application/json' //
-d '{"ticket": "110333"}'

# Windows (note the different quoting/escaping for the -d option):
curl -X POST http://10.20.30.40:8080/intg/etsearch/service //
-H "content-type: application/json" //
-d "{\"ticket\": \"110333\"}"

```

The request payload in this example is the value of the -d option:

```
{ "ticket": "110333" }
```

The response generated by the script and returned to the requester is a little more elaborate (Please note that this JSON has been heavily reformatted for better legibility!):

```
{
  "status"    : "OK" ,
  "ticket"    : "110333" ,
  "subject"   : "License Review 09.08.2017" ,
  "engineer"  : "Harald Huber"
}
```

When using the shared secret security provider, the token is accessible for the script from the **headers** map in a field named as defined in the configuration web interface under *Headers*. Provided the default "SECRET_TOKEN" is used, it is available in the script from this HTTP header field. This can also be returned explicitly in the response, if desired, please compare the example code snippet:

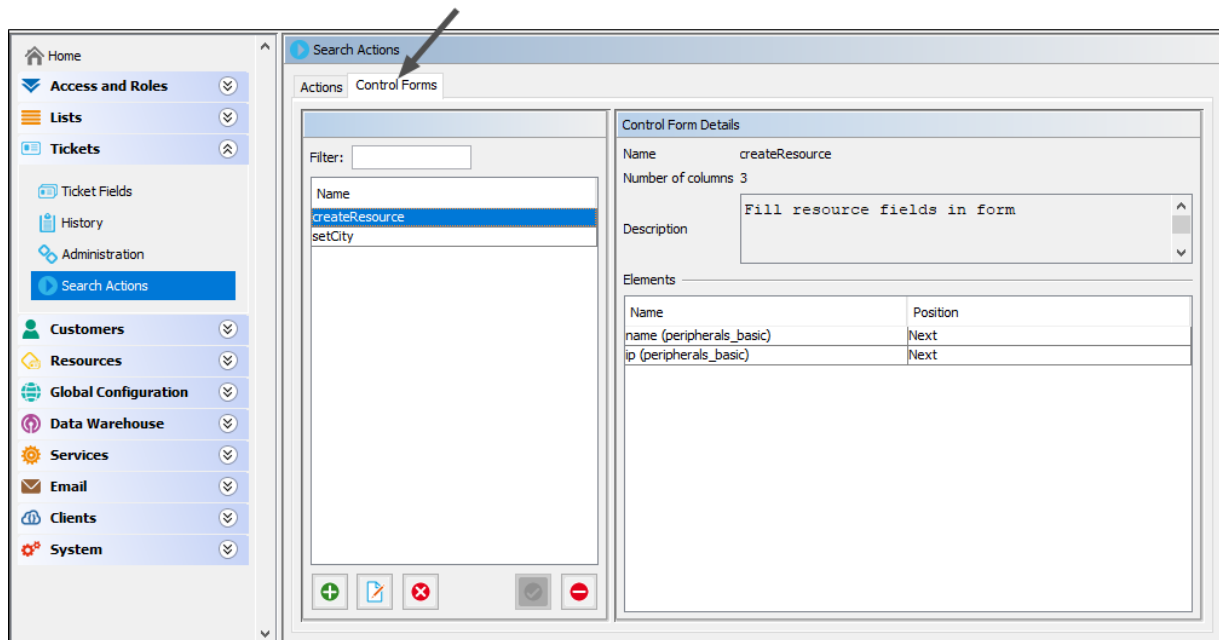
```
...
token = headers.get('secret_token')
...
return JsonOutput.toJson([status: 'OK', ticketId: ticket.getId(), token:
headers.get('secret_token')])
```

1.2.5 Forms for actions beyond workflow (#629429, #629430)

The data entry forms related to activities (ACFs) in **ConSol CM** are very common in the ticket workflow context. This type of functionality is now available in all action contexts beyond the workflow like customer or resource actions as well as search actions.

Action forms support all data types including dependent enums and scripted autocomplete fields.

The **Admin Tool** features new screens to configure action forms. The corresponding tab is called *Control Forms* and it is located next to the tab where the actions for the respective object are configured, i.e., in the navigation group *Tickets*, navigation item *Search Actions*, in the navigation group *Customers*, navigation item *Actions*, and in the navigation group *Resources*, navigation item *Actions*. The *Control Forms* tab shows a list of action forms for the respective type of object on the left and the form details on the right.



The creation and editing of an action form is similar to the existing Activity Control Forms. The main difference is that all data fields are available in action forms, i.e., it is possible to use data fields which do not belong to the object for which the action is performed. This enables the user to manipulate other objects within the action form.

Above the filter by data field group, the screen for configuring an action form features an additional filter by object type to display only ticket fields, customer fields, or resource fields.

Edit Resource Action Form

Please fill in required fields.

Details

Name: ReviewPatchData

Description: Please review the patch data before proceeding

Attributes

Number of Columns: 3

Custom Fields

Name	Position
name (coreReleaseData)	Next
patch_id (coreReleaseDa...)	Next
releaseDate (coreReleas...)	Next
state (coreReleaseData)	Next

Available

Filter by type of object

Filter by data field group

Save Cancel

Customer and resource action forms can be linked to manual actions or search actions and ticket action forms can be linked to search actions.

The respective screens for creating and editing the action have been extended with four new fields: *Control Form* for linking the action form, *Form Condition Script* for providing a *Control Form Condition* script, *Form Prefill Script* for providing a *Control Form Prefill* script, and *Control Form required fields* for selecting the required fields of the action form.

Edit resource type action

Please edit resource type action's data

Name:

Type:

Condition Script:

Action Script:

Control Form:

Form Condition Script:

Form Prefill Script:

Description:

Name	Required
name (coreReleaseData)	<input checked="" type="checkbox"/>
patch_id (coreReleaseData)	<input type="checkbox"/>
releaseDate (coreReleaseData)	<input type="checkbox"/>
state (coreReleaseData)	<input type="checkbox"/>

Control Form required fields

When the user clicks the action in the **Web Client**, the action form is displayed at the top of the customer, resource, or search page, provided that the control form condition script, if any, returns “true”.

Release

Review release data
Please review the patch data before proceeding

Name: * ID: X Release date:

State:

6.11.0.1
Releases Core Release
Internal | Created: 9/22/15 9:53 AM
Name 6.11.0.1 ID 6.11.0.1
State Planned

Activities
Start Release
Refresh assigned tasks
Review release data

Workspace
Workspace is empty
All your unsaved tasks are automatically listed in this workspace.

Favorites
Favorites are empty
Drag tickets, contacts

By default, the data fields which belong to the context object in which the action form is opened are mapped automatically. This automatic binding can be disabled by adding the following lines of code to the action script:

```
def isAutoBindingEnabled() {  
    return false;  
}
```


There are two new objects available in action scripts, form condition scripts, and form prefill scripts:

- **formFields** (`Map<FieldKey, AbstractField>`)
Data fields with values from the action form. It may be null if no action form was used or the variable was called in a form prefill script. It is only available in precondition scripts after the action form was filled.
- **controlForm** (class `TicketActionControlForm`, `UnitActionControlForm`, or `ResourceActionControlForm`, depending on the context)
Definition of the action form. It may be null if no action form is used. It is only available in precondition scripts after the action form was filled.

The interface `ControlFormService` has been extended to provide methods for working with the new action forms.

If automatic binding is enabled, the variable referring to the context object (depending on the context, `ticket`, `resource`, `contact` / `company`) contains the data fields filled in the action form.

The changes done to these objects within a form prefill script are only used to fill the action form. They are not persisted.

 The templates for action scripts have been updated to include the default setting for automatic binding and hints about the work with data from action forms.

Scripting examples

The **Form Condition Script** can be used to control whether the action form should be displayed or not. The following code example checks if there are resource fields which are not filled out before displaying the action.

```
if(resource.get("product.quantity") == null || resource.get("product.price") ==  
    null) {  
    return true  
}  
  
return false
```

The **Form Prefill Script** can be used to control which values are displayed when the action form is opened. If automatic binding is enabled, the values from the data fields belonging to the context object are already prefilled. The script can be used to overwrite these values or to prefill values to data fields which belong to other objects. The following code example prefills an action form with set values and adds a new row to a struct.

```

offices = "location.offices"

struct = new Struct()
struct.set("offices_struct_roomNumber", "12")
    .set("offices_struct_streetName", "Franziskanerstraße")
    .set("offices_struct_streetNumber", "38")

resource.add(offices, struct)
resource.set("location.city", "München")

```

The **action scripts** can use the values from the action form to create or update objects. If autobinding is enabled, the context object (single customer or resource, or ticket, customer or resource found in the search) is updated automatically with the values entered in the data fields belonging to it. The following code example is used for a search action. It updates the tickets found in the search with the delivery date entered in the action form.

```

def isAutoBindingEnabled() {
    return true;
}

for(Ticket t : tickets){
    t.set("orders.delivery_date", ticket.get("orders.delivery_date"))
}

client.success().withRefreshContent();

```

The values entered in other data fields can be used to manipulate other objects or perform internal calculations. The values of these data fields are accessed by using the **formFields** object. The following code example is used in a customer script. It opens the create ticket page. The values from the action form are prefilled and the current customer is already selected for the new ticket.

```

def isAutoBindingEnabled() {
    return true;
}

//get fieldkeys and retrieve values from formfields map
FieldKey ticketPriority = new FieldKey("helpdesk_standard", "priority")
priority = formFields.get(ticketPriority).value
FieldKey ticketModule = new FieldKey("helpdesk_standard", "module")
module = formFields.get(ticketModule).value

//create ticket and set values
Ticket ticketObject = new Ticket()
ticketObject.setSubject("Ticket subject set in customer action script")
ticketObject.setQueue(queueService.getByname("HelpDesk_1st_Level"))
ticketObject.set("helpdesk_standard.priority", priority)
ticketObject.set("helpdesk_standard.module", module)

//go to create ticket with ticket prefilled and unit from context of execution
client.goToCreateTicket(ticketObject).withCustomer(unit);

```

1.2.6 Unified interface for action-related operations (#629296)

The interfaces for user-triggered manual operations after executing an activity/action script which exist for different contexts have been unified to a common interface structure and extended in functionality. The new interface is available for both manual actions and automatic actions (create, update, delete, relation) in the context of customers and resources, as well as for ticket activities. It can be used in action scripts and workflow scripts, as well as directly in the **Process Designer**.

The previous mechanisms `ActionScriptResult` and `postActivityExecutionHandler` are now deprecated and will be removed in a future release. The mechanism to display messages in the workflow context, `workflowApi.addValidationError()`, remains in place.

The new interface extends the following, previously available functionality:

- Open a ticket, customer, or resource page
- Create a new ticket, customer, or resource
- Create a new ticket with a given customer
- Open an URL
- Open an Activity Control Form (ACF)
- Show a message

In addition, it provides the following new functionality:

- Roll back an action, e.g., to enforce business rules on ticket, customer, and resource data and relations
- Concatenate methods, e.g., open a customer page and display a message
- Implement wizards using ACFs

The new class `OperationResponseBuilder` has been added to support these features and replace the previous mechanism. The following table shows the new methods together with the previous ones. The keywords “resource”, “ticket”, and “unit” in the examples refer to a resource, ticket, or unit object.

New class/method	Deprecated class/method	Note
<code>client.failure()</code>	<code>PostActionType.FAILURE</code>	Sets the <code>OperationType = FAILURE</code> .
<code>client.goToCreateResource(resource)</code>	<code>actionScriptResultFactory.getPostAction(PostActionType.CREATE_RESOURCE, resource)</code>	Opens the create resource page pre-filled with data from <code>resource</code> object.
<code>client.goToCreateTicket(ticket)</code>	<code>actionScriptResultFactory.getPostAction(PostActionType.CREATE_TICKET, ticket)</code>	Opens the create ticket page prefilled with data from <code>ticket</code> object.

New class/method	Deprecated class/method	Note
client.goToCreateTicket(ticket).withCustomer(unit);	actionScriptResultFactory.getPostAction (PostActionType.CREATE_TICKET, ticket, unit)	Opens the create ticket page prefilled with data from ticket and unit object.
client.goToCreateUnit(unit)	actionScriptResultFactory.getPostAction (PostActionType.CREATE_UNIT, unit)	Opens the create customer page prefilled with data from unit object.
client.goToPage("url")	actionScriptResultFactory.getPostAction (PostActionType.GOTO_PAGE, "url");	Opens the given URL.
client.goToResource(resource)	actionScriptResultFactory.getPostAction (PostActionType.GOTO_RESOURCE, resource);	Opens the resource page of the given resource object.
client.goToTicket(ticket)	actionScriptResultFactory.getPostAction (PostActionType.GOTO_TICKET, ticket)	Opens the ticket page of the given ticket object.
client.goToUnit(unit)	actionScriptResultFactory.getPostAction (PostActionType.GOTO_UNIT, unit)	Opens the customer page of the given unit object.
client.openActivityForm ("acfpath")	actionScriptResultFactory.getPostAction (PostActionType.GOTO_TICKET, ticket, acfExecutionContext)	Opens the ACF of the current ticket. Used in the workflow context.
client.goToTicket(ticket).openActivityForm ("acfpath")	actionScriptResultFactory.getPostAction (PostActionType.GOTO_TICKET, ticket, acfExecutionContext)	Opens the ACF of the ticket. Used in the Action Framework context.
client.rollback()	-	Rolls back the action. Optionally, an error message can be provided.
client.showDebugMessage ("messageOrLabel")	actionScriptResultFactory.getPostAction (PostActionType.FAILURE, "messageOrLabel")	Shows a red message, the message text can be entered as a string or using a label.
client.showErrorMessage()	actionScriptResultFactory.getPostAction (PostActionType.FAILURE, "messageOrLabel")	Shows a red message, the message text can be entered as a string or using a label.

New class/method	Deprecated class/method	Note
client.showInfoMessage()	actionScriptResultFactory.getPostAction (PostActionType.SUCCESS, "messageOrLabel")	Shows a green message, the message text can be entered as a string or using a label.
client.showMessage("messageOrLabel")	actionScriptResultFactory.getPostAction (PostActionType.SUCCESS, "messageOrLabel")	Shows a green message, the message text can be entered as a string or using a label.
client.showWarningMessage()	actionScriptResultFactory.getPostAction (PostActionType.SUCCESS, "messageOrLabel")	Shows a green message, the message text can be entered as a string or using a label.
client.success()	PostActionType.SUCCESS	Sets the OperationType = SUCCESS.

The default script templates inserted in the **Admin Tool** when creating a new script for actions and other use cases have been adapted slightly to be more consistent in content and formatting.

Scripting examples

The following section contains some examples of code used in scripts.

Messages

The new API allows you to show different kinds of messages on the **Web Client** GUI.

The following code example is used in a ticket search action. The tickets found in the search are updated with a new delivery date, which the user enters in an action form. Afterwards a customized success message, which is managed as a label, is displayed.

```
def isAutoBindingEnabled() {
    return true;
}

//update ticket
for(Ticket t : tickets){
    t.set("orders.delivery_date", ticket.get("orders.delivery_date"));
}

client.showInfoMessage("delivery.date.update.success").withRefreshContent();
```

The standard success message can be displayed by using `client.success().withRefreshContent()`; instead of `client.showInfoMessage`.

Rollback

The `rollback()` method can be used to enforce business rules, e.g., to prevent that objects are updated with incorrect data.

The following code example is used in an automatic resource action of the type *Update*. It checks the value entered in the field holding the next maintenance date and informs the user that only future dates are allowed. The user cannot save the resource if a past date is entered.

```
current_date = new Date();

if(resource.get("product", "next_maintenance") < current_date) {
    client.rollback("The date for the next maintenance cannot be in the past.");
}
```

The same procedure can be used for customers. To perform a rollback for tickets, the `rollback()` method can be added to the script called by a business event trigger which fires when the respective ticket fields are updated.

Wizards using ACFs

The `openActivityForm()` method can be used to implement wizards. There is one ACF for each step of the wizard and upon saving the data entered in an ACF, the next ACF is opened automatically. To implement such a wizard, you create one workflow activity for each ACF. Then you add the code to open the next ACF to the script of each workflow activity.

The following code example is used in the script of the workflow activity to accept an order. It opens the ACF where the quote is prepared, which belongs to the next workflow activity and is the next step in the process.

```
client.goToTicket(current_ticket).openActivityForm("defaultScope/Analyse_
request/Prepare_quote")
```

To continue the wizard and open another ACF, a similar piece of code can be added to the script of the workflow activity where the quote is prepared.

1.2.7 Parallel editing handling for customers and resources (#630442)

The parallel editing of a ticket by two engineers at the same time has been introduced in **ConSol CM** in version 6.10.5.1. Since then the changes made by both engineers are not lost as long as these changes do not conflict. This of course is different, if the engineers edit the same data field or one of them closes the ticket. In such a case the engineer completing the operation last is informed about the outcome of the operation by a qualified message.

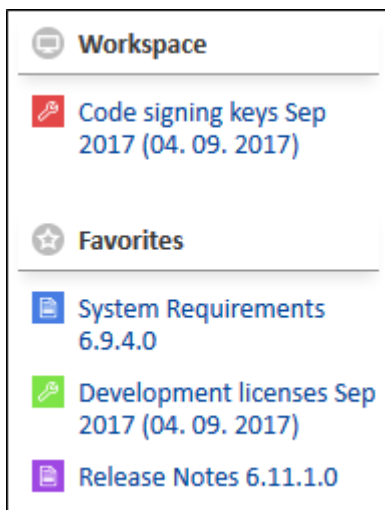
This parallel editing was not yet implemented for customers and resources, always rejecting the changes of the second engineer with a message that the object must be refreshed first and those changes were lost. This release allows for parallel modification of both customers and resources without the changes entered second being lost on the save attempt. The detailed behavior regarding engineer actions with potential for conflict will be explained in the manuals. Some of these specific combinations may require a page reload to show all data, status, and history changes afterwards, since deactivation, deletion, and action execution together with data modification have different internal implications. This may also hold true, if one of the engineers cancels his operation.

This new behavior for customers and resources can be disabled by a new system property in case the handling appears intransparent to the engineers or some negative performance impact is observed. The property can be found in the module `cmas-web-server-adapter` under the name `diffTrackingEnabledForUnitAndResource`. Setting its value to “false” in the **Admin Tool** will disable the new parallel editing capabilities. The property is added during setup and update with the default value “true”.

Please see [Concurrent editing handling optimization for customers and resources \(#631572, #631739\)](#) for additional changes introduced in the scope of this new feature.

1.2.8 Colored ticket symbols in favorites and workspace (#626079, #630889)

The favorites and workspace listed on the right side of the **Web Client** page located below the available activities only showed a fixed standard symbol for ticket entries with no dynamic variation in color in earlier **ConSol CM** versions. This version introduces dynamic symbols for the ticket elements in these two lists. The symbol shown for a ticket in the favorites and workspace is exactly the scope icon with the color as defined by its current workflow state. This additional information may help assessing the ticket significance for further actions. The image illustrates this new information compellingly.



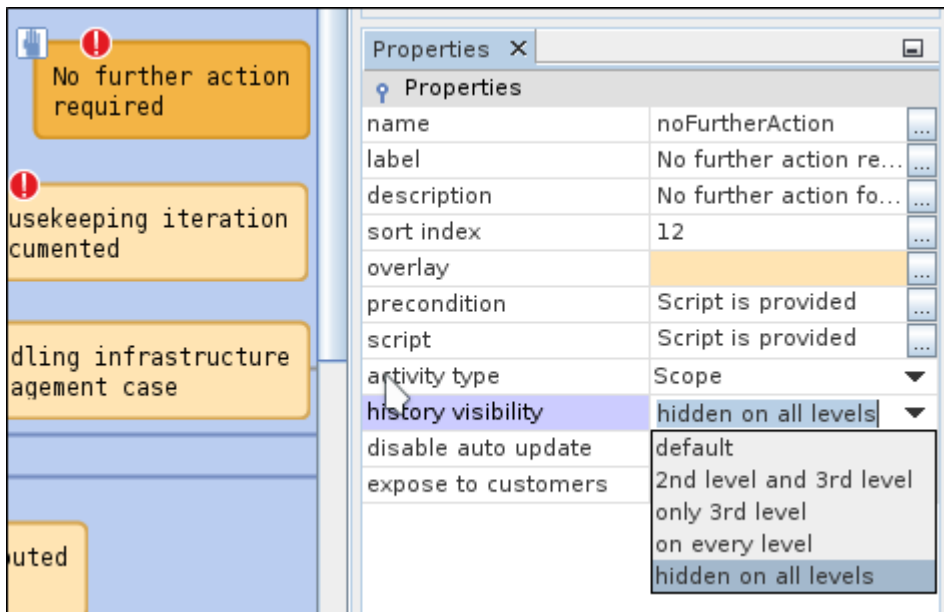
1.2.9 Specific page titles (#625806)

The page title in the **Web Client** used to be uniform for all the pages. This was the reason why the browser window/tab always displayed the same information, namely the **ConSol CM** product name with the release version. This uniform label, while being reasonable for the browser window/tab, on the other hand heavily constricted navigation in the browser history where every visited page had the same label. This has been changed, so that now the page title is specific for each page. This is a more informative label for the browser window/tab and also provides a highly informative list in the browser history. Depending on the content of the page this is either the ticket subject, the name of the customer or resource defined by the standard template, or a descriptive standard name for the operation performed on this page like *Login*, *Create ticket*, *User profile*, or *Text templates administration* among others. This label in the title is extended with the information shown previously as page title. Thus, with this extension page titles are helpful in more contexts than before.

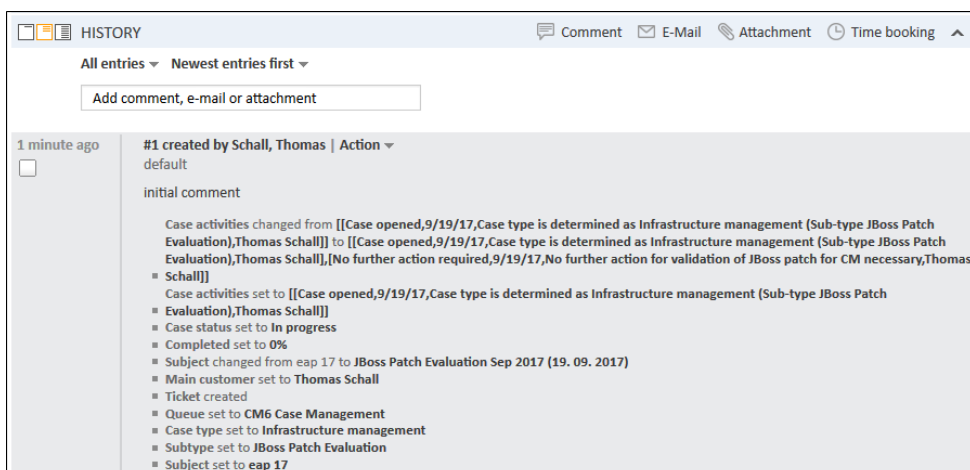
1.2.10 Workflow activity execution not listed in history (#631304)

A new option has been added to the **Process Designer** for workflow activities with regard to their history visibility. This option when selected will result in no visible entry in the ticket history from execution of this activity. This can help to keep the visible history entries more relevant, but also means that some operations on the ticket may not show in the ticket history at all.

The new option is labeled *hidden on all levels* when determining the history visibility for a single activity selected in the **Process Designer**. In the image below this option value can be seen for the selected activity *No further action required*.



The execution of this activity will not have a visible entry in the history at all. This can be seen in the screenshot from the **Web Client** below. Immediately after creating this ticket the activity *No further action required* has been executed. In the history entry no line about this execution is present even though the execution shows in the changed value of the ticket field *Case activities* recorded in the topmost history element.



1.2.11 Custom annotation usage in *Web Client* custom projects (#624043)

The internal implementation of custom annotations defined in the **Admin Tool** has been adapted. This change to an interface now allows for defining its own handler in a **Web Client** custom project adding a new level of flexibility to the usage of annotations in custom projects.

1.2.12 Widget style customization with CSS (#630386)

New classes of CSS styles have been added to allow styling customizations of dashboard widgets in custom projects. The new classes are added for modifying the `<div>` element enclosing widgets. There are two types of classes available for all widgets of the same widget type and for each individual widget:

- `widget_<widget_type>` is the class for all widgets of a given type.
- `<widget_name>_<widget_type>` is the class for a particular widget.

In a specific widget configuration several classes are automatically provided. For an example, the configuration may look like this: `[kpi:KPI, kpi_1:KPI],[ticketsInView:Chart]` having three different widgets of two different types. This would result in the following `<div>` tags with their CSS classes:

```
<div class="widget_KPI kpi_KPI"> ...
<div class="widget_KPI kpi_1_KPI"/> ...
<div class="widget_Chart ticketsInView_Chart"/> ...
```

These classes can be modified in the styles of a custom project.

1.2.13 New **Admin Tool** template function for abbreviating strings (#607391)

The templates defined in the **Admin Tool** can profit from a new function for trimming strings. This new function shortens a string value, for example from a field, to a specified number of characters and adds three dots to mark the ellipsis, if a part of the string is cut off. The new function is called `abbreviate()`:

```
abbreviate( String original_string, Integer length)
```

Its usage in any template in the **Admin Tool** template administration works just like any other method available in this context:

```
Company label: ${abbreviate(company.getFieldValue('company', 'legal_name'),
20)!}
```

One of the most important uses for this functionality are configurable ticket list entries which earlier had a fixed maximum line length. This function allows to control the length of strings and manually define line breaks for this usage. The hard-coded line length maximum also has been removed in the context of this change so it is strongly advised to use this function in the template used for the customers in the ticket list. The minimum abbreviation length is 4 (three dots and one character).

1.2.14 *Admin Tool* script templates for actions unified (#630726)

The script templates automatically inserted for actions in the **Admin Tool** when creating a new script for actions and other use cases have been unified after showing some slight variations earlier. The templates have been adapted slightly to be more consistent in content and style. They have been made uniform where possible and the example code has been updated to the recent changes for examples from the feature described in [Unified interface for manual operations \(#629296\)](#).

1.2.15 Web server adapter view object cache (#631275)

In this release a new internal cache for view object data has been implemented in the application server. This web server adapter cache is a cache for view objects of the **Web Client** in the application server for reuse in communication with the **Web Client** in order to access the database less often to prepare this specific type of view objects. It is turned off by default, but it can be activated by changing a system property value.

When changing the value of the system property `voCacheEnabled` in the module `cmweb-server-adapter` from “false” to “true” this cache is activated. For advanced configuration and optimization there are internal server settings, for example, to allocate more cache space. Please contact your **ConSol** representative in case you should see the need for such action!



Note that there can be a minor unwanted interaction between the enabled cache and the lazy loading feature as described below!

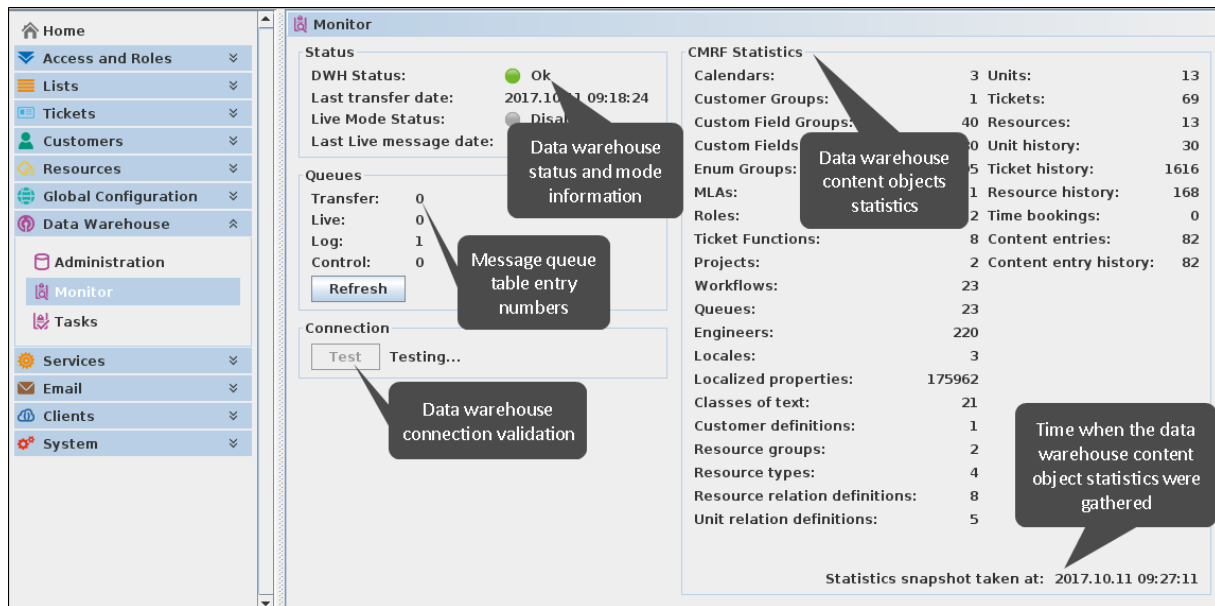
After expanding ticket sections in the **Web Client**, the page presentation regarding collapsed and expanded sections after a page reload does show expanded sections as before the reload and not the standard initial state, if configured collapsed. This is due to the fact that the information in the expanded section is cached.

1.2.16 Data warehouse monitor in *Admin Tool* (#630811)

The recent improvements of the data warehouse administration in the **Admin Tool** are complemented in this release with a new data warehouse monitor panel to inform the administrator about the current status of the data warehouse.

The monitor provides information regarding several aspects of the status:

- General operational status including **LIVE** mode,
- Live mode queues status
- Connection validation
- Content object statistics



The section *Status* shows the current operational status, the last date of a transfer initiated from the **Admin Tool**, the **LIVE** mode status, and the date and time of the last **LIVE** mode message.

The color icons for the status lines correspond to these specific messages:

Status symbol color	DWH status message	LIVE mode status message
green	OK	OK
yellow	Uninitialized	Uninitialized
	Action is processed	Action is processed
	Action is paused	Action is paused
	Action is planned	Action is planned
		Started (update is needed)
red	Error	Error
		Error (update is needed)
grey	Disabled	Disabled

The *Queues* section informs about the message entries in the internal data warehouse message queue tables:

- *Transfer*: Number of messages in the data warehouse transfer table (**INT_TRANSFER_QUEUE**)
- *Live*: Number of messages in the LIVE messages table (**INT_LIVE_QUEUE**)
- *Log*: Number of messages in the log messages table (**INT_LOG_QUEUE**)
- *Control*: Number of messages in the control messages table (**INT_CONTROL_QUEUE**)

The button *Refresh* queries the system to refresh these values.

In the section *Connection* only the button *Test* is available to validate the data warehouse connection currently configured. Right next to the button the current validation activity or the last result is displayed in a short message text.

The large section on the right side labeled *CMRF Statistics* shows detailed information about the count of the different content objects in the data warehouse. On the bottom below this extensive list the date and time of when these statistics were gathered is displayed.

1.2.17 Configuration import transaction encapsulation (#630480)

The import of the configuration data in a scene is separated into several database transactions. This can help in limiting memory usage on the database server. However, in the case of fatal errors during the configuration items import this could result in an inconsistent system state.

A new system configuration property has been introduced which additionally encapsulates all configuration items of a scene import in one large database transaction. Therefore, when enabling this option a fatal error on import of a configuration item cannot leave the system in an inconsistent state. Rather the whole encapsulating transaction will be rolled back putting the system configuration-wise into the state before the import. The transaction encapsulation does not include localization strings or runtime data which are handled independently of this setting.

The property to enable this option is called `config.import.global.transaction.enabled` and located in the module `cmas-core-server`. The configuration import encapsulation in one large transaction is enabled by setting the property value to “true”. The value “false” will result in the previous behavior described above which is also the default system behavior.



Please be aware that enabling this transaction encapsulation may result in longer import execution times and higher memory requirements on the database server!

1.2.18 CM/Track user switch history entry (#627732)

In previous releases of **ConSol CM** after switching the associated **CM/Track** user profile of a contact the contact history did not show this change. This has been extended so that a new entry line will report the switching of the contact from one assigned **CM/Track** user profile to another one. This line will inform about the previously assigned user profile as well as about the new one.

1.2.19 CM/Track user definition handling (#625328)

The handling and presentation of the definition of a so-called **CM/Track** user profile for a contact to access **ConSol CM** has been made more explicit and transparent by extending the configuration. This enhancement brings more flexibility as well as improved presentation in the **Web Client**.

A choice between three modes of assignment of a **CM/Track** user profile to a contact has been added. This explicit definition allows to show this assignment in the **Web Client** only, if the choice of a **CM/Track** user profile must be made explicitly:

- **Manual:** This value will allow/offer the choice of an available **CM/Track** user profile when editing or creating the contact in the **Web Client** or via REST API.
- **Fixed:** The definition of a **CM/Track** user profile is done in the **Admin Tool** for the whole customer group of the contact. There will be no choice available in the **Web Client** since the selection is predetermined from the customer group default, and an attempt to change it via REST API will return a status message `METHOD NOT ALLOWED` saying that setting the **CM/Track** user profile for this customer is not allowed.

- **None/Internal:** The association of a **CM/Track** user profile is not allowed in any client and can only be done by script, if desired. There will be no choice available in the **Web Client**, and an attempt to change it via REST API will also return a status message **METHOD NOT ALLOWED**. Also there is no default to be defined in the **Admin Tool**.

The assignment mode is defined in the **Admin Tool**, but reflects in the **Web Client** when editing a contact. The selection of a **CM/Track** user as the contact's user profile will only be shown for *Manual*. This does not affect the visibility of the data fields for the **CM/Track** login name and password for a contact which is a completely separate configuration.

The **Admin Tool** configuration can be found in the navigation group *Customers*, navigation item *Customer Groups*. The detail information block when selecting a customer group features a new section **CM/Track** which shows the assignment mode and the specific engineer used for fixed assignment.

The mode is defined in a drop-down menu on the bottom of the window when creating or editing a customer group.

When choosing *Fixed*, the second selection on this line will become active to determine the user profile to be used for all contacts of this customer group. This specific mode cannot be changed anymore in the **Admin Tool** as long as there are any customers in the customer group.



Please be aware that the assignment mode *Fixed* cannot be changed anymore as soon as the customer group has contacts!

When changing the mode from *Manual* to *Fixed*, this will not reflect in the customer history since this is not a change made by a regular engineer in the **Web Client**. Additionally when a script attempts to change the assignment in this case, for example from a workflow action, and fails, this will not be present in the contact history but only in the log files.

The mode choice will also be available for company-only data models even though it will not have any visible effect in this particular usage.

There are no additional changes in restrictions regarding the selection of a user profile for a contact. This means that it is still possible to assign a **CM/Track** user with insufficient permissions for the user group in question. It is up to the administrator to ensure sufficient privileges, otherwise an **InsufficientPermissionException** will be returned by the REST API.

During the update the user profile assignment mode *Manual* is automatically chosen which will keep the previous behavior intact.

1.2.20 Database performance logging (#631213)

This release adds another logging aspect to the **ConSol CM** system. This addition provides logging of database performance.

The entries in this new log file provide detailed information about database access times of atomic operations and may look like the following examples:

```
2017-09-15 13:54:42,218 DEBUG [ervice.aspect.DaoMetricsAspect] [ticket(100791)-a2c2cdd8-9a0c-11e7-bcf2-652ffaafaf00] myloginname, GenericDao.getAll, 7 ms
2017-09-15 13:54:42,248 DEBUG [ervice.aspect.DaoMetricsAspect] [ticket(100791)-a2c2cdd8-9a0c-11e7-bcf2-652ffaafaf00] myloginname, TicketDao.getByName, 11 ms
```

The corresponding new log file is named **operationtimes-db.log** and it is located in the same standard log directory as other log files. Its output can be fine-tuned with the use of two new system properties. These two properties control the filtering of entries which will be added to the log file. The new properties are part of the module **cmas-core-server** and their names are these:

- **dao.log.threshold.milliseconds**: Filters entries so that only those operations are logged in the file which take longer than the specified value in milliseconds, defaults to the value "5000".
- **dao.log.username**: Filters entries so that only those operations are logged in the file which are associated with the specified engineer login name, has no default value. Only one single user name can be supplied, and the value must exactly match the desired engineer login.

The database performance logging is generally active when properly configured. It can be deactivated indirectly by supplying a very high duration threshold or an unused login name, effectively filtering every entry.

These properties are added during setup and update with the specified default value for the first property. In a new setup this kind of logging is active by default. In existing installations it needs to be configured after the update to be activated. The configuration requires the addition of a handler and a logger entry in the logging subsystem section of the configuration file **cm6.xml** (or the corresponding file used in the particular instance).

The default configuration on a freshly installed system is the following setup:

```
<subsystem xmlns="urn:jboss:domain:logging:1.5">
[...]
```

```
<size-rotating-file-handler name="DAO_OPERATIONS_LOG" autoflush="true">
  <level name="DEBUG"/>
  <formatter>
    <pattern-formatter pattern="%d %-5.5p [%30.-30c] [%X{context}-%X
{sessionId}] %m%n"/>
  </formatter>
  <file relative-to="jboss.server.log.dir" path="operationtimes-db.log"/>
  <rotate-size value="300m"/>
  <max-backup-index value="6"/>
  <append value="true"/>
</size-rotating-file-handler>
[...]
```

```
<logger category="com.consol.cmas.core.server.service.aspect.DaoMetricsAspect"
use-parent-handlers="false">
  <level name="TRACE"/>
  <handlers>
    <handler name="DAO_OPERATIONS_LOG"/>
  </handlers>
</logger>
```

1.3 Changes

This section illustrates all relevant software product changes for version 6.11.1.0 of **ConSol CM**.

1.3.1 Layout improvements

Missing spacing between fields in a row (#630922)

There was no spacing between two neighboring fields in the same row, but rather the label of the second field was placed immediately adjacent to the entry field of the first one. This missing spacing has been added, so that the association of labels and fields is clearer for several fields in a row.

Headline label for main customer removed (#631842)

The extra headline label for the main customer of a ticket has been removed since it does not provide any information. The topmost customer always is the main customer. Therefore it is not necessary.

Current ticket not offered for creating a relation (#630838)

The current ticket will not be offered anymore when trying to create a ticket relation. Previously it could be chosen, but then showed a validation error, so that creating a circular relation was prevented already.

Gap between checkbox and its label too narrow (#630944)

The distance between a checkbox and its label was too small which impaired the readability. This gap has been widened for a better visual appearance and readability.

Initial gap for empty resource relation section removed (#630121)

The resource relation section on the ticket detail page showed a larger vertical gap to the next section header, if it was expanded but no relation data existed to be displayed. This uninformative variation has been settled and the vertical spacing is now the minimal standard distance for collapsed and empty sections.

Improved queue selection in detail search (#629784)

The selection of queues on the detail search page has been changed to the same kind of selection as for engineers. Thus, the new selector allows filtering by the entered string and multiple selected queues in the same way as the engineer selection.

Switching detail search result tabs while validation message is displayed (#630097)

After introduction of the new skin it was prohibited to switch to another result tab on the detail search page while a validation message was displayed for a search field. This behavior proved as undesired and thus has been changed. Now it is possible again to switch the result tabs in this case.

Dialog when deleting customers or resources in new skin styling (#630276)

The confirmation dialog which is displayed when trying to delete a customer or a resource previously was the browser's default style dialog. This has been changed to a confirmation dialog in the new skin style introduced in **ConSol CM** 6.11.0.

Language dependent line width in resource relation boxes (#630627)

The line width in the resource relation boxes on the customer page was defined relative to the English *Add* command link. In other languages, like German, which use a longer word for this command this caused overlapping text on the top line with long resource relation names. This was most disturbing in the most compact relation view. The line width now has been adjusted to be dynamic, depending on the actually needed space for the command link.

Improved long label handling in workspace and favorites (#630712)

The line overflow and interaction with the delete symbol only appearing when the mouse pointer is over the label have been visually improved.

Confirmation dialog title bar visual improvement (#630713)

The new skin style confirmation dialog had minor visual deficits regarding its title bar. The properties of the title bar have been changed to address these and the visual appearance is now improved.

Ticket list engineer display only for tickets with engineer assignment (#630794)

The ticket list showed a line for the engineer for each ticket when configured to display the engineer assignment. This line was present, even if the ticket had no assigned engineer, then showing the label only. This unnecessary line has been removed and now the line with the assigned engineer is only shown, if the ticket actually has an engineer assignment.

Transfer ticket to another company input field wider (#630798)

The input field to select a new company when transferring data to another company was too narrow to even show the whole watermark text. It has been made as wide as the corresponding input field in the context of transferring data to another contact. It is now wider and shows the whole watermark text.

Page navigation bar for document template mail merge fields (#630954)

The page navigation bar which should be shown for the mail merge fields of document templates when many of these fields are present occasionally was not present where it should be. This deficit has been resolved so that the navigation bar is displayed when it should be.

Date-time entry too large for table cell in a STRUCT (#630957)

The entry fields for date fields with the accuracy date-time to include a time took more space than the table cell offered, if the field was part of a STRUCT. The table cell for entry of such a date field in a STRUCT has been enlarged to offer enough space for the input fields.

Transfer data button on company page with standard styling (#630959)

The button labeled *Transfer data and continue* on the company page for transferring data to another company had the standard button styling. It has been changed to the different styling for a default action button to be in line with the other transfer data operations.

Visual improvement of left text border in history entries (#630962)

The left border line of a history text separating it from the timestamp and the text class selector has a changed styling now. This especially improves the appearance with colored classes of text. Now the line always appears darker than the background even on darker background colors for a class of text.

Date-time entry for *Exchange* calendar appointment layout (#630994)

The date-time value entry for appointments in the section of an integrated **Microsoft Exchange** calendar has been improved. Whereas the date selector and the time entry previously always were on separate lines, they now are on one line. The date selector which previously was overly wide has been reduced in its width.

Text template inactive action text formatting (#630998)

In the table of the text template administration the inactive actions for a template had the same text formatting as available operations in the new skin. This has been adapted so that unavailable commands are now presented in italics with a lighter gray font color than the other ones, just like in the previous skin.

Omitted label replacements for term “resource” (#631012)

For a small number of labels the term “resource” was not correctly replaced with the term defined in the **Admin Tool Labels** panel. For example, message texts when transferring data to another company or contact were affected. This has been modified so that all known omissions have been removed and the replacement should be uniform now.

Font color initially too light for collapsed ticket relation section (#631015)

The font color of the ticket relation section header was too light, if it was collapsed, but relations existed. It showed the color of a section with no content. This deficit has been addressed so that the font now is correctly in the darker color of a section with content.

Color theme for context menu icons on **Admin Tool** data warehouse panel unified (#631016)

The symbols shown in the context menu of the data warehouse panel in the **Admin Tool** recently were having a common color. The button symbols on the panel differed in their color. This has been unified so that now the context menu icons each have the same color as the corresponding button symbol.

Company symbol placement on the company edit page (#631024)

The company symbol was placed badly on the company edit page sitting in the top left corner without any margin. This position has been adjusted to use the same margin as the symbols on other edit pages.

Cancel button removed from customer definition on ticket create page (#631225)

The *Cancel* button which was previously shown when adding or creating a customer on the create ticket page has been removed. This button serves no purpose since every ticket must have at least one customer. Thus, defining the customer of the ticket by adding or creating one cannot be omitted or canceled. The button has been removed for this reason.

Autocomplete fields right border overlap (#631324)

Parts of the right border line of autocomplete fields were overlapped by the white area of the arrow symbol in the field. Therefore the border was not visible. This overlap has been corrected by correcting the symbol position and the border line is visible throughout again as desired.

Identical German terms for autocomplete scripts in **Admin Tool** (#631427)

The German terms for autocomplete scripts in the **Admin Tool** filter list and the scripts table differed. This inconsistency has been eliminated and the correct German term is used in both places.

Small visual improvement of confirmation dialog title bar (#631593)

The title bar of confirmation dialogs has been changed internally which provides a minor general visual improvement in its appearance.

Resource Pool terminology adjustments in **Admin Tool** (#631604)

On the resources panel in the **Admin Tool** a few labels have been changed to better match the terminology generally used for the Resource Pool, for example in documentation.

Symbols on content page harmonized with workspace and favorites list (#631647)

The symbols in the workspace and in the favorites list were not the same as on the corresponding content page for the ticket, customer, or resource in numerous cases. These discrepancies have been eliminated so that now the symbol in the workspace and the favorites is just a smaller version of the one from the corresponding content page.

Unnecessary busy indicator removed (#631710)

In the **Web Client** a busy indicator was present when clicking on navigation bar links. This indicator was unnecessary since there was no waiting time, or alternatively the linked page did show one, so it was not visible usually. Therefore it has been removed.

Busy indicator customer data edit section coverage (#631214)

The customer data edit section on the customer pages was not fully grayed out by the busy indicator while updated data were processed. The top part was displayed unchanged which has been adjusted. Now the whole area is grayed out as expected.

Underline on *Cancel* button removed for ticket relation addition (#631924)

The text label on the *Cancel* button when trying to add a relation to another ticket on the ticket page was underlined. This format is not used in this context otherwise so the underline has been removed.

Display text simplified for most ticket list grouping headers (#629801)

The grouping headers in the ticket now feature shortened and simplified texts for many cases. Mostly the somewhat redundant term ticket has been omitted and occasionally the phrasing has been adjusted to this change.

Scrollbar added for large STRUCT lists in customer field group (#626847)

A large STRUCT with many fields located inside a customer field group in a tab could not be fully edited for the rightmost fields. These were not accessible in edit mode since no scrollbar was available and the fields were outside the visible area in the browser window. The scrollbar needed to access these fields has been added so that every field in a large STRUCT can be edited now.

[Ticket list queue filter only uses queues defined in view \(#631143\)](#)

The queue filter of the ticket list offered all available queues, even those which could not contain any ticket since they were not included in the view selected. This unnecessary multiplication of empty queue sections has been addressed and now only those queues are offered which are defined in the view. Therefore unreasonable choices of queues which cannot show any result for the current view are not available anymore.

[Fixed ticket list line length limit removed \(#607391\)](#)

The line length for values in the ticket list previously was fixed. This limit has been removed in the context of the new feature [New template function for abbreviating strings \(#607391\)](#). The functionality introduced there can now be used to limit the length of string values to be displayed on a line in the ticket list.

1.3.2 Performance improvements

[Accelerated display of large ticket history entries \(#630763, #630764\)](#)

The procedure to display ticket history entries, especially large ones, has been improved in order to work faster. For this end the internal display structure and operations to assemble the required data have been modified. This should result in shorter times to display an entry which may be best observable on very large history items.

[Workflow due trigger identification speedup \(#630468\)](#)

A highly complex database query used to identify triggers, which are due for execution, could have strong performance impact on the server. Adding a specialized database index helps speeding up this query massively and avoids its performance impact.

[Data warehouse dynamic table integrity check optimized \(#631297\)](#)

The integrity check of the dynamic tables of the data warehouse by the CMRF during system startup often was taking very long for larger installations. This procedure has been optimized internally and should perform notably faster now on all installations.

1.3.3 *CM/Track V2* improvements

[Availability of changes from *CM/Track V2* version 6.10.7.0 \(#632393\)](#)

This release features all new features, changes, and corrections introduced in *CM/Track V2* version 6.10.7.0. The full support introduced in the release 6.11.0.4 implies that this release incorporates all modifications developed for the mentioned version.

[New favicon for *CM/Track V2* in browser \(#632407\)](#)

The favicon of the *CM/Track V2* client in the browser window/tab has been exchanged to show the new product logo.

1.3.4 *WebLogic* application server support restored (#630596)

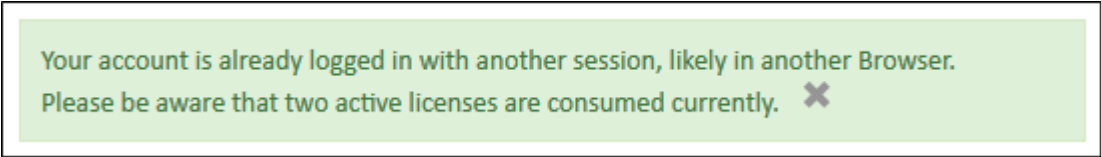
The support for the *WebLogic* application server product was suspended since the release of *ConSol CM* version 6.11.0 due to deficits in the application template for this server product concerning the use of CMRF. The template jar defects were corrected and this platform is supported again with this 6.11.1.0 release.

1.3.5 New splashscreen design and application icons for *Admin Tool* and *Process Designer* (#630919)

The **Java Web Start** applications of **ConSol CM** have been supplied with new splashscreen graphics and application icons to better match the design of the **Web Client**. This also includes new product related graphics in the *About* dialogs of both tools.

1.3.6 User login session handling enhancements (#623625, #630644)

The handling of issues with the login sessions of the current engineer has been improved in two important aspects. The user now gets an informational message immediately after login, if this newly created session is not the only login session that exists for this particular user. This message informs about the multiple active logins and the fact that each one of these consumes a license.



Your account is already logged in with another session, likely in another Browser.
Please be aware that two active licenses are consumed currently. X

This message can appear in several cases. The most common one is mentioned in the message itself: The engineer is logged in with two different browsers. Since both manage the session cookies separately different licenses are used. Another case happens, if an old login session cannot be used anymore because some browser operation turned the session cookie unusable. So it is not possible anymore to use the existing session cookie and associate it with the existing open session. In this case the message will appear on every login until the orphaned session expires.

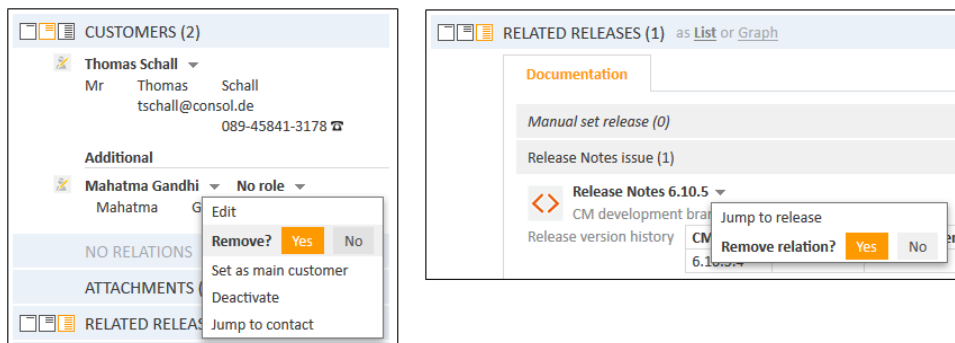
Besides the user message an enhancement of the system's logging has been implemented in order to deal with the latter case. Additional lines in the log inform about the problem with decoding the cookie information and associating it with an existing session. If there is no valid session that can be used the user is then shown the login page.

1.3.7 Undoing delete/remove operations replaced by confirmations (#627311)

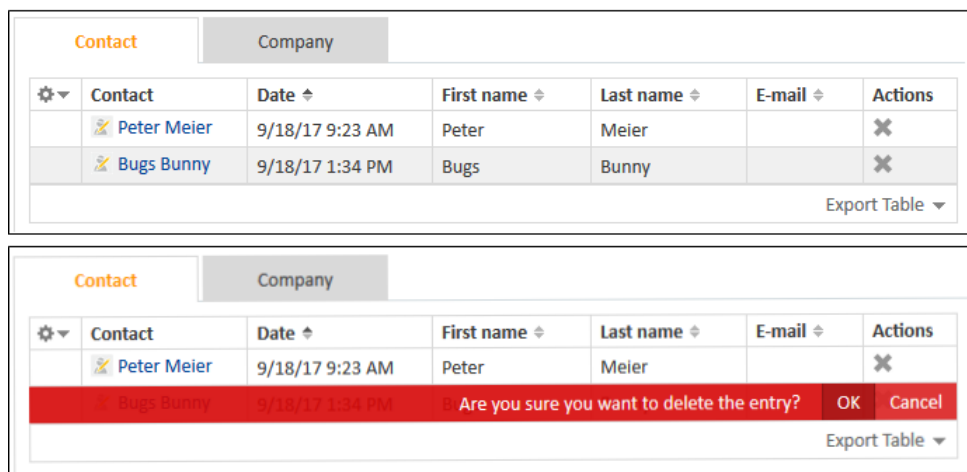
The behavior of the *Undo* command shown after removing some kind of relation between objects which was present in many places across the pages of the **Web Client** was varying and showed unexpected results in some contexts. Due to the different nature of the usage contexts unification of the behavior proved problematic. Therefore, in order to be able to offer a uniform behavior with respect to safe deletion of object relations the *Undo* functionality has been completely removed and replaced. When trying to delete such a relation now the engineer is confronted with a confirmation on the spot. After confirming the delete operation the relation is removed without an option to undo the operation. This affects among others relations between tickets and additional customers and engineers, relations among customers as well the different kinds of relation of resources to other objects.

There are basically two ways the confirmation on the spot is presented depending on the presentation of the *Delete* command.

A *Delete* command in a context menu must be confirmed on the same line of the context menu as the command itself. The images below show this for a customer and resource page:



A *Delete* command on the right side of a table row by clicking the cross symbol will turn the whole row to red and show the confirmation question and its options:



In either presentation form the relation between the objects will be removed after confirmation by clicking Yes . This cannot be undone. A relation deleted wrongly must be created again just like before when the *Undo* option was not offered anymore after a page reload.

1.3.8 Concurrent editing handling optimization for customers and resources (#631572, #631739)

The internal implementation of handling the concurrent editing of customers and resources by two different engineers has been optimized (also see [Parallel editing handling for customers and resources \(#630442\)](#)). Some cases of showing unexpected values after canceling data editing have been resolved with this optimization as well. The improvement provides improved consistency and reliability. In some specific cases it still may be necessary to reload the page after saving the own changes to see all changes by both engineers. However, the system behavior is unchanged in principle.


1.3.9 Relation targets filtered for existing relations (#631302)

The system offers possible target objects when creating a relation on several pages. The list of objects qualifying for the new relation which was offered in some cases did include the objects already in this exact kind of relation to the page main object. For example when adding an additional customer to a ticket the customers already associated with the ticket were offered as well. When choosing such an object a validation message was shown and the creation was prohibited. On other pages such a

selection simply was not provided. This inconsistency has been addressed and now target objects for which this exact kind of relation already exists are not offered anymore. Other types of relations, relevant especially for resources, are not affected.

The following pages and relations are all showing this behavior uniformly now:

Source object page	Target object relation
Ticket detail page	Additional customer
	Resource relation
Contact detail page	Contact relation
	Company relation
	Resource relation
Company detail page	Contact relation
	Company relation
	Resource relation
Resource detail page	Ticket relation
	Contact relation
	Resource relation

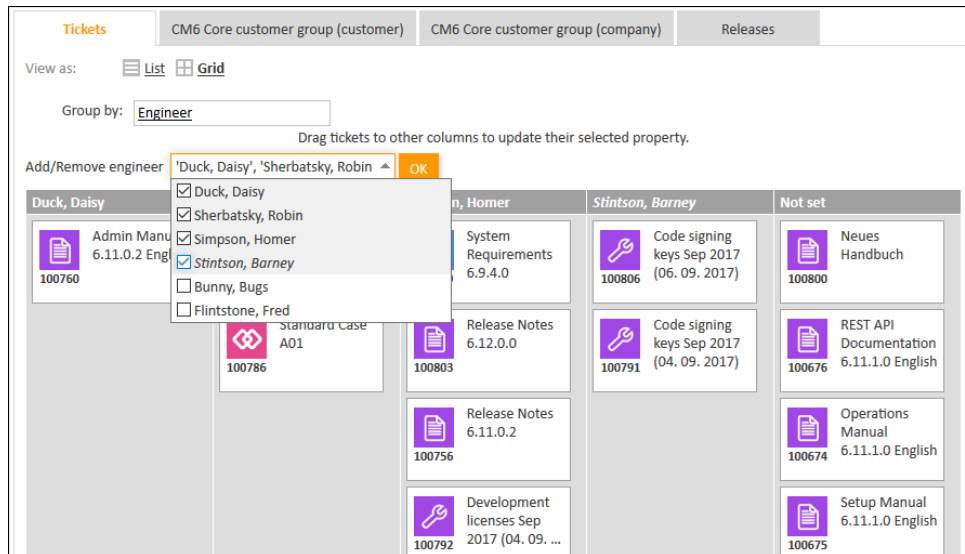
 Please note that when creating a relation between two different tickets a ticket with some existing relation still is offered for selection, since it is allowed to both have a *Reference* and a *Master-Slave* relation with the current ticket! This has technical reasons and may be addressed in a future change.

1.3.10 Usage of disabled ENUM values in grid view (#630995)

The display of disabled values from the ENUM field selected for grouping in the grid view display of search results has been removed in **ConSol CM** version 6.10.5.3 since these values may not be assigned to a ticket and thus are not a target for drag-and-drop. This implied in consequence that possibly not every ticket from the search was displayed, if some had the deactivated values assigned. Therefore these tickets could not be assigned a valid value in this view. The selective display of a column for a deactivated ENUM value has been re-enabled with the goal to allow this assignment operation.

The disabled entries of an ENUM or of the engineer list used for grouping will initially not be presented, thus not changing the initial display behavior for either case. Only active values or engineers will be shown in a column each. However, a new selector for the grouping values will be present directly above the grid view. It will list the disabled entries as well, for distinguishing in italics, only unselected: the checkbox next to the value is unchecked. All active values are selected with a check mark. When adding a column for a disabled value by checking its entry in the selector, an additional column for the disabled value will appear in the grid view showing the tickets with this value. This column will have its header in italics to indicate that this value is disabled. Tickets can be dragged from this

column to another one to change to an enabled value, but tickets from other columns cannot be dropped on this column, so that no disabled value can be assigned. The screenshot below shows a column with a deactivated engineer and its entry in the selector.



1.3.11 History optimization for localized labels (#631218)

The internal management of history entries for localized labels and their changes created numerous unneeded history entries with hardly any helpful information. The creation of these superfluous internal history entries is now suppressed so that the internal history management in this aspect is leaner now without sacrificing information. The existing history is pruned during the update to this version in order to remove unneeded entries and a task script is added to the system. This script, named `translations_cleanup`, can be used to also remove unneeded entries without object association. This case can occur when removing workflow elements with localized labels, for example. The script should be executed as task in the Task Execution Framework and will remove existing orphaned history entries of localized labels from the tables `cmas_localized_value_history` and `cmas_localized_value`.

1.3.12 Representation notification enhancement (#624305)

The behavior of **ConSol CM** when setting up engineer representations has been changed to be more explicit about the actions taken. Previously there was no information for either the representing or the represented engineer when a representation was configured.

Now both engineers will receive a notification email when a new representation is introduced or removed. The content of these emails is defined in a different **Admin Tool** template for each of these cases. Both of the two new templates have pre-defined names:

- `representation-create-email`: Template name for the email when a representation is set up.
- `representation-delete-email`: Template name for the email when a representation is removed again.

Both are added during setup and update. These templates introduced via this way also show examples how to reference the represented and representing engineers. For these templates there are two special engineer objects available named `represented` and `representing` which can be used for creating the message:

```
{representing.name} is representing ${represented.name}
```

A representation of both types can only be created, if both engineers share at least one common role. This was not strictly enforced previously for adding a representation where the engineer who is logged in selected another one to represent. This condition is checked and enforced for both types now when creating the representation. It is not re-evaluated until deleting the representation which implies that it will have no effect on the representation, if the shared role is taken away from either one of the engineers during the representation.

1.3.13 Copy/paste of fields in *Admin Tool* ignores annotation *contains contact* (#630820)

The functionality to copy and paste fields in the *Admin Tool* introduced in *ConSol CM* release 6.11.0.1 also copied the annotation `contains contact` with its value. In fact this is undesired since the annotation may be present only once in a customer data model. Copying and thus duplicating it inside a customer data model would render the model invalid. The same holds true for another data model which already has one field with this annotation. Only when pasting it to another customer data model without any field annotated this way would yield a benefit. Pasting this annotation together with its field to a ticket or resource field would turn the annotation irrelevant. Therefore, the copy/paste behavior has been modified to not copy this annotation at all. Thus, the potential pitfalls are avoided and the enforced manual control over adding this annotation ensures to only use it when properly justified.

1.3.14 Clarification of information messages when deleting customers (#627639)

The messages informing the engineer about the implications on existing data when deleting customers could cause misunderstandings, especially when interacting with limitations imposed from the engineer permissions on the affected data. The phrasing of these messages has been modified to provide clear information without communicating facts which may not be shown to the engineer.

The message when choosing another customer who should be the target for the data transferred from the contact to be deleted will now show:

- the number of open tickets and resources to be transferred to which the engineer has sufficient access privileges (can be zero)
- an additional statement informing about the existence of other objects of the contact to which the engineer has no sufficient access privileges, if there are any

The open tickets and resources not accessible to the engineer will only be transferred, if the customer will be deleted. In case only a transfer is made, those objects remain associated to the customer, and a proper message will be displayed that some data have not been transferred due to access restrictions.

1.3.15 REST API returning FAQ tickets streamlined (#630997)

The way the REST API returns tickets from FAQ queues for customers has been streamlined internally. The changes implement the expected behavior that, when querying the REST API for FAQ tickets without additional restrictions, all tickets from the FAQ queues to which the engineer assigned to the customer has access. This means tickets from FAQ queues and having a customer both of which must be accessible with the engineer privileges. Since FAQ information is generic to some degree the irrelevant customer information for the tickets is removed from the response. In the context of this change the security handling for all ticket queries using the REST API has been unified more generally since there was some minor variation previously.

1.3.16 Resource relation output ETL plugin enhancement (#630633)

The ETL plugin for the *Resource Relation Output Step* required to select one specific resource relation when configured. This design made it impossible to re-use a transformation for other relations. In consequence it may have been necessary to design numerous identical transformations which only differed in the resource relation for which they imported data. This shortcoming has been addressed in this release.

Starting with this release two configuration patterns exist in parallel and it is up to the transformation designer to choose one:

- Define one specific resource relation used in this transformation like before. Or
- Provide a table with relation names used in the import file mapped to actual relations in the system.

This enhancement reflects in a changed dialog when editing this type of step.

ConSol*CM resource relation output

Resource Relation Import

Step name: ConSol*CM resource relation output

Batch size: 1

Update: ☐

Comment input field: Comment

Endpoints input fields: Resource id, Ticket name

Definition input field: RelationName

#	Definition input value	Definition
1	Manuals	manualSetCase manualSet to tickets in queues CM6_CaseManagement
2	ReleaseNotes	releaseNotesCase releaseNotes to tickets in queues CM6_CaseManagement

Dropdown menu options:

- manualSetCase manualSet to tickets in queues CM6_CaseManagement
- releaseAssignedTasks core_release to tickets in queues CM6 Bugfix Development, CM_coordination
- releaseNotesCase releaseNotes to tickets in queues CM6_CaseManagement
- systemRequirementsCase systemRequirements to tickets in queues CM6_CaseManagement

Buttons: Get definitions, Get fields, OK, Cancel

The *Definition* drop-down selection to identify the relation has been removed, but below the *Endpoints input fields* definition new configuration data can be entered. The *Definition input field* is the name for the field in the input data file which holds an identifying string for the relation to be selected in the import. All the values used in this field could be mapped in the table below. For each value entered in the first column *Definition input value* a specific relation from the system must be selected

in the second column *Definition*. The drop-down selector in this column lists the resource relations with their technical names, the resource name, and the name of the related object, for tickets including the queue.

The configuration change described here allows this way to define the relation to use in the import data. It still is possible to have a fixed selection of the relation to be used. This can be configured by leaving the values for *Definition input value* empty. This would match any value in the input file column and always choose this relation.

It also is possible to have only one row in the matching table without a *Definition input value*. In this case it matches all values not specified in other rows of the matching table, providing a default relation.

The previous behavior can be fully restored by leaving both the *Definition input field* and the *Definition input value* empty, but selecting one relation in the first line of the *Definition* column. With the field name and content unspecified the input file is not analyzed for a column identifying the relation, and the selected default relation will be used for all records. Therefore use of the same input data file from the transformation for an earlier **ConSol CM** release can be continued and will yield the same results.

1.3.17 Customer data model switch restriction added (#632133)

In previous versions of **ConSol CM** the customer data model for a customer group could be switched in the **Admin Tool** to another model, even if customer data records had been created in this customer group. However, actually switching the model resulted in exceptions in this case. To avoid this kind of problem a new restriction has been added. It prohibits altering the data model used for a customer group when records are existing already. So now a switch as described above is not possible anymore.

1.3.18 *Process Designer* script validation disabled when offline (#631149)

The script validation in **Process Designer** showed errors previously when the connection to the server was lost during editing of workflow scripts. This could be due to network connectivity issues or a server shutdown, for example. These errors will not be shown anymore, but rather the online script validation will be turned off automatically as soon as the connection to the server is lost.

1.3.19 Activity availability in workflow scopes with triggers and scope activities (#631072)

The list of activities offered to the user in a special workflow setting did not match the expectations. The ticket needed to be in a workflow scope that had a trigger with a subsequent automatic activity set up as a so-called interrupt. This scope also needed to have a scope activity introduced in **ConSol CM** release 6.11.0.0. In case the interrupt was triggered the list of activities offered to the user was limited to the scope activity only. Even though this may be correct from a highly technical standpoint, this did not show the expected list. The user does expect the same list of manual and scope activities as before the trigger was executed. This is exactly the list which will be provided now after changing the way how to handle this workflow situation.

1.3.20 Class imports for scripts unified (#630718)

The implicit imports of API classes in scripts differed in previous versions of **ConSol CM** for scripts managed in the **Admin Tool** and those managed in the **Process Designer**. These subtle differences were present in the scripts saved for execution in both tools as well as in the validation of the scripts

while editing them. These variations between the tools and handling have been eliminated and now a uniform list of classes is imported in any script for execution and validation in the **Admin Tool** and the **Process Designer**.

 Please refer to the manuals of **ConSol CM** version 6.11.1.0 for the detailed list of classes!

In conjunction with unifying the class imports the script validation has been slightly changed. Since the type validation of method parameters sometimes wrongly reported type mismatches, now only the number of parameters for a method is validated, but not the types.

1.3.21 Identification of automatic activities for data warehouse reporting extended (#625781)

The identification of automatic activities from triggers for reporting previously took automatic escalations, mail and time triggers into account. Business event triggers and automatic workflow changes were not considered. This has been unified so that all these automatic events qualify as automatic activities for reporting from the data warehouse. In consequence this extension makes the reporting on this flag more transparent.

1.3.22 ENUM changing operations written directly to the database (#630330)

The API methods which change (or add) ENUM content previously did rely on the database access layer to write the changes to the database. In some scripts situations could arise when this caused errors, because the database write operation was delayed and a second operation caused a conflict. Therefore for all write operations on all objects of ENUM structures the database write operation is explicitly performed now without being delayed. Thus, no conflicts from delaying operations on the database layer can arise anymore. This change affects the following methods:

Class	Method
EnumService	createGroup()
EnumService	createType()
EnumService	createValue()
EnumService	deleteGroup()
EnumService	deleteType()
EnumService	deleteValue()
EnumService	updateGroup()
EnumService	updateType()
EnumService	updateValue()

1.3.23 Trimming long file names of email attachments (#625239)

An attachment in an incoming email could cause an exception and fail to be imported into **ConSol CM**, if the file name was overly long. This would still happen even when trying to re-import it in the **Admin Tool** after the initial failure. The way to deal with overly long attachment file names has been improved, so that now the name is shortened to 255 characters, if longer than that, and thus, the former exception will not occur in either context.

1.3.24 WorkflowEngineStatus MBean output extended (#631175)

The server MBean **WorkflowEngineStatus** located in *consol.cmas -> admin -> global* has been extended regarding the output of the operation **showStatus()**. A new line showing the timestamp of the generation of this report has been added. The intention of this addition is to allow easier identification of the time the report was generated, relating it to the server operation and putting it into sequence, if desired. An example request with its new, extended return value is shown below. Please note that the return value has been heavily reformatted for better legibility!

```
curl --user jolokiauser:password http://cm-server:8080/jolokia/exec/consol.cmas:type=admin,topic=global,name=workflowEngineStatus/showStatus/

{
  "timestamp": 1505458745,
  "status"    : 200,
  "request"   : {
    "operation": "showStatus",
    "mbean"
: "consol.cmas:name=workflowEngineStatus,topic=global,type=admin",
    "type"      : "exec"},
  "value"     : "<table>
    <tr><td>Report timestamp</td><td>Fri Sep 15 08:59:05 CEST
2017</td></tr>
    <tr><td>initial idle interval (seconds)</td><td>5</td></tr>
    <tr><td>additional idle interval (job-executor-
1)</td><td>0</td></tr>
    <tr><td>locking limit</td><td>5</td></tr>
    <tr><td>lock timeout</td><td>60</td></tr>
    <tr><td>max retries</td><td>5</td></tr>
    <tr><td>number of threads</td><td>1</td></tr>
    <tr><td>retry interval (seconds)</td><td>30</td></tr>
    <tr><td>tx timeout (seconds)</td><td>60</td></tr>
    <tr><td>currently active jobs</td><td>0</td></tr>
    <tr><td>currently active and locked jobs</td><td>0</td></tr>
    <tr><td>number of jobs under retry</td><td>0</td></tr>
    <tr><td>number of jobs delayed</td><td>0</td></tr>
    <tr><td>errors counter (job-executor-1)</td><td>0</td></tr>
    <tr><td>timestamp of last executor hit</td><td>{job-executor-
1=Fri Sep 15 08:59:04 CEST 2017}</td></tr>
    </table>"
}
```


1.3.25 *JBoss* cache item life span configurable (#630565)


The data elements kept in the *JBoss* application server internal cache had an infinite life span by default previously. There was no mechanism to modify this setting from inside *ConSol CM*, it had to be done by editing *JBoss* configuration files. This default configuration could result in performance issues on large installations. Increasing CPU load was needed for re-organisation of an ever-growing cache.

The potential issues are addressed in this release by generally limiting the life span of the objects in the application server cache. This life span can be configured by a system property which defaults to the interval of one day. The property `eviction.lifeSpan` is introduced in the module `cmas-core-cache` setting the interval after which an object will be invalidated and removed from the cache. Its value is defined in milliseconds and defaults to "86400000" representing one day.

It has proven to resolve performance issues resulting from an oversized cache in larger installations when reducing this value to "1800000" (half an hour), for example. The optimal setting for a particular system may be different from this or the default value.

1.3.26 Explicit deployment timeout setting for *JBoss EAP* (#630754)

The *JBoss* application server product introduced a timeout property for deployments in order to deal with container stability in the EAP versions 6.3 and 6.4. This property `jboss.as.management.blocking.timeout` defaults to 300 seconds which means that after this time the application server is shut down, if the deployment has not completed. The intention is to have it as a safety mechanism in installations with many deployments. In reality this setting interferes with *ConSol CM* deployments, for example when updating the system. Especially in the case of updating to the newer 6.11 versions without pre-migration of the data this caused updates to fail. Therefore, the property has been added to the standard configuration files with a value of 100 hours ("360000"). Depending on the deployment scenario this value can be lowered by modifying it in `cm6.xml`, for example, as long as it is made sure that the new setting does not interfere with operation and updates of *ConSol CM*. Since it is new to the configuration file `cm6.xml` and its counterparts it applies only to new installations and should be introduced manually to existing installations where reasonable.

 Please note that this change originally has been introduced in *ConSol CM* version 6.10.6.0 and is referenced here for completeness only!

1.3.27 *JBoss* setting for HTTP parameter limit provided (#631199)

A new setting for configuring an aspect of the *JBoss* application server has been added to the *ConSol CM* configuration. It allows to explicitly define the maximum of parameters for a request when submitting data in the *Web Client*. The value defaults to "512" which previously was the inaccessible implicit value as well. There are cases when this has proven to be too low, for example, when editing a list of STRUCTs with many fields and many rows.

This value can be increased now by changing the value for the following line in the configuration file `cm6.xml` (or the respective file used in a specific installation):

```
<property name="org.apache.tomcat.util.http.Parameters.MAX_COUNT" value="512"/>
```

The line can be found in the section `<system-properties>` of the configuration file.

1.3.28 Node clock offset tolerance increased for index synchronization in a cluster (#630525)

The index synchronization procedure with several cluster nodes could previously miss some data changes, if the clocks on the nodes were not well-synchronized. This could not happen in any other case. In order to address this uncommon case so that index synchronization will not miss to index data an increased tolerance for unsynchronized clocks on the nodes has been provided. Now an offset up to ten minutes is tolerated between the clocks, independent of the fact that such an offset should be eliminated by the system administrators.

1.3.29 Dedicated category labels for resource operations in operationtimes.log (#630787)

The special log file `operationtimes.log` for analyzing performance bottlenecks in the interaction with the system did not feature specific category labels for resource operations in earlier releases. All these operations were labeled “UNKNOWN”, but could clearly be identified by the URLs in the log line. In order to provide greater ease of categorizing the resource operations in this log file, three new category labels have been added:

- “RESOURCE_LOAD”: Operations in the context of loading resources.
- “RESOURCE_UPDATE”: Operation in the context of updating a resource.
- “RESOURCE_DASHBOARD”: Operations in the context of the resource dashboard.

1.3.30 Log file for the deletion of customers reintroduced (#631158)

In **ConSol CM** version 6.10.5.4 a separate log file recording the deletion of customers was introduced. The file default name is `unit-deletion.log` and should be the only location to log the deletion of customers. This change was not included in the 6.11.0 releases by mistake. The current version 6.11.1.0 includes this functionality again fully. It can be used and configured in the original manner now.

1.3.31 REST API logging default setting and content changed (#621191)

The default logging configuration for the REST API has been modified. Earlier the logging for usage of the REST API was disabled. This has been changed to logging activated on the **INFO** level.



Please be aware that this configuration change will only be applied to new installations. For existing installations using log4J, the logging of REST API usage must be activated manually.

In order to activate this logging on an existing installation the following entry in the configuration file `cm6.xml`, `cm6-cmr-f.xml` (or the corresponding file in the specific installation) must be added, changed, or uncommented:

```
<logger category="com.consol.cmas.restapi">
  <level name="DEBUG"/>
</logger>
```

Please use a specific file handler for the logger, if desired.

The log messages on the **INFO** level have been extended to now include successful and failed authentication attempts also. More detailed logging of the operations performed from REST API requests can be activated when switching to **DEBUG** level.

1.3.32 Scene export error logging improvement (#623575)

The exact object causing an export to fail with an error, for example for non-XML characters, previously could not be identified easily from the log message of the error. Therefore an in-depth analysis always was required on such an export failure before the data issue could be addressed.

This process has been made easier by extending the log with detailed information about the object causing the export failure, so that the analysis to identify the problematic data is provided in the log files. Depending on the object type this may include IDs, descriptions, relevant transfer keys, and many others useful to identify the affected database rows. The identified problem still has to be resolved manually, possibly in the database directly, depending on the specific issue.

1.3.33 Irrelevant log messages from server start filtered out (#629645)

During the server start numerous messages from the warmup phase during server start which were almost identical were listed in the log files. These messages regarding internal start and end tags were irrelevant for all means, so they are filtered out now. Thus, this type of message effectively has been eliminated from the log files.

1.3.34 Unused system properties removed (#631640)

There were a few system properties which had become obsolete due to earlier changes which were not yet removed from the system for different reasons. These properties have been eliminated in this release. See below for a list of the removed properties.



Please verify that there is no false expectation that these system properties and their settings influence your system behavior!

Module	Property
cmas-core-cache	communication.mode
cmas-core-index-common	database.notification.max.attempts
cmas-core-security	resetCode.expirationPeriod
cmas-core-server	ldap.certificate.attribute
	ticket.list.limit
cmas-dwh-server	notification.error.host
	notification.error.password
	notification.error.port
	notification.error.user
cmas-esb-mail	mail.redelivery.retry.count

Module	Property
cmas-setup-jcr	dataStore.class
	externalBLOBs
	journal.class
cmas-workflow-jbpm	jobExecutor.jobExecuteRetryNumber
	jobExecutor.timerRetryInterval
	mail.sender.address
core-content-jcr	perform-update

1.3.35 Mail classes added to the API documentation package (#631310)

The archive file with the scripting related API documentation has been enhanced by a packet and now also documents classes in the context of the New Incoming Mail Handler. For example, it now includes the classes `MailHolder`, `MailSupportService`, and `MailPipeContext`.

This archive file is offered on demand for each **ConSol CM** release for local usage on computers used for script development.

1.4 Bugs fixed

The following defects have been corrected in this **ConSol CM** version.

Number	Description
622412	Deleted workflow information erroneously re-used Once an existing workflow had been deleted in the Process Designer and later a new one with the same name has been created, this new workflow wrongly was flagged as deleted. It also had the next higher version number from the deleted one assigned. This erroneous behavior has been fixed and the new workflow is completely handled as a new one without the deleted flag and a new version counter.
624637	Customer field group order not imported In a customer data model which consisted of several field groups a different field group order was not updated from an imported scene, but the current order was retained. This has been changed so the order of field groups will be updated from the scene during an import now, if it differs from the current configuration.
624995	Apostrophe in ticket field localization prevented usage in templates An apostrophe in the localized name of a ticket field made it impossible to use this field in text templates. The field could not be chosen for a template then. This limitation has been removed and such data fields can be chosen without problem now.
625371	Missing JGroups reference added in WebLogic package In the ConSol CM package for usage on Oracle WebLogic application servers a correct reference to the JGroups library provided was missing. This could cause the server startup to fail, if the application server did provide a different version globally, for example for another application deployed to it. The reference has been added so that the correct internal library version is used and the server startup will not fail for this reason anymore.
626156	Web Client session not invalidated after different login The user session in the Web Client was still available and partially used after a user left the system with the back button on the browser and after that a different login was entered to access the system. This unwanted and mixed behavior has been corrected so that in such a scenario no previous session is used or available.
627759	Date format tooltip used wrong locale/abbreviation The tooltip for date fields showing the acceptable date format showed <i>yy</i> instead of <i>jj</i> as abbreviation for the year in all German browser locales. This has been adjusted to display the correct abbreviation.

Number	Description
629672	<p>Dependent read-only field issue</p> <p>Dependent read-only fields showed a small issue when editing on the create ticket page. In this case the read-only configuration did not apply and they were editable while on the create customer page they were read-only as desired. It was connected with correctly identifying the customer field value for which to apply the read-only property and has been resolved. Thus, the fields will be read-only on both pages now.</p>
630292	<p>Switching ticket history filter while writing a new entry defect</p> <p>When editing a new comment or email after replying to another email and then switching the history entry filter from <i>Communication only</i> to <i>All entries</i> the new filter setting was not applied. This defect has been corrected and the changed filter will result in a changed history display now.</p>
630360	<p>Resources transfer from one customer to another had deficits</p> <p>When transferring data from one customer to another resources related to this contact were not transferred to the other customer chosen. The message informing about the objects to transfer also did not include any information about resources. This has been changed so that the message informs about resources to transfer and the related resources will be transferred to the other customer as well.</p>
630624	<p>Wrong script displayed for editing in <i>Admin Tool</i> editor</p> <p>After filtering the script list offered in the <i>Admin Tool</i> and then creating a new script which does not match the current filter, the script offered in the script editor for modification was the last one selected after filtering and not the newly created script. This highly undesired presentation has been corrected so that now the filter is reset and the new script will be offered, even if it does not match the filter. The same change has been provided for the copying scripts and <i>Admin Tool</i> templates.</p>
630626	<p>Tooltip does not adjust to selection of history filter</p> <p>The tooltip text shown for the filter of the history entries used to restrict the display to communication entries did not change on changing the selection. After switching from "Communication only" to "All entries" the tooltip text was still explaining the former selection. This has been fixed so that after the switch the correct description is shown as a tooltip.</p>
630698	<p>Scene export of only latest workflow versions results in irrelevant warning messages on import</p> <p>When choosing the option to export only the latest version of the workflows in a scene export multiple irrelevant warning messages were issued and displayed on the import of this scene ("... control forms ignored: missing parent entities."). The scene export/import of workflows has been modified to not yield these warnings anymore.</p>
630767	<p>License count for user and role administrators inaccurate</p> <p>The used license count shown on the <i>Admin Tool</i> start panel always listed zero consumed licenses out of zero available ones for administrators which only had the privilege to configure users and roles. This wrong information has been altered to show the correct values of used and available licenses now.</p>

Number	Description
630772	<p>Quoted email address in descriptive address part caused error and failure to send the email as reply</p> <p>In case an incoming email contained an email address with a descriptive section containing the address itself in quotes, replying with <i>Reply to:</i> failed with an error. It was not possible to have this address in the CC: field and send the reply email. The issue has been addressed and this kind of address will not make a reply fail anymore.</p>
631020	<p>Context menu click not executing command</p> <p>In recent releases a click in any context menu did not execute the command on the line of the click, if the click was not above the command label text but on an empty part of the line. This just closed the context menu which is undesired. This click area restriction has been removed and a click anywhere in the line of the command will execute it now.</p>
631057	<p>Missing German translation of transfer message</p> <p>The message informing the user about the objects to transfer when initiating such a transfer from a customer to another had no German translation and therefore the English message was shown in the German locale. This translated message has been supplied so that the English message is not shown anymore for the German locale.</p>
631078	<p>Scripted autocomplete fields error on company edit page</p> <p>When using a scripted autocomplete field for companies accessing it on the company edit page was not possible. It showed a busy indicator permanently and produced an error in the logs. This problem has been resolved and scripted autocomplete fields work in this context as well now.</p>
631210	<p>Image re-use issues in <i>Web Client</i></p> <p>When re-using one or more images from a history entry in a new email or comment, images partly were not correctly present in all cases. This could happen both when using copy and paste as well as when using drag-and-drop. In some cases the re-used inserted images could not be seen in an email client, in others they were not available in the <i>Web Client</i> history. Occasionally when operating on several images, some were replaced by another one from the set. These image re-use issues have been corrected except for one border case to be resolved later. This border case is listed in the Known issues section under the issue ID 631731.</p>
631227	<p>MLA value set during ticket creation not shown in history</p> <p>The value of an MLA field chosen while creating a new ticket with this field was not represented in the ticket history like any other field value. Changing the value when later editing the ticket was shown as expected. This omission in the history has been fixed so that the initial value setting of an MLA field during ticket creation will also show in the ticket history.</p>
631254	<p>Changing a script name suffix not immediately recognized</p> <p>When changing the suffix of a script name the old script was internally not immediately replaced by the new renamed one. This could cause undesired script results and has been corrected now.</p>

Number	Description
631273	Last row of address autocomplete table not offered in <i>Web Client</i> When using an address autocompletion in the <i>Web Client</i> the last row (after ordering) coming from the corresponding database table could not be found in the auto-complete field. This omission was addressed and the missing entry will be supplied now.
631279	Replying to email address with one-letter mailbox name prohibited When trying to reply to an email containing an email address which had a mailbox name before the “@” character made up of one letter only, like “a@mail.com”, the email could not be sent due to a wrong validation error claiming an incorrect address. These perfectly valid addresses are allowed in replies now.
631291	Doubled confirmation dialogs and messages when deleting resource with relation(s) When deleting a resource with one or more existing relations a confirmation dialog was shown two times, both of which also returned a success or error message. This unwanted duplication has been removed so that only one dialog and message will be shown now.
631295	Wrong method name returned in exception message The method <code>getStringFieldValue()</code> from the class <code>MailContactCriteria</code> did not return its own method name, but the wrong <code>getStringGroupValue()</code> in the case of an exception. This was corrected so that now the correct own method name is shown in an exception.
631366	Setup failure when reinitializing system on existing database The system setup failed when reinitializing the system with deleting all data on a pre-existing database filled with data. An internal database table was not properly removed in this uncommon case. This problem has been addressed and reinitialization now works in this case as well.
631547	Wrong message after transferring data instead of deleting contact When starting to delete a customer, but selecting to transfer the data from the context menu instead of selecting a target customer in the delete and transfer form, the message after completing the operation wrongly stated that the customer was deleted. The data had only been transferred as desired, thus the system behavior was correct, but the message was incorrect. This message has been switched to a correct one informing about the real operation performed.
631649	Long names abbreviated differently in workspace and favorites Long names for ticket subjects, customers, and resources were abbreviated differently when added to the workspace and favorites list. In the favorites the label string for all these was trimmed and appended with dots for the ellipsis while in the workspace the string was just cut off after the last word fitting in the available space. Now both lists uniformly show the dotted strings as previously only the favorites.

Number	Description
631814	Deleting a resource while another user edited it caused error and logout In case an engineer was deleting a resource while another engineer was editing the same resource and wanted to access the Quick Search this engineer editing the resource was logged out and left with a blank browser page. This error has been corrected and the engineer now gets the message that the resource to be edited was deleted.

2. Version 6.11.1.1 (31.01.2018)

Version 6.11.1.1 includes all **ConSol CM** changes and additions of

- 6.11.0 versions up to 6.11.0.6
- 6.10 versions up to 6.10.8.0
- 6.9 versions up to 6.9.4.7
- 6.8 versions up to 6.8.5.8

Newer versions and the respective changes are not included in this **ConSol CM** version.

2.1 Update and installation instructions

Please read all information on update and installation of **ConSol CM** specific for version 6.11.1.1 in this section before performing a system update or installation. Please consult the general product documentation on update and installation procedures as well as earlier version Release Notes, if applicable.

2.1.1 System upgrade from 6.11.0 and earlier versions

This **ConSol CM** release needs specific planning for an update from earlier releases. Please check the instructions for release 6.11.1.0 ([System upgrade from 6.11.0 and earlier versions](#)).

2.1.2 REST API client modification requirements

The following REST API enhancements may require to adjust clients accessing **ConSol CM** by using the REST API. The changes might offer a potential for client optimization.

- [New handling of scripted autocomplete fields \(#632555\)](#)

2.1.3 Script modification requirements

This release features some interface changes in API methods. **Please be aware that scripts using the respective methods may need adjustments!** The affected methods are listed in the following sections of this document:

- [New handling of scripted autocomplete fields \(#632555\)](#)
- [Support for HTTP status / error codes for the Webhook \(#632695\)](#)
- [Action type available in relation action scripts \(#632812\)](#)
- [Class SecurityHelper deprecated \(#632911, #632937\)](#)

2.1.4 Configuration modification requirements

The changes in this release require several configuration items to be reviewed and possibly adjusted. Please see the following sections for details:

- [Localized title for Recent changes and Recently visited widgets \(#632405\)](#)
- [Synchronized XML configuration files \(#632410\)](#)
- [New caching mechanism enabled \(#632523\)](#)
- [New handling of scripted autocomplete fields \(#632555\)](#)
- [Support for annotation list-type \(#632699\)](#)
- [Default value of the property unit.description.mode updated \(#632882\)](#)

No further instructions available.

2.2 New features

The features newly introduced in this **ConSol CM** version 6.11.1.1 are described in detail in this section.

2.2.1 New handling of scripted autocomplete fields (#632555)

The support for scripted autocomplete fields has been improved. This new feature introduces a new way of handling scripted autocomplete fields in the **Web Client** and **CM/Track V2**.

A new data field type *autocomplete* has been introduced in the **Admin Tool** for tickets, customers, and resources. When creating a new autocomplete field, the user needs to select the type of object and the script used to retrieve the objects. The type of object can be *Ticket*, *Unit*, *Resource*, or *Engineer* for the respective objects or *String* for a string which does not reference any object. The script has to be saved in the **Admin Tool** with script type *Text autocomplete*.

The screenshot shows a 'Create field' dialog box. Inside, there's a 'Field details' section. The 'Name' field contains 'engineer_autocomplete'. The 'Data type' dropdown is set to 'autocomplete'. Below this, there's a message: 'Please choose type and script for autocomplete field'. Under this message, there are two more dropdowns: 'Type' and 'Script'. Arrows from the text in the paragraph point to the 'Data type' and 'Type' dropdowns.

Two values are saved for autocomplete fields:

- **Internal value:** For fields of type *Ticket*, *Unit*, *Resource*, or *Engineer*, this is the reference to the respective object. For fields of type *String*, this is the string itself.
- **Display value:** This is the value shown to the user in the **Web Client** or **CM/Track V2**. The display value can be configured to be either retrieved from the referenced object or as a static label. For fields of type *String*, it is always a static label.

The type of display value is determined in the script which provides the logic for the autocomplete list. It depends on how the results are added to the result set (object of type `ScriptAutocompleteResult`) in the method `onSearchInput`.

To define a static label, i.e., the display value is not updated when there are changes to the referenced object, add a value from the referenced object or any string to the result set. For example:

- `resultset.add(ticket, ticket.getSubject());`
- `resultset.add(ticket, "some description");`

To define a dynamic label, i.e., the display value is retrieved from the referenced object and is updated in case of relevant changes in the referenced object, add the object to the result set. For example:

- `resultset.add(ticket);`

For dynamic labels, the following information is displayed both in the field and in the history.

- engineer -> engineer description (first and last name, if available, otherwise the login)
- ticket -> ticket name and subject
- resource -> resource description taken from the template
- unit -> unit description taken from the template

The value of an autocomplete field can be accessed later using the new class `AutocompleteValue`:

```
AutocompleteValue myAutocompleteValue = <object>.get("<field  
group>", "<field>");  
myAutocompleteValue.getInternalValue();  
myAutocompleteValue.getDisplayValue();
```

It is also possible to set the values of an autocomplete field using the API. The following example sets a dynamic label. To set a static label, omit the second line of the code example.

```
AutocompleteValue myAutocompleteValue = new AutocompleteValue("<internal  
value>", "<display value>");  
myAutocompleteValue.setDynamicDisplayValue(true);  
<object>.set("<field group>.<field>", myAutocompleteValue);
```

Permissions are checked during the execution of the autocomplete script when a field is edited in the **Web Client** and **CM/Track V2**, i.e., the user can only see and select entries referring to objects which he is allowed to view according to his assigned roles. When viewing autocomplete fields, permissions are not checked. Therefore, the user might see labels referring to objects for which he has no permissions in autocomplete fields and history entries.



Note regarding permissions in *CM/Track V2*:

Permissions related to scripted autocomplete fields work the same way as for the **Web Client**. *CM/Track V2*-specific restrictions, namely the limitation that a user only sees his own tickets (and tickets of other contacts of his company, if configured) and contacts of his company, are not part of the default configuration. This means that, by default, the user has permissions to see and select entries referring to tickets belonging to queues for which the Track user has at least read permissions and customers belonging to customer groups for which the Track user has at least read permissions. In short, the customer would see data belonging to other customers.

If this is not desired, the autocomplete script must be adapted using methods of the new class `customerSecurityCriteriaBuilder`, see section [Scripting examples](#) below for more details.

The class `customerSecurityCriteriaBuilder` has been added to the **ConSol CM** API. It provides the following methods to control permissions for autocomplete results in *CM/Track V2*:

Method Summary	
Methods	
Modifier and Type	Method and Description
boolean	<code>setCompanyCaseCriteria(TicketCriteria pCriteria)</code> Set additional criteria for case COMPANY: ticket has customer company or its member assigned
void	<code>setCurrentCompanyCriteria(UnitCriteria unitCriteria)</code> If the security.restrict.unit.access.to.own.data flag is set to true it will add criteria which limits unit access to the logged customers company and its members.
boolean	<code>setFaqCaseCriteria(TicketCriteria pCriteria, Set<Long> requestedQueues)</code> Set additional criteria for case FAQ: queue has faq nature (not readable ticket customer is removed).
void	<code>setMyCaseCriteria(TicketCriteria pCriteria)</code> Set additional criteria for case MY: ticket has customer assigned
void	<code>setOwnCaseCriteria(TicketCriteria pCriteria)</code> Set additional criteria for case OWN: ticket has customer assigned as main contact

The **REST API** has been enhanced to provide support for autocomplete fields. In the context of tickets, the list of suggestions can be retrieved using REST:

POST `/restapi/autocomplete/ticket`

Parameters:

- *ticketName*: name of the ticket
- *field*: name of the ticket field
- *group*: name of the ticket field group
- *queue*: name of the queue
- *pattern*: search pattern

It is not possible to retrieve suggestions for customers or resources.

For all objects (tickets, customers, and resources), the internal and display value of autocomplete fields can be retrieved.

The values can be updated using REST with the following commands:

PUT `/restapi/tickets|units|resources/id`

Parameters:

- `<field group>.<field>.internalValue`: internal value, object ID of the ticket, customer, resource, or engineer or string
- `<field group>.<field>.displayValue`: display value, can be a static string or be generated from a template
- `<field group>.<field>.dynamic`: *true* or *false*, determines if the display value should be generated from a template (*true*) or a be a static string (*false*)

The following limitations exist when using the new way of configuring scripted autocomplete fields:

- No support for **CM/Track V1**. Scripted autocomplete fields are not supported in **CM/Track V1**.
- No transfer to the DWH. Scripted autocomplete fields are not transferred to the DWH and cannot be used in reports.
- No support for ETL. Import and export of scripted autocomplete fields using ETL is not supported.
- Scripted autocomplete fields cannot be indexed. Therefore they are not available for search.
- If an object which is referenced in an autocomplete field is deleted, the field might point to a non-existing object. This has to be considered in case the field value is later used in scripts.
- Scripted autocomplete fields cannot be used in text templates and document templates.

This new way of configuring autocomplete fields replaces the previous method, which used the annotation `text-type = "autocomplete"`. Existing fields which use the old method will remain functional as of version 6.11.1.1. These fields will not be modified during an update. The previously used method `onEditDisplayEntered` still works for existing fields. Nevertheless, it is not needed anymore and will be removed in a future version of **ConSol CM**.

Scripting examples

The following example is used in an autocomplete field with resource objects. The user needs to enter at least 3 characters to start the search. The script returns a list of resources matching the entered search string. As display values, dynamic labels are used. For this purpose, the `resource` objects are added to the result set.

```
ScriptAutocompleteResult onSearchInput(String pSearchStr, FieldKey pKey,
Context pContext) {
    def result = new ScriptAutocompleteResult()
    def criteria= new ResourceCriteria()
    if (pSearchStr.length() <= 3) {
        return new ScriptAutocompleteResult( "Please type more than 3 characters")
    } else {
        criteria.setPattern("*" +pSearchStr+"*")
        for (Resource resource:resourceService.getByCriteria(criteria)) {
            result.add(resource);
        } return result
    }
}
```

The following autocomplete script limits the search results of an autocomplete field used in **CM/Track V2** to tickets belonging to the user.

```
ScriptAutocompleteResult onSearchInput(String pSearchStr, FieldKey pKey,
Context pContext) {
    def result = new ScriptAutocompleteResult();
    def criteria= new TicketCriteria();
    criteria.setPattern(pSearchStr+ "*");
    customerSecurityCriteriaBuilder.setMyCaseCriteria(criteria);
    for (Ticket ticket : ticketService.getByCriteria(criteria)) {
        result.add(ticket);
    }
    return result;
}
```

The following autocomplete script limits the search results of an autocomplete field used in **CM/Track V2** to contacts which belong to the user's company.

```
ScriptAutocompleteResult onSearchInput(String pSearchStr, FieldKey pKey,
Context pContext) {
    def result = new ScriptAutocompleteResult()
    def criteria= new UnitCriteria()
    criteria.setPattern("."+pSearchStr+".")
    customerSecurityCriteriaBuilder.setCurrentCompanyCriteria(criteria)
    result.add(unitService.getByCriteria(criteria));
    return result;
}
```

2.3 Changes

This section illustrates all relevant software product changes for version 6.11.1.1 of **ConSol CM**.

2.3.1 Layout improvements

German wording in *Recent changes* and *Recently visited* widgets (#632486)

The German translation of the *Show all* link in the *Recent changes* and *Recently visited* widgets in the **Web Client** was inconsistent. This problem has been corrected and both widgets show the same wording (*Alle anzeigen*).

Header of the window to create a new calendar appointment (#632613)

The header of the screen to create a calendar appointment in the *Calendar* section of the **Web Client** has been corrected. It is now *New appointment* in English and *Neuen Termin erstellen* in German.

Width of the library of markers increased (#632681)

The width of the selection box *Library of markers* in the text template manager of the **Web Client** has been increased, so that as many levels as possible can be displayed without having to scroll.

Width of the template selector increased (#632682)

The width of the drop-down field used to add a template when writing a comment or an email in the **Web Client** has been increased, so that longer template titles can be displayed.

Flexible width of autocomplete enums (#632864)

The drop-down list shown when selecting a value of an ENUM field of the type *autocomplete* in the **Web Client** now has a flexible width depending on the length of the displayed values.

2.3.2 CM/Track V2 improvements

Availability of changes from **CM/Track V2** version 6.10.8.0 (#632393)

This release features all new features, changes, and corrections introduced in **CM/Track V2** version 6.10.8.0.

Unneeded horizontal rules removed (#632575)

When viewing a ticket in **CM/Track V2** on a small screen, additional horizontal rules were displayed in some places. They have been removed as they were not necessary.

Support for annotation *list-type* (#632699)

The annotation *list-type* which was previously available for the **Web Client** only, now also works for **CM/Track V2**. It allows the administrator to configure if the user can add or delete rows in fields of type STRUCT or LIST.

Unneeded stacktrace in **CM/Track V2** removed (#632434)

When a `TooGenericSearchCriteriaException` occurred in **CM/Track V2**, an unnecessary stacktrace was written to the log files. Now, only the name of the exception is logged when the user tries to open the ticket list or FAQ list with too many tickets in **CM/Track V2**.

2.3.3 Localized title for *Recent changes* and *Recently visited* widgets (#632405)

The *Recent changes* widget and the *Recently visited* widget have been improved.

It is now possible to localize the widget title in the page customization. The setting is done in the sub-scope of the respective widget, i.e., *recentChangesWidget* -> *welcomePage* -> `<widget_name>` or *recentlyVisitedWidget* -> *welcomePage* -> `<widget_name>`. You need to enter the localized values in the attribute *localization* and set the attribute *title* to reference it.

The following example sets a localized title for the *Recent changes* widget in German and English.

- Attribute *localization*:

```
de: {title:'Letzte Änderungen'}, en: {title:'Recent changes'}
```

- Attribute *title*:

```
_('title')
```

2.3.4 Synchronized XML configuration files (#632410)

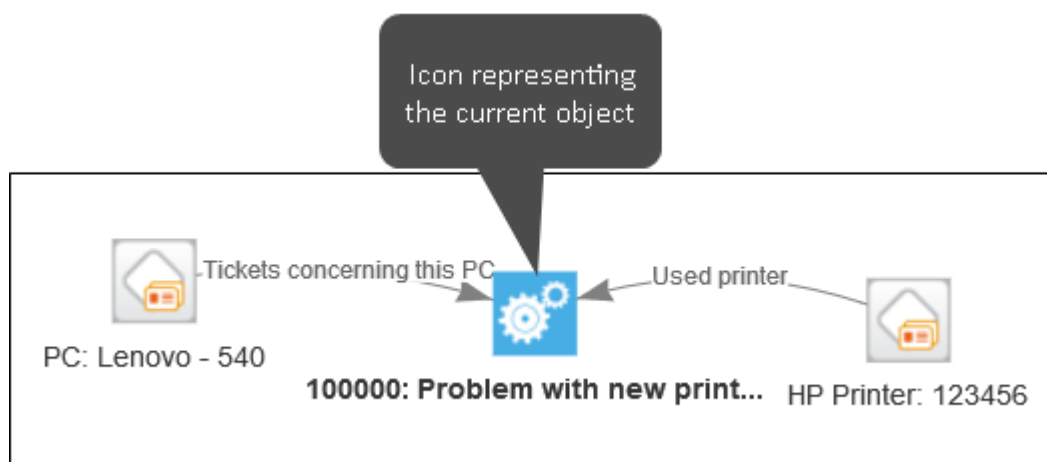
The default content of the files `cm6.xml`, `cm6-cmr-f.xml`, `domain.xml` has been synchronized for all databases. Some unnecessary differences between the files have been corrected.

2.3.5 Webhooks integration for WebLogic (#632412)

The Webhooks feature introduced in version 6.11.1.0 (see [Webhook integration interface \(#631699\)](#)) is now supported on WebLogic application servers.

2.3.6 Layout of the relation visualization graph improved (#632454)

The layout of the relation visualization graph introduced in version 6.11.1.0 (see [Relation visualization graphs \(#630556\)](#)) has been improved. The text of the icon which represents the current object, i.e., the root node, is now displayed in bold. In addition, the root node is shown in the center of the graph.



2.3.7 *Recent changes* widgets improved (#632466, #632471, #632472, #632549, #632550)

Several aspects of the *Recent changes* widget have been improved.

The page customization has been enhanced and it now lists which values can be used for the attribute *excludedHistoryTypes*.

Attribute name	Description
excludedHistoryTypes	Comma-separated history types that won't be displayed. Possible values: ACTIVITY_CHANGE, ATTACHMENT_DELETE, BOOKING, CONTACTROLE_SWAP, CONTACTROLE_UPDATE, CONTACT_ADD, CONTACT_REMOVE, CONTENT_ADD, ENGINEER_CHANGE, OPEN, QUEUE_CHANGE, REOPEN, RESOURCE_RELATION_CREATED, RESOURCE_RELATION_REMOVED, RESOURCE_RELATION_UPDATED, SUBJECT_CHANGE, TICKETRELATION_CREATE, TICKETRELATION_MOVE, TICKETRELATION_REMOVE, TICKETRELATION_UPDATE, TICKET_USER_ADDED, TICKET_USER_MOVED, TICKET_USER_REMOVED, UPDATE

The visibility selection (*All actions* or *User actions only*) is now saved in the user preferences.



If more than one *Recent changes* widget is used, each widget needs to have a unique name for this feature to work, e.g., `[rc1:RecentChanges]` and `[rc2:RecentChanges]`

Previously, the login name of the engineer who authored a change was displayed. The engineer names are now displayed according to the following rules:

- engineer has first name and last name: first name and last name are displayed
- engineer has only last name: last name is displayed
- engineer has only first name: first name and login are displayed
- engineer has only login: login is displayed.

The page customization *excludedQueues* now also excludes queues whose names contain a blank space.

The changes which belong to the same ticket are now grouped. Please see the following figure:

Recent Changes		⚙️ ▼ show all
<input checked="" type="checkbox"/>	100895 - ACF for search action - fields checker ticket	by Susan ServiceDesk
	• Start Point has been triggered, new Scope is No description	2 hours ago
	• Comment added	2 hours ago
	• Customer added	2 hours ago
	1 more changes	
<input checked="" type="checkbox"/>	349 - Please buy me a new workstation!	by Harald Huber
	• Ticket has been transferred to new activity Start Point	7 hours ago

2.3.8 Unneeded error log for non-existing unit (#632469)

When a user tried to access a non-existing unit, an error message with unnecessary stacktrace was written to the log files. This behavior has been corrected, and now a warning message without stacktrace is logged in this case.

2.3.9 New caching mechanism enabled (#632523)

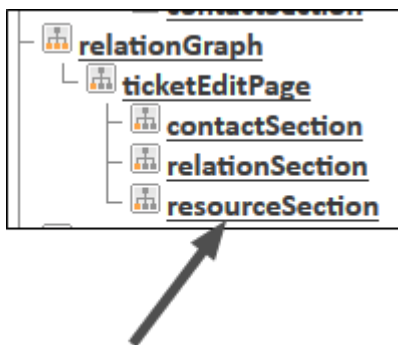
The **ConSol CM** system property `cmweb-server-adapter, voCacheEnabled`, introduced in version 6.11.1.0, is now set to “true” automatically during the setup or update of **ConSol CM**. This property enables additional caching, so called voCaching, of complete objects, thus improving performance.

Notes

- When voCaching is enabled and lazy loading is used for folding ticket history entries, once the engineer unfolded an entry, he cannot fold it again by reloading the page or opening the ticket from the workspace.
- When using the dynamic mode for displaying engineer and customer names in the ticket history (as configured in the system properties `cmas-core-server, engineer.description.mode` and `cmas-core-server, unit.description.mode`), the new version of the engineer and/or customer name is only displayed after the ticket has been changed.

2.3.10 Relation graph subscope renamed in page customization (#632569)

The subscope *relation* of the section *relationGraph* in the page customization has been renamed. It is now called *resourceSection* according to the content of the respective section. This applies to the ticket page, company page, and contact pages.



2.3.11 Support for list fields in control forms (#632667)

Fields of the data type LIST and STRUCT can now be used in control forms for search, customer, and resource actions. Previously, this usage was not possible and caused errors.

2.3.12 Support for HTTP status / error codes for the Webhook (#632695)

The Webhook feature introduced in version 6.11.1.0 now provides support for HTTP status and error codes. The following default HTTP status codes are supported:

- **200 (OK)**
- **401 (Unauthorized)**
One or more security provider is enabled but the data provided by the client did not match the expected data (e.g., wrong token or IP address)

- **404 (Not Found)**
A script for webhook with the given name could not be found / webhook is disabled / no security provider is enabled for webhook
- **405 (Method Not Allowed)**
Any HTTP method other than POST
- **500 (Internal Server Error)**
Any problems / exceptions on the server side (CM)

Additional status codes and HTTP headers can be defined in the **Admin Tool** script implementing the webhook (script type *Integration*). This is done by assigning the desired values to the response object.

The **Admin Tool** scripts of type *Integration* now return an object of the class `IntgServiceResponse`. The desired HTTP status code, HTTP header, and JSON body can be assigned to this object. The following variables are used:

- **statusCode**: defines the HTTP status code, optional, if not provided, only the default status codes are used
- **headers**: defines the response header, optional
- **body**: defines the content of the response, optional, if not provided, the response is empty

The script template in the **Admin Tool** has been adapted to reflect the new return value.



Existing integration scripts need to be adapted. Instead of JSON output, they need to return an object of the class `IntgServiceResponse`.

Scripting example

The following code example shows an integration script which retrieves the name and assigned engineer of a given ticket. Only the response body is set.

```

import com.consol.cmas.intg.service.IntgServiceResponse
import groovy.json.JsonSlurper
import groovy.json.JsonOutput

def jsonSlurper = new JsonSlurper()
def message = jsonSlurper.parseText(payload)
def ticketname = "n/a"
def ticketsubject = "n/a"
def engineername = "n/a"

if (message.ticket) {
    ticket = ticketService.getByName(message.ticket) ?: "n/a"
    if (ticket) {
        ticketname = message.ticket
        ticketsubject = ticket.getSubject()
        if (ticket.engineer) {
            engineername = ticket.engineer.getFirstname() + " " +
                ticket.engineer.getLastname()
        }
    }
}

//Please enter your script code below
def response = new IntgServiceResponse()
//response.httpStatusCode = 200 //optional
//response.httpHeaders = ['Content-Language':'en', 'Warning':'A general warning
    about ...'] //optional
response.body = JsonOutput.toJson([ticket: ticketname, subject: ticketsubject,
    engineer: engineername]) // optional
return response

```

2.3.13 Restore triggers after execution of scope activity (#632698)

When a ticket temporarily leaves its scope because of an interrupt which occurs when the user clicks a scope activity located in another scope, triggers from the original scope are now restored when the ticket reenters its original scope. Previously, triggers were deactivated when the ticket left its scope, but not restored when the ticket reentered it.

This new behavior applies only when the scope activity is executed manually. It does not apply if it is executed from a script or another activity.

2.3.14 Action type available in relation action scripts (#632812)

Scripts for customer and resource actions now provide a context parameter holding the type of relation operation. This parameter can be used for relation actions, i.e., automatic actions which are performed when a relation has been created or deleted. The parameter is called **actionType** and it can have two values: "ADD" and "REMOVE". It enables distinguishing between the creation and deletion of relations.

For example, it is now possible to implement a condition script which ensures that the corresponding action script is only executed when a relation has been added.

```
if(actionType=="ADD") {  
    return true;  
} else {  
    return false;  
}
```

Alternatively, such a condition can be implemented directly in the action script as well.

The templates for the customer and resource action and condition scripts have been adapted accordingly.

2.3.15 Data version updated (#632834)

The data version of the scene import and export files has been modified. It is now 37.

2.3.16 Default value of the property unit.description.mode updated (#632882)

The default value of the **ConSol CM** system property `cmas-core-server.unit.description.mode` is now "PROTOCOL" instead of "DYNAMIC". This is due to the introduction of voCaching (see [New caching mechanism enabled \(#632523\)](#)). If the dynamic mode is used and voCaching is enabled, the customer name is only updated in the ticket history after there was a change in the ticket.

2.3.17 Class SecurityHelper deprecated (#632911, #632937)

The class `SecurityHelper` is now deprecated. Its methods `canAccessCompanyTickets()` and `getCurrentContactId()` have been moved to `SecurityService`. Therefore, these two methods are now accessible as shown in the following example:

```
securityService.canAccessCompanyTickets();
```

2.4 Bugs fixed

The following defects have been corrected in this **ConSol CM** version.

Number	Description
632411	Caching issue when updating a list of structs When updating a customer on the ticket page, entered values in a list of structs were not shown in some cases. This problem has been solved, so that lists of structs work as expected now.
632457	Exception when accessing a recently visited item after server restart In some cases, an exception could occur when the user clicked a link in the <i>Recently visited</i> widget. This happened after a server restart only. This problem has been solved and this exception does not occur anymore.
632511	Status information about an email being sent permanently visible The status information “sending ...”, which is displayed in the ticket history when an email has just been sent from the ticket, did not disappear. This problem has been solved and the message now disappears after reloading the page.
632577	Error when adding appointment to collapsed calendar An error occurred when trying to add a new appointment to a calendar in a collapsed <i>Calendar</i> section. This problem has been solved and it is possible to add an appointment when the <i>Calendar</i> section is collapsed.
632749	Two asterisks displayed for required MLAs in lists of structs Two red asterisks were displayed in case a required MLA belonged to a list of structs. This behavior has been corrected and only one asterisk is displayed now.
632809	Request parameters were logged for CM/Track in DEBUG mode In CM/Track , request parameters, including user passwords, could be logged when running in DEBUG mode. This problem has been solved and request parameters are not logged anymore.
632878	Corrupted database when adding the same field twice to an ACF or control form In some cases, it was possible to add fields which were already present in an ACF or control form a second time to the same form. This caused a corrupted database and the Admin Tool could not be opened again. This problem has been corrected and it is not possible anymore to add the same field twice to the same ACF or control form.

3. Version 6.11.1.2 (14.02.2018)

Version 6.11.1.2 includes all **ConSol CM** changes and additions of

- 6.11.0 versions up to 6.11.0.6
- 6.10 versions up to 6.10.8.0
- 6.9 versions up to 6.9.4.7
- 6.8 versions up to 6.8.5.8

Newer versions and the respective changes are not included in this **ConSol CM** version.

3.1 Update and installation instructions

Please read all information on update and installation of **ConSol CM** specific for version 6.11.1.2 in this section before performing a system update or installation. Please consult the general product documentation on update and installation procedures as well as earlier version Release Notes, if applicable.

3.1.1 System upgrade from 6.11.0 and earlier versions

This **ConSol CM** release needs specific planning for an update from earlier releases. Please check the instructions for release 6.11.1.0 ([System upgrade from 6.11.0 and earlier versions](#)).

No further instructions available.

3.2 Bugs fixed

The following defects have been corrected in this **ConSol CM** version.

Number	Description
633031	Visibility level deleted during update on MySQL and Microsoft SQL The visibility level <i>1st level short</i> , <i>2nd level full</i> , <i>3rd level full</i> was deleted from the visibility configuration in the <i>Classes of Text</i> section of the Admin Tool during an update on MySQL and Microsoft SQL. Text classes with this visibility level had an empty visibility level after the update. This problem has been solved and visibility configurations are not deleted anymore when updating ConSol CM .
633032	Deleted values of scripted autocomplete fields are sent to REST After editing the value of a scripted autocomplete field in CM/Track V2 and deleting it afterwards, the field was sent to the REST API as a string with value "null". This problem has been corrected and fields whose value has been deleted are now sent to the REST API as "null" objects.
633035	Customer security check not working for ticket fields in LISTS or STRUCTS Ticket fields belonging to LISTS or STRUCTS could be retrieved using the REST API when logging in as a customer even if they were annotated with <i>customer-exposure</i> = "none". This problem has been solved and these data fields are not available via REST API anymore.
633055	Update from older ConSol CM version could fail In some cases, an update from an older version of ConSol CM could fail due to some incorrect configuration. This problem has been solved and updates from old versions do not fail anymore.

4. Version 6.11.1.3 (05.03.2018)

Version 6.11.1.3 includes all **ConSol CM** changes and additions of

- 6.11.0 versions up to 6.11.0.6
- 6.10 versions up to 6.10.8.0
- 6.9 versions up to 6.9.4.7
- 6.8 versions up to 6.8.5.8

Newer versions and the respective changes are not included in this **ConSol CM** version.

4.1 Update and installation instructions

Please read all information on update and installation of **ConSol CM** specific for version 6.11.1.3 in this section before performing a system update or installation. Please consult the general product documentation on update and installation procedures as well as earlier version Release Notes, if applicable.

4.1.1 System upgrade from 6.11.0 and earlier versions

This **ConSol CM** release needs specific planning for an update from earlier releases. Please check the instructions for release 6.11.1.0 ([System upgrade from 6.11.0 and earlier versions](#)).

No further instructions available.

4.2 Changes

This section illustrates all relevant software product changes for version 6.11.1.3 of **ConSol CM**.

4.2.1 Performance improvements

Update performance improved for **MySQL** (#632989)

The database update script for updating to 6.11.1.3 has been optimized to improve update performance on **MySQL**.

4.2.2 Layout improvements

Size of the soccer ball icon reduced (#633095)

The soccer ball icon, which is displayed in the **Web Client** to indicate the assigned engineer of a ticket if there are additional engineers, was displayed too large in the engineer selector. The size of the soccer ball icon has been reduced to avoid overlappings. In addition, a more modern image is now used.

4.3 Bugs fixed

The following defects have been corrected in this **ConSol CM** version.

Number	Description
632929	SMTP password written to log file during system startup During the startup of the ConSol CM system, the password used to connect to the SMTP server was written to the <code>server.log</code> file. This problem has been solved and passwords are not written to the log files anymore.
633013	Problem when creating a relation When creating a relation (ticket-ticket or contact-company) a message that the request cannot be processed was displayed in some cases. This problem has been solved and creating a relation does not cause this error anymore.
633016	Error when using the <code>executeActivity()</code> method When using the method <code>ticket.executeActivity()</code> in workflow scripts and other scripts, a code validation error occurred. This problem has been fixed and the method can be used now.
633109	Empty attachment list when using paging When navigating the attachment list on a customer page in the Web Client for a customer with many attachments, an empty list could be displayed. This problem has been solved and the attachments are correctly displayed now.
633110	Error during update when Track user name or password contains a single quote The update to version 6.11.1.2 failed if a customer field with the annotation <code>contact authentication = "username"</code> or <code>"password"</code> contained a value including a single quote character. This problem has been solved and the update does not fail anymore in this case.
633110	Exception when editing customer When editing a customer whose data model contains a customer field of type STRING called <code>company</code> or <code>customer</code> , an exception occurred. This problem has been solved and the field names <code>company</code> and <code>customer</code> can now be used in customer data models.
633140	Caching problem when executing a customer or resource action In some cases, a caching problem could prevent the execution of a customer or resource action if the customer / resource data included an MLA inside a STRUCT. This problem has been solved and does not occur anymore.

5. Version 6.11.1.4 (30.03.2018)

Version 6.11.1.4 includes all **ConSol CM** changes and additions of

- 6.11.0 versions up to 6.11.0.6
- 6.10 versions up to 6.10.8.0
- 6.9 versions up to 6.9.4.7
- 6.8 versions up to 6.8.5.8

Newer versions and the respective changes are not included in this **ConSol CM** version.

5.1 Update and installation instructions

Please read all information on update and installation of **ConSol CM** specific for version 6.11.1.0 in this section before performing a system update or installation. Please consult the general product documentation on update and installation procedures as well as earlier version Release Notes, if applicable.

5.1.1 System upgrade from 6.11.0 and earlier versions

This **ConSol CM** release needs specific planning for an update from earlier releases. Please check the instructions for release 6.11.1.0 ([System upgrade from 6.11.0 and earlier versions](#)).

5.1.2 Script modification requirements

This release features some interface changes in API methods. **Please be aware that scripts using the respective methods may need adjustments!** The affected methods are listed in the following sections of this document:

- [API for scripted autocomplete fields improved \(#632870\)](#)
- [Session upgrade and downgrade \(#632899\)](#)
- [Method client.goToCreateUnit\(unit\) \(#633156\)](#)

5.1.3 Configuration modification requirements

The changes in this release require several configuration items to be reviewed and possibly adjusted. Please see the following sections for details:

- [Option to hide empty groups in the ticket list \(#633106\)](#)

No further instructions available.

5.2 New features

The features newly introduced in this **ConSol CM** version 6.11.1.4 are described in detail in this section.

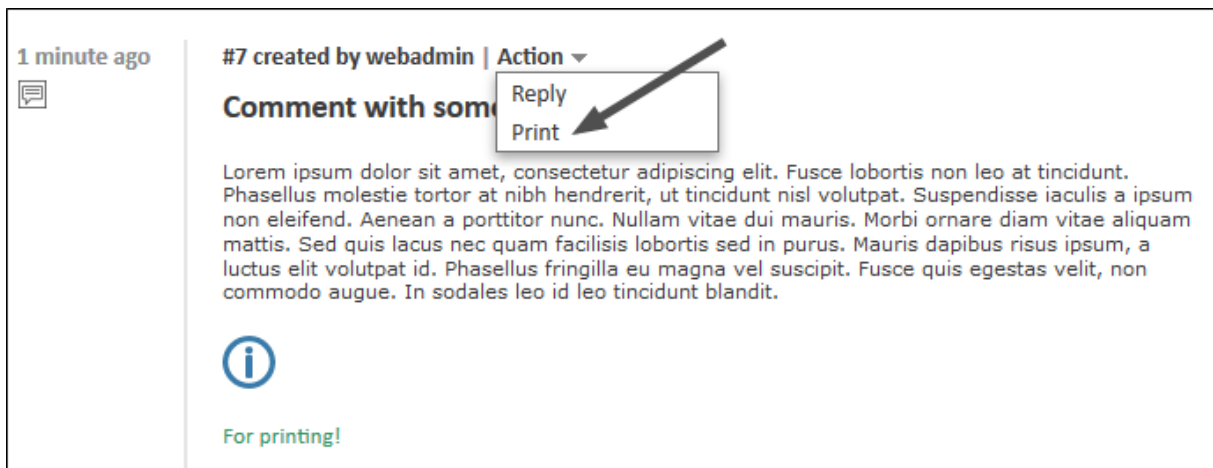
5.2.1 Printing emails and comments (#633142)

A new option *Print* has been added to the *Action* menu in the ticket history of the **Web Client**. This option is available for both incoming and outgoing emails and comments. It allows to print a file with the content of the email or comment. In addition, the date and time of the entry, the ticket number and subject, and the author of the entry are added to the printed output. This output can be sent to a printer, or printed to a PDF file using an installed PDF utility of the operating system.

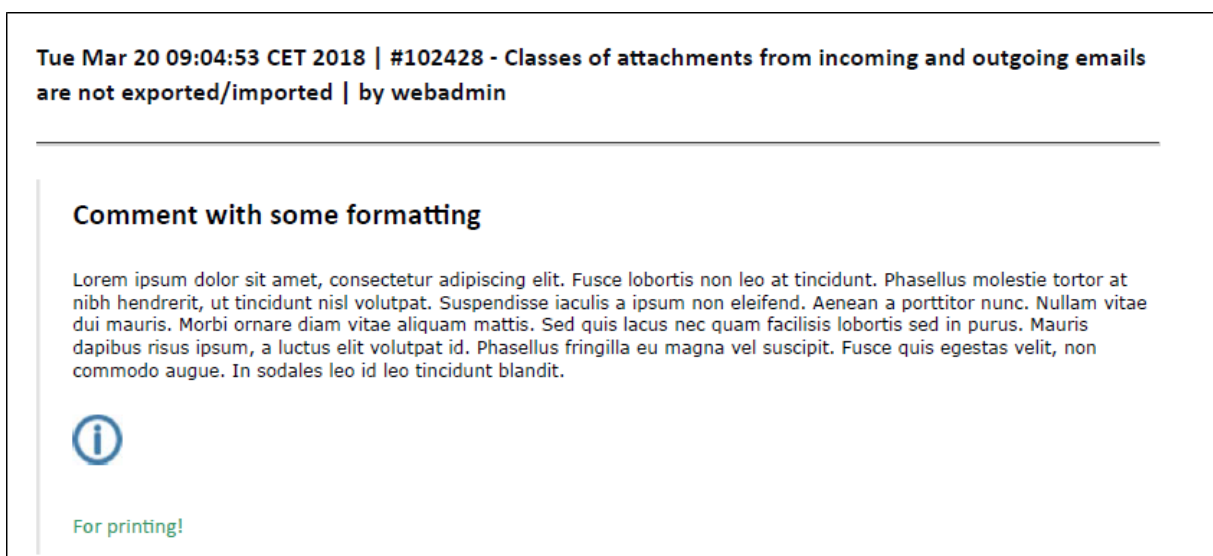


The PDF utility is not part of the **ConSol CM** delivery.

The following figure shows the new option:



The printed comment looks like this:





Entries which are shortened due to limits applied to the number of characters shown in the basic and/or extended view, are also shortened when printed, i.e. the content which is displayed in the **Web Client** is the content which is printed.

5.3 Changes

This section illustrates all relevant software product changes for version 6.11.1.4 of **ConSol CM**.

5.3.1 Layout improvements

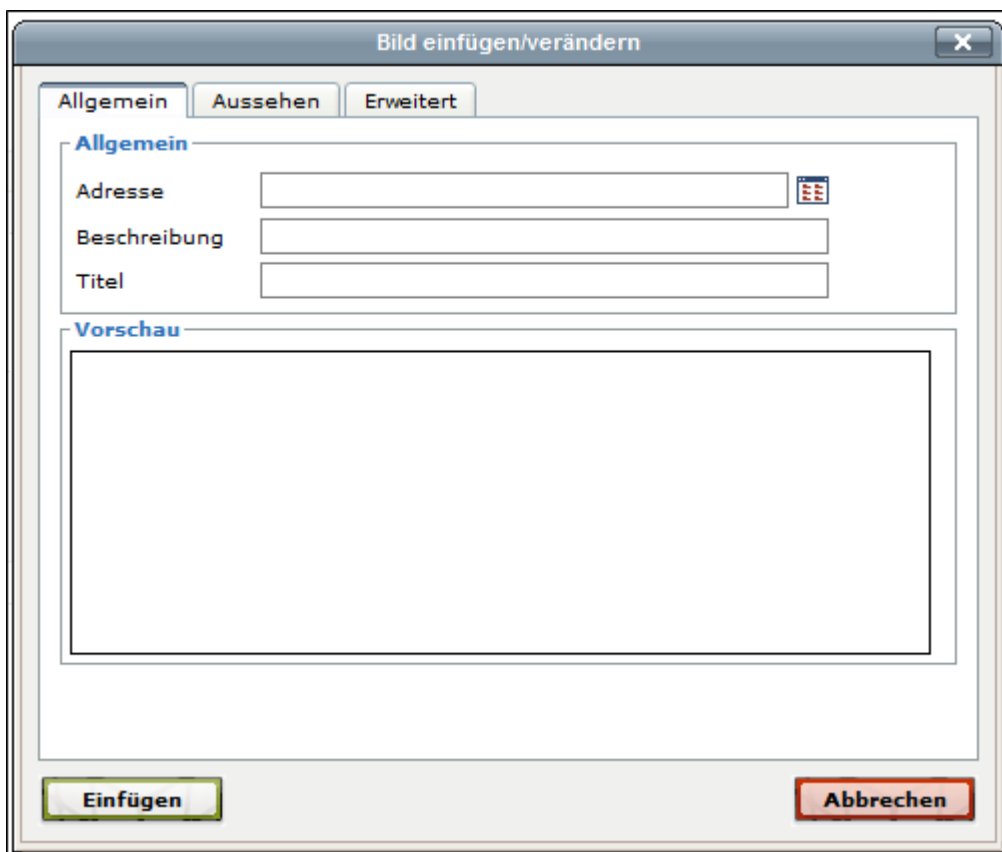
Position of date fields (#633176)

In the edit mode of the **Web Client**, data fields of type DATE overlapped the label of the next field if the date field was annotated with **accuracy** = "date-time" and the **position** annotations did not reserve two columns for the date field. The layout has been improved so that these date fields now fit into the designated space.

5.3.2 CM/Track V2 improvements

Improvement of the image editor (#633076)

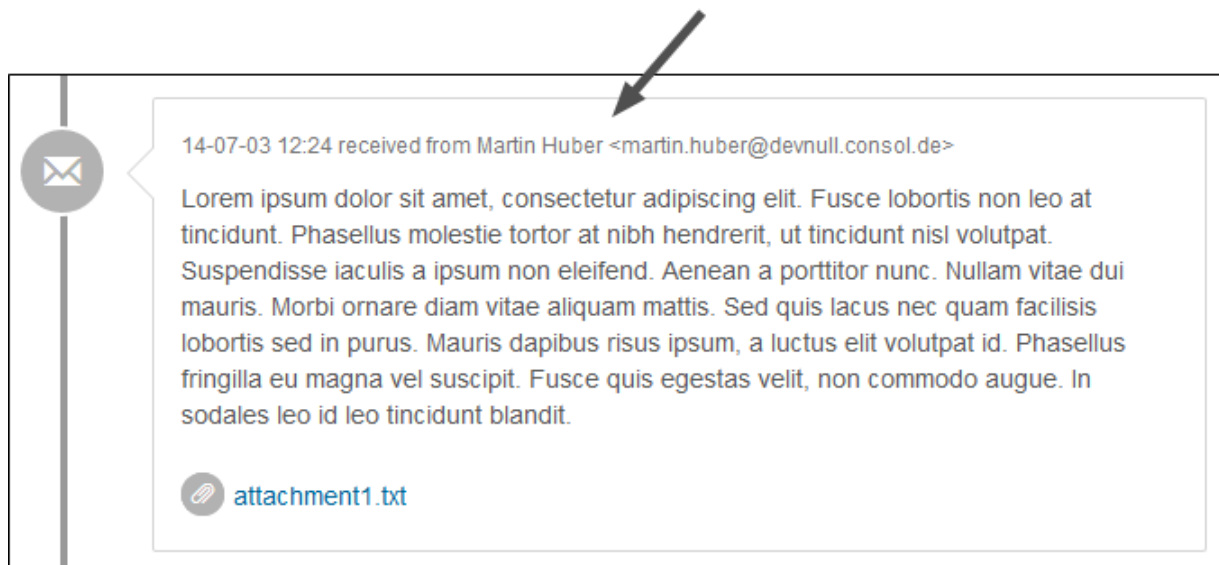
The image editor, which can be accessed by clicking the image icon in the comment editor of **CM/Track V2**, is now available in German. Previously, the user interface texts were not displayed correctly when using the editor with a German locale set in the browser.



In addition, the file browser to select an image now works as expected.

Information about email sender (#633112)

Emails which are listed in the ticket history in **CM/Track V2** now include more details about the sender. The header of the email entry now states the name and email address of the sender and whether the email was sent or received.



5.3.3 API for scripted autocomplete fields improved (#632870)

The **ConSol CM** API for the scripted autocomplete fields feature has been improved. Previously, the `add()` method had only two signatures allowing to add any object and optionally a string (see [New handling of scripted autocomplete fields \(#632555\)](#) for further information).

Now, there are two signatures for each type of object (engineer, resource, ticket, unit, and string). One signature containing only the object itself, which is used to implement a scripted autocomplete list with dynamically rendered labels, and one containing the object and a string, which serves as a static label.

The previous signatures are still valid and can be used for adding autocomplete values to the list.

The following figure shows all available methods:

Method Summary	
Methods	
Modifier and Type	Method and Description
<code>ScriptAutocompleteResult<T></code>	<code>add(Collection<?> pElements)</code> Add many <code>AutocompleteResultItems</code> (without static display values).
<code>ScriptAutocompleteResult<T></code>	<code>add(Engineer pEngineer)</code> Add an <code>AutocompleteResultItem</code> with a dynamically rendered display value.
<code>ScriptAutocompleteResult<T></code>	<code>add(Engineer pEngineer, String pDisplayValue)</code> Add an <code>AutocompleteResultItem</code> .
<code>ScriptAutocompleteResult<T></code>	<code>add(Map<?,String> pElements)</code> Add many <code>AutocompleteResultItems</code> (with static display values).
<code>ScriptAutocompleteResult<T></code>	<code>add(Object... pElements)</code> Add many <code>AutocompleteResultItems</code> (without static display values).
<code>ScriptAutocompleteResult<?></code>	<code>add(Resource pResource)</code> Add an <code>AutocompleteResultItem</code> with a dynamically rendered display value.
<code>ScriptAutocompleteResult<T></code>	<code>add(Resource pResource, String pDisplayValue)</code> Add an <code>AutocompleteResultItem</code> .
<code>ScriptAutocompleteResult<?></code>	<code>add(String pInternalValue)</code> Add an <code>AutocompleteResultItem</code> .
<code>ScriptAutocompleteResult<T></code>	<code>add(String pInternalValue, String pDisplayValue)</code> Add an <code>AutocompleteResultItem</code> .
<code>ScriptAutocompleteResult<?></code>	<code>add(Ticket pTicket)</code> Add an <code>AutocompleteResultItem</code> with a dynamically rendered display value.
<code>ScriptAutocompleteResult<T></code>	<code>add(Ticket pTicket, String pDisplayValue)</code> Add an <code>AutocompleteResultItem</code> .
<code>ScriptAutocompleteResult<?></code>	<code>add(Unit pUnit)</code> Add an <code>AutocompleteResultItem</code> with a dynamically rendered display value.
<code>ScriptAutocompleteResult<T></code>	<code>add(Unit pUnit, String pDisplayValue)</code> Add an <code>AutocompleteResultItem</code> .
<code>List<AutocompleteResultItem<T>></code>	<code>getItems()</code> Get aggregated <code>AutocompleteResultItems</code> .
<code>String</code>	<code>getMessage()</code> Get message to display.
<code>static ScriptAutocompleteResult<String></code>	<code>noResults(String pMessage)</code> Create a <code>ScriptAutocompleteResult</code> .

5.3.4 Session upgrade and downgrade (#632899)

The **ConSol CM** API has been enhanced to provide a method for upgrading and downgrading an engineer session. This enables advanced integration scenarios using the webhooks functionality. The webhook script is executed with administrator privileges, but the script can include the credentials of an engineer and some code to downgrade the session to the engineer's privileges.

The method `executeWithUserPermissions` was added to the class `SecurityTemplate` for this purpose.

5.3.4.1 Scripting example

The following code example shows the usage of this method in a script. The session is downgraded to the permissions of the user identified by the login and password included in the JSON payload. Then, a unit search for the unit with the ID "12345" is performed with these permissions.

```
import com.consol.cmas.common.security.template.SecurityCallbackWithoutResult;
import com.consol.cmas.common.security.template.SecurityTemplate;
import groovy.json.JsonSlurper

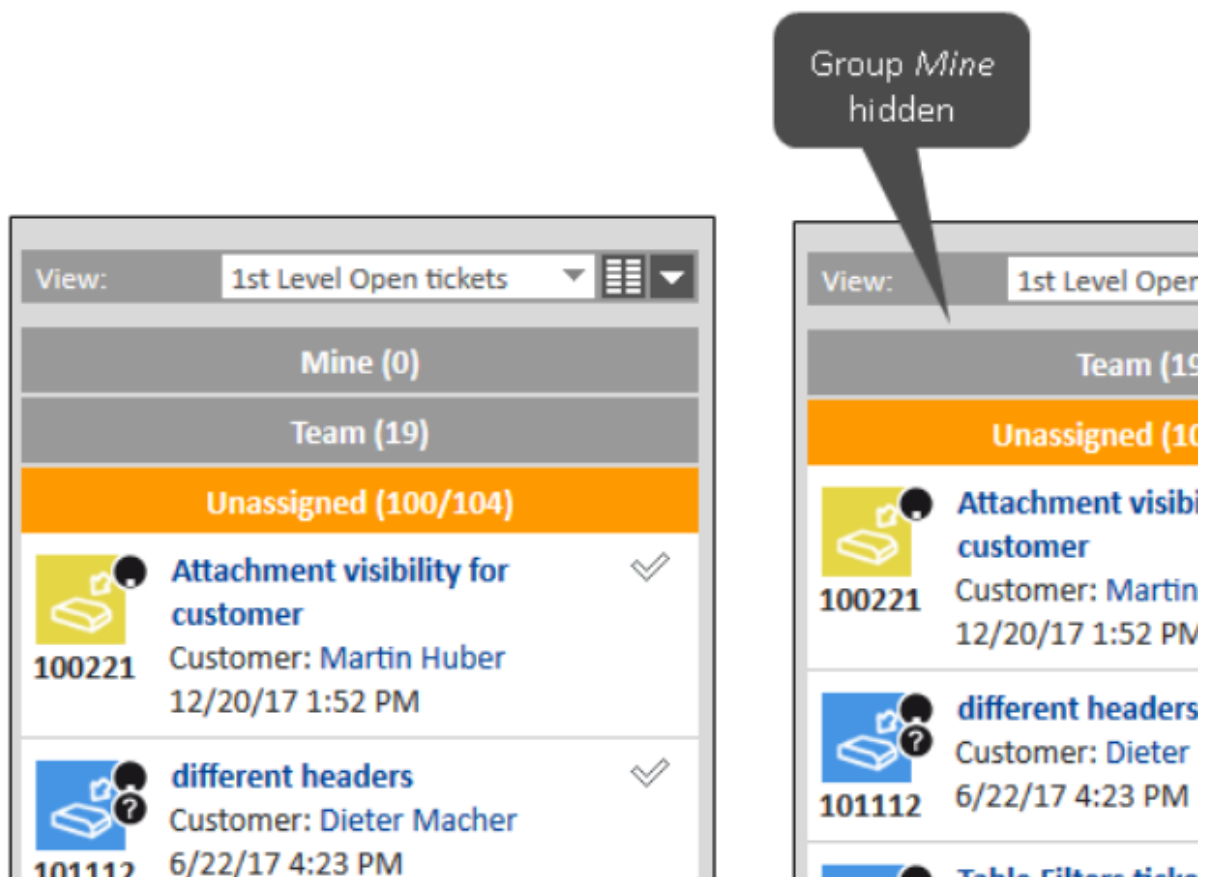
def jsonSlurper = new JsonSlurper()
def message = jsonSlurper.parseText(payload);
SecurityTemplate.executeWithUserPermissions(message.login, message.password,
    new SecurityCallbackWithoutResult() {
        @Override
        public void doInSecurityContextWithoutResult() {
            unitService.getId(12345)
        }
    })
});
```

5.3.5 Calendar section expanded for adding appointment (#632906)

When adding an appointment to the calendar in the **Web Client**, the calendar section is now automatically expanded if it was collapsed. This change prevents problems which occurred when opening an appointment added while the calendar section was collapsed.

5.3.6 Option to hide empty groups in the ticket list (#633106)

Empty groups in the ticket list of the **Web Client** can now be hidden to make the ticket list more concise. For this purpose, the new attribute *hideEmptyGroups* has been added to the page customization of *accordionTicketList* and its subscopes. This attribute allows the administrator to decide how empty groups should be handled. By default the attribute is set to “false”, meaning that all groups are displayed. You can set it to “true” to hide the empty groups, as shown in the figure below.





A ticket list group is not hidden if it has the focus. In this case the group is displayed when the ticket list is loaded, and hidden once the user clicks another group.

If the ticket list is grouped by queue, the queues are now sorted alphabetically by localized queue name.

5.3.7 Method `client.goToCreateUnit(unit)` (#633156)

The method `client.goToCreateUnit(unit)` of the **ConSol CM** API class `OperationResponseBuilder` only worked correctly for contacts in a one-level customer data model. It can now be used for companies in one-level data models and for contacts and companies in two-level data models as well.

5.4 Bugs fixed

The following defects have been corrected in this **ConSol CM** version.

Number	Description
632722	Error after renaming view After renaming a view in the Admin Tool , it could happen that engineers whose roles had this view assigned could not log in. This problem has been solved and renaming a view does not cause engineers to be unable to log in anymore.
632743	URLs truncated in tables Data fields with the annotation <code>texttype = "url"</code> which contain long values are truncated when displayed in tables in the Web Client , for example in the Detail Search or the contacts section of a company page. It was not possible to click the URL from a table because the complete, non-truncated URL was not available. This problem has been solved and URLs are only shortened for display but the link underneath remains complete and can be clicked to open the desired URL.
632793	Wrong history entry when adding or removing resource relation If lazy loading is enabled for tickets in the Web Client , ticket history entries referring to additions or deletions of resource relations were not displayed correctly. They showed the label that the resource was not available instead of the resource template. This problem has been solved and the resource template is now used.
633009	Superfluous white space in ticket list In some cases, some additional white space was displayed at the lower end of the ticket list in the Web Client . This white space was not needed and therefore removed.
633137	Duplicated ACF initialization script after workflow deployment ACF initialization scripts from the Process Designer were duplicated instead of overwritten during the redeployment of a workflow. This problem has been solved and these scripts are not duplicated anymore.
633163	Cache issue after workflow deployment in clustered environment After workflow deployment, caching problems could occur in clustered environments resulting in exceptions in the log files. This problem has been solved, so the caches are correctly cleared after workflow redeployment.
633187	Error in script execution framework There was an error in the script execution logic which could cause memory leak when large update operations were performed. This problem has been solved and such memory leaks do not occur anymore.

6. Version 6.11.1.5 (26.04.2018)

Version 6.11.1.5 includes all **ConSol CM** changes and additions of

- 6.11.0 versions up to 6.11.0.6
- 6.10 versions up to 6.10.8.0
- 6.9 versions up to 6.9.4.7
- 6.8 versions up to 6.8.5.8

Newer versions and the respective changes are not included in this **ConSol CM** version.

6.1 Update and installation instructions

Please read all information on update and installation of **ConSol CM** specific for version 6.11.1.0 in this section before performing a system update or installation. Please consult the general product documentation on update and installation procedures as well as earlier version Release Notes, if applicable.

6.1.1 System upgrade from 6.11.0 and earlier versions

This **ConSol CM** release needs specific planning for an update from earlier releases. Please check the instructions for release 6.11.1.0 ([System upgrade from 6.11.0 and earlier versions](#)).

No further instructions available.

6.2 Changes

This section illustrates all relevant software product changes for version 6.11.1.5 of **ConSol CM**.

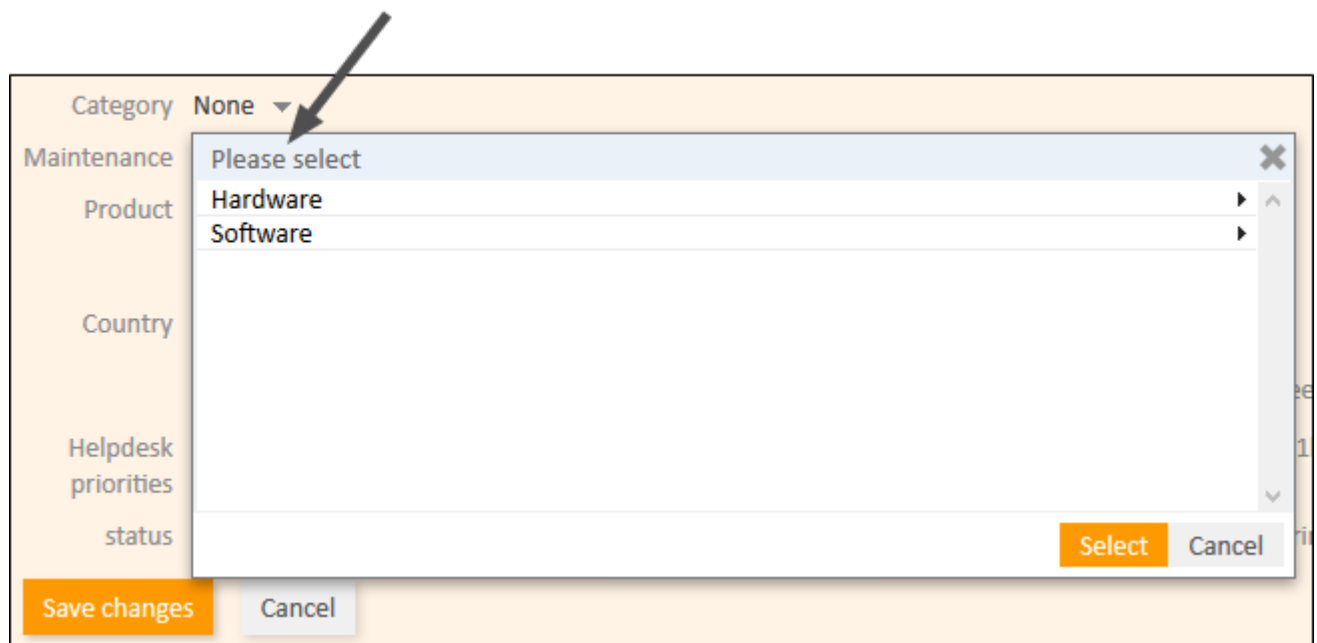
6.2.1 Usability improvements

User name for Task Execution Framework (#633283)

The name of the author of history entries generated by the Task Execution Framework has been changed. Now, the author is called “System Task” in English and “Systemtask” in German. This applies to the ticket, customer, and resource history, as well as the recent changes widget of the **Web Client**.

Heading of MLA box improved (#633297)

The heading of the MLA box in the **Web Client** has been improved. It is now *Please select* in English and *Bitte wählen* in German.



6.2.2 CM/Track V2 improvements

Creation of tickets improved (#633269)

The creation of tickets in **CM/Track V2** has been improved so that the *Create ticket* button is disabled while the ticket creation is being processed by the **ConSol CM** server. This avoids situations where the ticket is created twice because the user clicks the *Create ticket* button twice in case the ticket creation takes longer than expected.

6.2.3 Method to open action forms for customers and resources (#633102)

The **ConSol CM** API has been enhanced with a method which allows to open an action form for a customer or resource using the new interface for action-related operations, which was introduced in version 6.11.1.0 (see [Unified interface for action-related operations \(#629296\)](#)).

The additional method `openActionForm()` is available for customers or resources. Please see the following examples.

Customers:

```
client.goToUnit(unit).openActionForm('EditContact');
```

Resources:

```
client.goToResource(resource).openActionForm('EditResource');
```

6.2.4 Duplicate entries removed from domain.xml (#633285)

The XML configuration file `domain.xml` has been updated to remove a duplicate entry concerning the audit log.

6.2.5 VOCaching disabled in cluster mode (#631155)

The VOCaching feature introduced in **ConSol CM** version 6.11.1.1 (see [New caching mechanism enabled \(#632523\)](#)) is now automatically disabled in clustered environments. This change was necessary to prevent problems that changes made to objects on one node were not visible on the other nodes.



This change overrides the setting of the system property `cmweb-server-adapter.voCacheEnabled` on systems where `cmas-core-shared.cluster.mode` is set to “true”.

6.3 Bugs fixed

The following defects have been corrected in this **ConSol CM** version.

Number	Description
631108	Hanging tasks in the <i>Admin Tool</i> It could happen that tasks in the status <i>Canceling</i> remained in the <i>Task Execution</i> section of the Admin Tool . Tasks whose task script includes <code>ExecutionSpecification().setRetryRequested(true)</code> are now automatically removed after a certain time frame.
631692	Pasting images did not work in the <i>Web Client</i> When the user copied an image from a comment or email in the Web Client and tried to paste it into another comment or email, a symbol was displayed instead of the image. This problem has been solved for copying content from the same ConSol CM system.
633135	Detail search filter does not work as expected for ENUM data fields When ENUM data fields were used as search criteria in the Detailed Search in the Web Client , the localized value of the field was taken into account leading to unexpected search results in some cases. This problem has been solved and the Detailed Search now works as expected.
633279	Switching to DWH LIVE mode did not work in clustered environments In clustered environments, the DWH mode could not be switched to “LIVE” while the server was running. This problem has been solved and the LIVE mode is now activated correctly.
633333	Scene import failed due to scripts The import of an existing scene could fail due to missing information for scripts. This problem has been solved and the scene is correctly imported now.

7. Version 6.11.1.6 (29.05.2018)

Version 6.11.1.6 includes all **ConSol CM** changes and additions of

- 6.11.0 versions up to 6.11.0.6
- 6.10 versions up to 6.10.8.0
- 6.9 versions up to 6.9.4.7
- 6.8 versions up to 6.8.5.8

Newer versions and the respective changes are not included in this **ConSol CM** version.

7.1 Update and installation instructions


Please read all information on update and installation of **ConSol CM** specific for version 6.11.1.6 in this section before performing a system update or installation. Please consult the general product documentation on update and installation procedures as well as earlier version Release Notes, if applicable.

7.1.1 System upgrade from 6.11.0 and earlier versions

This **ConSol CM** release needs specific planning for an update from earlier releases. Please check the instructions for release 6.11.1.0 ([System upgrade from 6.11.0 and earlier versions](#)).

7.1.2 Data warehouse update

After updating the **ConSol CM** system to version 6.11.1.6, the DWH needs to be updated due to changes for the contact deletion feature (see [Contact deletion for GDPR \(#633392\)](#)) and the secure SMTP support ([Support for secure SMTP introduced \(#633298\)](#)). The update scripts for each database are available from your ConSol representative. Please execute the update script for your database engine. This step can be skipped if a full reinitialization of the DWH is performed in the **Admin Tool**.

 At the first start of the **ConSol CM** system after updating to 6.11.1.6., some log entries regarding data inconsistencies might be added to the log files. These log entries can be ignored as the inconsistencies are fixed automatically.

7.1.3 Script modification requirements

This release features some interface changes in API methods. Some methods were added or changed. **Please be aware that scripts using the respective methods may need adjustments!** The affected methods are listed in the following sections of this document:

- [Contact deletion for GDPR \(#633392\)](#)
- [Rendering of empty scripted visualization fields introduced \(#632892\)](#)
- [API method RequestLocale.get\(\) improved to retrieve locale in CM/Track V2 \(#633249\)](#)
- [API and logs for clusters improved \(#633357\)](#)

7.1.4 Configuration modification requirements

The changes in this release require several configuration items to be reviewed and possibly adjusted. Please see the following sections for details:

- [Support for secure SMTP introduced \(#633298\)](#)


No further instructions available.

7.2 New features


The features newly introduced in this **ConSol CM** version 6.11.1.6 are described in detail in this section.

7.2.1 Contact deletion for GDPR (#633392)

A new option to delete contact data from **ConSol CM** has been introduced in order to comply with the GDPR (General Data Protection Regulation) which takes effect on May 25, 2018. This feature allows to completely delete a contact from the system, including his tickets (where the contact is the main contact) and relations to other entities (resources, contacts, companies, and other tickets). References to the contact in the history of other objects are anonymized. This applies both to **ConSol CM** and **CMRF**.

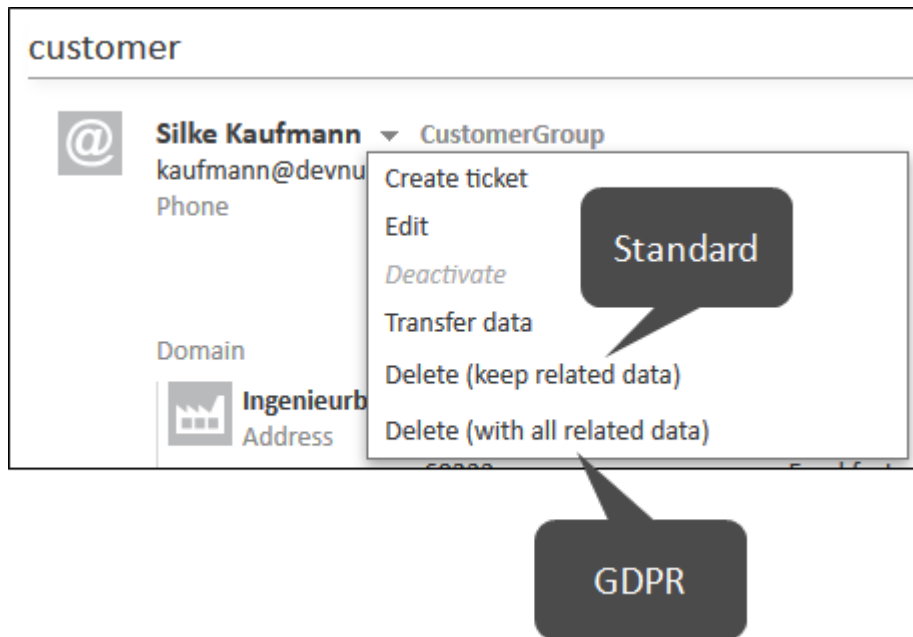
 If a contact is deleted, he is completely removed from the **CMRF** as well. This means that neither the contact nor his tickets will show up in reports anymore.

The deletion can be performed both manually in the **Web Client** and via script.

 Please note that this feature is available only for customers of the type *contact*. Nevertheless, you can try to delete a company using a script as long as the deletion does not violate any business constraint (e.g., there are contacts which belong to the required company to be deleted).

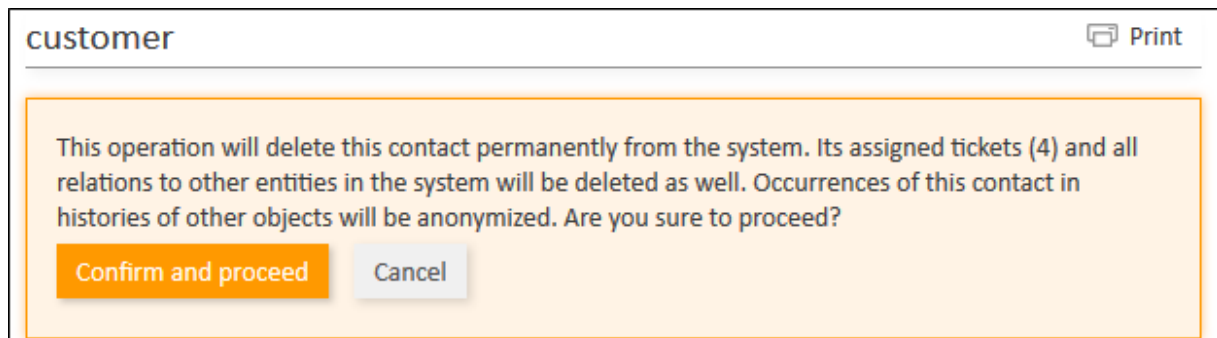
The context menu of contacts in the **Web Client** has been enhanced to provide two different actions for deleting the contact:

- **Delete (keep related data)**
This is the standard deletion (previous behavior). The contact is deleted but some related data might remain in the system. The contact's tickets are transferred to another contact.
- **Delete (with all related data)**
This is the new feature for GDPR-compliant deletion. The contact is deleted with his related data and all remaining references are anonymized.



Both options are available for users with delete permissions for the respective customer group. The GDPR-compliant contact deletion cannot be switched off.

After clicking the *Delete (with all related data)* option, a warning similar to the following is displayed to inform the user about the scope of deletion.



The **ConSol CM** API has been enhanced to provide methods for deleting contacts via script. The class **UnitGDPRService** has been added for this purpose. It provides the following two methods to delete the contacts identified by their IDs:

Method Summary	
Methods	
Modifier and Type	Method and Description
void	<code>delete(Collection<Long> pUnitIds)</code> Execute unit deletion and anonymization logic in a current thread managing transactions internally.
<code>Future<?></code>	<code>deleteAsynchronously(Collection<Long> pUnitIds)</code> Execute unit deletion and anonymization logic in a separate thread managing transactions internally.

The customer deletion script is executed directly after initiating the delete operation and before deleting the contact. It is therefore possible to introduce some custom code for the deletion procedure. The newly introduced variable `deleteType` can be used in customer action scripts for delete actions. It can have two values:

- **delete**
The executed action is *Delete (keep related data)*.
- **delete_all**
The executed action is *Delete (with all related data)*.

The delete operation is executed in the following order:

- Relations to customers, resources, and engineers are deleted.
- Tickets where the contact is the main customer are deleted.
- The contact is removed from tickets where he is an additional customer.
- Data fields of the type *autocomplete* (unit) are deleted.
- The history and content of the contact is deleted.
- References to the contact within the history of other tickets, customers, and resources are anonymized (contact name is replaced by “not available anymore”).
- Favorites pointing to the contact are deleted.
- Recent items pointing to the contact are deleted.
- The contact himself is deleted.



Please take into account the following specifics when working with the GDPR deletion feature:

- Only tickets where the contact is the main customer are deleted. This means that personal data remains in the CM system if it is located in tickets where the contact is now an additional contact, even if the contact was the main contact before. This is the case for example when a customer creates a ticket by sending an email, and another contact is set as the main customer later on.



A DWH update is required for this change. The update scripts for each database are available from your ConSol representative. Please execute the update script for your database engine.

Scripting examples

The following code example shows a customer search script which deletes the contacts found in the search.

```
import java.util.concurrent.*;

def ids = []
for (Unit unit : units) {
    ids.add(unit.id)
}

def future = unitGDPRService.deleteAsynchronously(ids);
```

```
try {
    future.get(50, TimeUnit.SECONDS);
} catch (InterruptedException e) {
    log.warn("some data may not have been deleted, script interrupted, unit ids "
        + ids);
    future.cancel(true); // here we decide NOT to continue in the background
    return client.failure().showWarningMessage("some data may not have been
        deleted, please retry").withRefreshContent();
} catch (TimeoutException e) {
    log.warn("some data may not have been deleted, deletion timeout, continuing
        in the background, unit ids " + ids);
    // here we decide to continue in the background
    return client.success().showWarningMessage("some data may not have been
        deleted, continuing in the background, check later").withRefreshContent()
} catch (ExecutionException e) {
    // e.g. deleting a company where its customer must have company assigned, or
    // other validation failures
    log.warn("some data may not have been deleted, execution exception, unit ids
        " + ids, e);
    return client.failure().showWarningMessage("some data may not have been
        deleted, fix the cause and retry: " + e.cause.message).withRefreshContent();
} catch (CancellationException e) {
    log.warn("some data may not have been deleted, execution cancelled, unit ids
        " + ids);
    return client.failure().showWarningMessage("some data may not have been
        deleted, please retry").withRefreshContent();
}

Thread.sleep(2000); // let the indexer catch up
return client.success().showInfoMessage("success").withRefreshContent();
```

The following code example shows how to distinguish between the two different delete actions in a customer script executed after delete operations:

```
if (deleteType == "delete") {
    // react on normal deletion
} else if (deleteType == "delete_all") {
    // react on deletion with related data
}
```

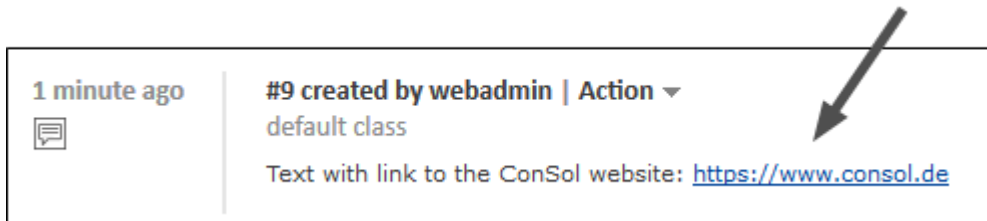
7.3 Changes

This section illustrates all relevant software product changes for version 6.11.1.6 of **ConSol CM**.

7.3.1 Usability improvements


Link styling improved (#632771)

Links added in the comment or email editor of a ticket in the **Web Client** are now displayed in a standard link style (blue color and underline) to help the user to distinguish them from the rest of the text.



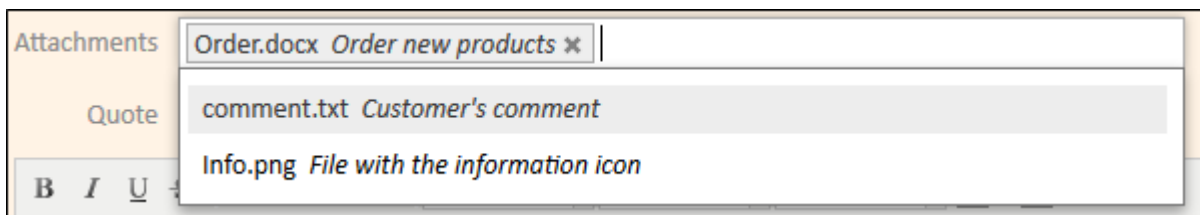
In addition, the links to the attachments in the *Attachments* section of a ticket, links in data fields, and links in the history are now displayed in blue color.

When the user hovers over a link, the link is displayed in orange color.

 This applies only to links which do not have any custom CSS style.

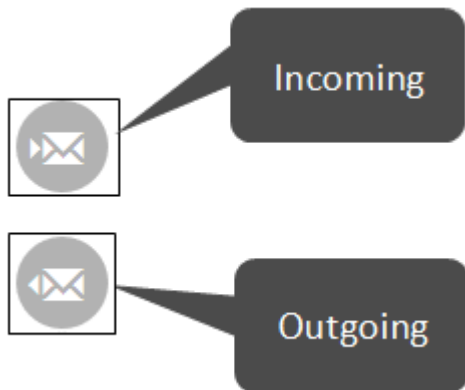
Attachment selector improved (#633183)

The attachment selector in the comment and email editors of a ticket in the **Web Client** has been improved. The newest attachments are now displayed first and the space for the attachment name has been increased. In addition, the attachments are now selected by clicking them in the drop-down list. They can be removed by clicking the little cross next to the attachment name in the multiselect field.



Email icons in **CM/Track V2** improved (#633212)

The ticket history in **CM/Track V2** now features two different icons for emails to distinguish incoming emails from outgoing emails. The following icons are now used:



Alignment of the control elements in the email section improved (#633286)

The alignment of the control elements for quoting when drafting an email for a ticket in the **Web Client** has been improved to avoid space problems in German.

Copyright updated (#633290)

The copyright displayed in the internal splash screen of **ConSol CM** and in the **Process Designer** has been updated to 2018.

CM/Phone installation page updated (#633364)

The installation page for **CM/Phone** has been updated. It was adapted to the current corporate design and now contains the latest system requirements.

7.3.2 Rendering of empty scripted visualization fields introduced (#632892)

The possibility to configure if a scripted visualization field should be rendered in the **Web Client** if it is empty has been added to **ConSol CM**. Previously, empty scripted visualization fields were treated like other data fields and therefore were never rendered in view mode. It was necessary to insert a dummy value for the visualization script to be executed. This dummy value is not required anymore, and configurations where the scripted visualization field should not hold any value but display data retrieved from other places are now supported.

The annotation **visualize-when-empty** (annotation group **common**) has been added to the **Admin Tool** to allow the rendering of empty scripted visualization fields in the **Web Client**. If this annotation is assigned to the field and set to "true", the field's visualization script is executed even if the field is empty.

This annotation only takes effect if it is assigned to a data field which is used for scripted field visualization (annotation **visualization**). It cannot be used for label fields.



This change only applies to the **Web Client**. Support for **CM/Track V2** is planned for a future release.

Field visualizations do not work in the *Related resources* section of the **Web Client**. This limitation also applies to the rendering of empty scripted visualization fields.

7.3.3 Usage of wildcards in the Quick Search improved (#633086)

By default, wildcards are appended automatically to each search term entered in the Quick Search in the **Web Client**. The behavior of these wildcards has been changed, so that wildcards are not appended to single characters. This applies both to single letters and to single numbers. Wildcards are now appended only to search terms consisting of two or more characters.

7.3.4 API method `RequestLocale.get()` improved to retrieve locale in **CM/Track V2** (#633249)

The **ConSol CM** API method `RequestLocale.get()` can now be used for autocomplete fields to retrieve the locale set in the browser from which the user accesses **CM/Track V2**. Therefore, it is now possible to properly localize scripted autocomplete fields in **CM/Track V2**.

It is necessary to import `com.consol.cmas.restapi.RequestLocale` in the code of the autocomplete script in order to use this method.

7.3.5 Support for secure SMTP introduced (#633298)

ConSol CM now supports secure SMTP for sending emails. Previously, it was supported only for receiving emails.

Two new system properties have been introduced for this purpose:

- **cmas-core-server, mail.smtp.tls.enabled**
Activates secure SMTP for sending emails from the **Web Client** and scripts. The default value is "false". If it is set to "true", secure SMTP is activated for sending emails.
- **cmas-dwh-server, notification.tls.enabled**
Activates secure SMTP for sending notification emails from the DWH. The default value is "false". If it is set to "true", secure SMTP is activated for sending notifications from the DWH.



Depending on the configuration of the used email server, it might be necessary to set the following system properties to values matching the SMTP user:

- cmweb-server-adapter, mail.from
- cmas-workflow-engine, jobExecutor.mailFrom
- cmas-core-security, admin.email



A DWH update is required for this change. The update scripts for each database are available from your ConSol representative. Please execute the update script for your database engine.

7.3.6 API and logs for clusters improved (#633357)

The **ConSol CM** API for reacting on cluster events and the cluster logs have been improved.

The new class `ClusterTopologyEvent` provides several methods for analyzing the cluster and reacting on cluster events. In addition, the interface `CacheService` has been extended with methods to clean, enable, and disable caches.

Changes to the cluster size are now written to the `cluster.log` file. The `domain.xml` configuration file has been updated with new settings for the cluster log file.

7.4 Bugs fixed

The following defects have been corrected in this **ConSol CM** version.

Number	Description
632971	<p>Workflow deployment not possible as workflow administrator</p> <p>A user with workflow administrator permissions was not authorized to deploy a workflow using the Process Designer. The permission <i>Administrate system</i> was required. This problem has been solved and it is now possible to deploy a workflow with workflow administrator permissions.</p>
633034	<p>CM/Track V2 session was lost</p> <p>In some cases, the user could get logged out from CM/Track V2 due to missing synchronization between the ConSol CM server session and the CM/Track V2 web container session. This problem has been solved and the CM/Track V2 session timeout is now aligned with the server session timeout.</p>
633181	<p>Unnecessary history entries added to the customer page in the Web Client</p> <p>When an ETL transformation which includes a setting for the default Track user name of a contact was executed, a history entry about the change of the Track user was written to the customer history, although the Track user was not changed by the transformation. This problem has been solved and such history entries are only written if the Track user has been updated.</p>
633211	<p>Validation error for list ignored in ACF</p> <p>Validation errors for lists of structs were ignored in an ACF in the Web Client when clicking the <i>Save and continue</i> button for the second time. It was then possible to send an ACF although a validation error for the list existed. This problem has been solved and validation errors for lists of structs are not ignored anymore when trying to send the ACF for the second time.</p>
633465	<p>Customer login was not possible after updating customer data</p> <p>If a customer was updated using an API method call after an update of the ConSol CM system to a 6.11 version, it could happen that the customer was not able to log in to CM/Track V2 or the REST API afterwards. This was due to the password being changed erroneously during the customer update. This problem has been solved and the password is now changed only if a new password is set.</p>

8. Version 6.11.1.7 (11.07.2018)

Version 6.11.1.7 includes all **ConSol CM** changes and additions of

- 6.11.0 versions up to 6.11.0.6
- 6.10 versions up to 6.10.8.0
- 6.9 versions up to 6.9.4.7
- 6.8 versions up to 6.8.5.8

Newer versions and the respective changes are not included in this **ConSol CM** version.

8.1 Update and installation instructions

Please read all information on update and installation of **ConSol CM** specific for version 6.11.1.7 in this section before performing a system update or installation. Please consult the general product documentation on update and installation procedures as well as earlier version Release Notes, if applicable.

8.1.1 System upgrade from 6.11.0 and earlier versions

This **ConSol CM** release needs specific planning for an update from earlier releases. Please check the instructions for release 6.11.1.0 ([System upgrade from 6.11.0 and earlier versions](#)).

8.1.2 Data warehouse update

After updating the **ConSol CM** system to version 6.11.1.7, the DWH needs to be updated due to changes for the contact anonymization feature (see [Contact anonymization for GDPR \(#633502\)](#)). The update scripts for each database are available from your ConSol representative. Please execute the update script for your database engine. This step can be skipped if a full reinitialization of the DWH is performed in the **Admin Tool**.

8.1.3 Script modification requirements

This release features some interface changes in API methods. Some methods were added or changed. **Please be aware that scripts using the respective methods may need adjustments!** The affected methods are listed in the following sections of this document:

- [Contact anonymization for GDPR \(#633502\)](#)
- [New API methods related to ticket closing date \(#627989\)](#)
- [Automatic ticket update for Webhooks \(#633444\)](#)
- [Content of pContext variable in actions changed \(#633475\)](#)

8.1.4 Configuration modification requirements

The changes in this release require several configuration items to be reviewed and possibly adjusted. Please see the following sections for details:

- [Contact anonymization for GDPR \(#633502\)](#)
- [Whitelist for links in incoming emails \(#632591\)](#)
- [Forwarding of manual emails configurable \(#633499\)](#)

No further instructions available.

8.2 New features

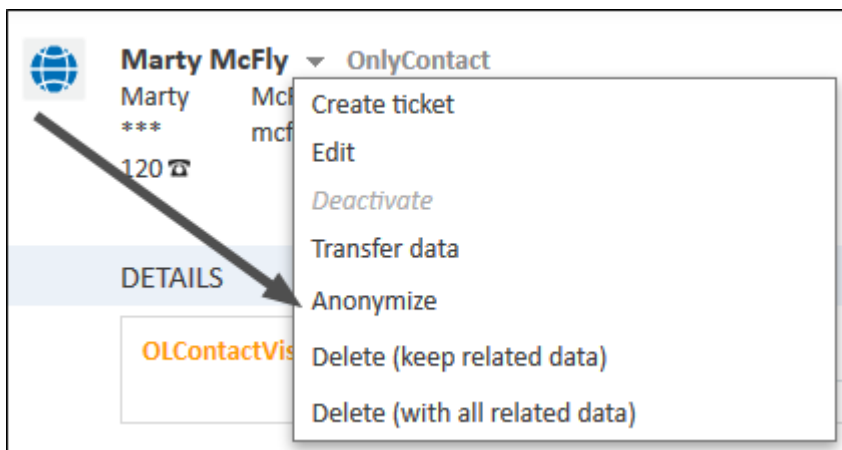
The features newly introduced in this **ConSol CM** version 6.11.1.7 are described in detail in this section.

8.2.1 Contact anonymization for GDPR (#633502)

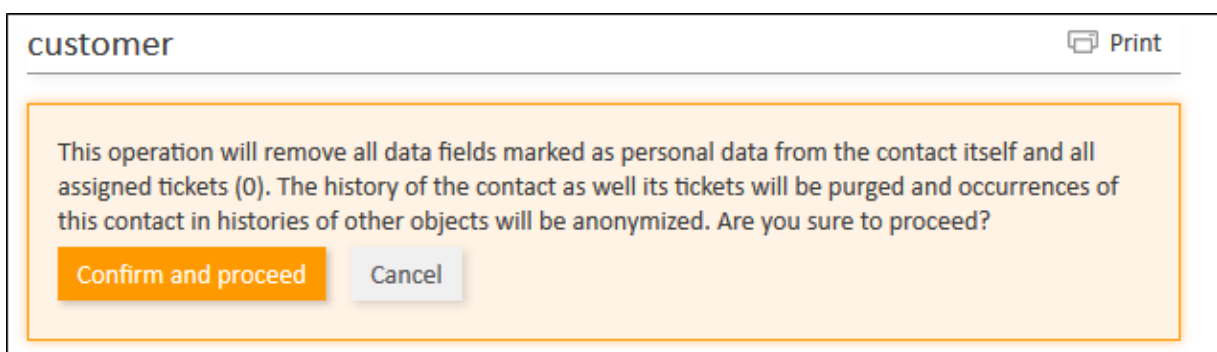
A new option to anonymize contact data from **ConSol CM** has been introduced in order to comply with the GDPR (General Data Protection Regulation) which took effect on May 25, 2018. This feature extends the [Contact deletion for GDPR \(#633392\)](#) feature introduced in 6.11.1.6 by allowing to remove personal data from **ConSol CM** and **CMRF** while preserving some basic information for reporting purposes.

The anonymization can be performed both manually in the **Web Client** and via script. Delete permissions for the respective customer group are required to anonymize a contact. The contact anonymization feature cannot be switched off.

The context menu of contacts in the **Web Client** has been enhanced to provide a new action for anonymizing contact data:



After clicking the *Anonymize* option, a warning similar to the following is displayed to inform the user about the scope of anonymization.



The annotation **personal-data** (annotation group **anonymize**) has been added to the **Admin Tool**. This annotation can be assigned to ticket and contact data fields containing personal data. The anonymization action is performed for the data fields which have this annotation.

i The annotation needs to be assigned to data fields of simple data types, i.e., numbers, strings, enums, and MLAs. It cannot be used for lists and structs.

The **ConSol CM** API has been enhanced to provide methods for anonymizing contacts. Two methods have been added to the class `UnitGDPRService` for this purpose:

- `anonymize()`
- `anonymizeAsynchronously()`

In addition, the `Ticket` and `Unit` classes have been enhanced with two methods:

- `getAnonymizationDate()` to retrieve the date when the contact was anonymized
- `isAnonymized()` to check whether the contact was anonymized

The new value “anonymize” has been added to the variable `deleteType`. It means that the executed action is *Anonymize*, and can be used in contact actions scripts for delete actions.

During the anonymization, the contact fields holding personal data (as defined by the annotation) are deleted along with the contact history, comments, and attachments. Afterwards, an entry that the contact has been anonymized (*Data was anonymized*) is added to the contact history. The tickets, where the anonymized contact is the main customer, are also anonymized (ticket fields with the annotation are deleted, the complete history is deleted, and an entry that the ticket’s customer has been anonymized is added to the ticket history). The ticket subject is changed to the ticket name. Further relations of the contact to other objects (tickets where the contact is an additional customer, relations to resources and other customers) are removed, and the history of the affected objects shows *Not available anymore*. An entry stating that the contact has been anonymized is added to the log file for documentation purposes.



Note that it is still possible to edit an anonymized contact.



Please take into account the following specifics when working with the anonymization feature:

- The anonymization of the contact in the history of a scripted autocomplete field in other objects does not work if the history data were imported into **ConSol CM** using an import of runtime data.
- Contact anonymization is handled as a regular update in terms of business event triggers and contact update actions. This means that the contact update action is executed, and business event triggers which react on changes to fields containing personal data fire.
- Only tickets where the contact is the main customer are anonymized. This means that personal data remains in the CM system if it is located in tickets where the contact is now an additional contact, even if the contact was the main contact before. This is the case for example when a customer creates a ticket by sending an email, and another contact is set as the main customer later on.



A DWH update is required for this change. The update scripts for each database are available from your ConSol representative. Please execute the update script for your database engine.

8.3 Changes

This section illustrates all relevant software product changes for version 6.11.1.7 of **ConSol CM**.

8.3.1 Usability improvements

German localization of REST API messages added (#633451)

The error messages which can be returned by the REST API of **ConSol CM** have been translated into German because some of the messages might be displayed in **CM/Track**.

Line wrapping of Activity Control Form descriptions (#633546)

The layout has been adapted so that the description text of Activity Control Forms wraps in case it is longer than one line.

8.3.2 New API methods related to ticket closing date (#627989)

The **ConSol CM** API has been extended with two new methods which improve handling of the ticket closing data in scripts. The following two methods have been added to the **ticket** object:

- **isClosed()**
Returns whether a ticket is closed or not.
- **getCloseDate()**
Returns the date when the ticket was closed.

8.3.3 Whitelist for links in incoming emails (#632591)

Links in incoming emails are deactivated by default in the **Web Client** due to security reasons. This means that the link text is displayed in the ticket history but the link cannot be clicked.

The new system property **cmas-core-server, ticket.from.incoming.message.accepted.links** has been added to the *System Properties* section of the **Admin Tool** to allow the administrator to define a whitelist of domains for which these links should not be deactivated. This applies to links in incoming emails and to links in comments added via REST API.



This property does not apply to links in comments added via **CM/Track** or ETL.

Regular expressions can be used to specify the allowed URLs. It is possible to add several URLs by using a whitespace as delimiter.

Examples:

- **http://consol.de**
Allows links to **http://consol.de**
- **https://cm6\consol.de**
Allows links to **https://cm6.consol.de**
- **https://.*\google\..***
Allows links to **https://<any>.google.<any>**
- **.+**
Allows all URLs

If the property is empty, no links are allowed (previous behavior).



Please note that whitelisting domains might make **ConSol CM** vulnerable to cross-scripting and other attacks. Choose the domains you whitelist carefully!

8.3.4 Logging of optimistic locking exceptions improved (#633415)

Optimistic locking exceptions caused by the workflow engine are no longer logged with stacktraces on the “INFO” level. As the system is prepared to handle optimistic locking failures, they do not constitute a problem.

8.3.5 Content of pContext variable after queue change (#633441)

The content of the `pContext` variable as available in autocomplete scripts has been changed. When the user edits a ticket in the **Web Client**, changes its queue, and edits a scripted autocomplete field before saving the change of the queue, the `pContext` variable now contains the new queue and its fields. Previously, the old queue was available.

8.3.6 Automatic ticket update for Webhooks (#633444)

When updating a ticket using a Webhook integration script, it was necessary to manually trigger the ticket update event by using `ticketService.update(targetTicket)`, in order to avoid exceptions. Now, this update happens automatically and it is no longer necessary to manually invoke a ticket update.

8.3.7 Content of pContext variable in actions changed (#633475)

The content of the `pContext` variable as available in autocomplete scripts has been changed. When the user executes a customer or resource action in the **Web Client**, edits a data field within the action, and then edits a scripted autocomplete field, the `pContext` variable now contains the new value for the data field. Previously, the old value was available.

8.3.8 Forwarding of manual emails configurable (#633499)

ConSol CM includes a representation functionality which allows an engineer to select another engineer who should represent him when he is absent. Previously, all emails which were manually sent from the **Web Client** to the represented engineer were forwarded to the representing engineer by default.

This might not be desired in some use cases where the same person is an engineer and a customer in **ConSol CM**. Therefore, this behavior is now configurable. The system property `forward.mails.to.representatives` in the module `cmweb-server-adapter` has been introduced for this purpose. If it is set to “false”, manual emails are never forwarded to representatives. The default value of the property is “false”, meaning that the previous behavior is changed.



The handling of automatically sent emails has not been changed. It depends on the used Java method if they are forwarded or not.

8.4 Bugs fixed

The following defects have been corrected in this **ConSol CM** version.

Number	Description
632915	Exception after concurrent modification of favorites When a user removed a ticket from the favorites in one Web Client session and tried to access the same ticket from the favorites in a concurrent second session, an exception could occur. This problem has been solved and such an exception does not occur anymore.
632953	Exception when clicking <i>No results</i> for time booking When a user tried to add a time booking on the user profile page of the Web Client , clicked <i>No results</i> in the ticket selector, and clicked <i>Cancel</i> , an exception was written to the logs. This problem has been solved and such an exception does not occur anymore.
633352	Exception when logging in to the Web Client Sometimes, an exception could occur when the user tried to log in to the Web Client in several tabs. This problem has been solved and this exception does not occur anymore during login.
633411	Passwords displayed in clear text when filtering system properties The values of system properties of the type <i>Password</i> in the Admin Tool were not masqueraded when the list of system properties was filtered by module. Masquerading only worked for the unfiltered list of system properties. This problem has been solved and passwords are now always masqueraded in the Admin Tool .
633414	Setting the Track user via API did not work with enabled voCache Setting the Track user of a contact using the ConSol CM API, for example in a contact action, did not work properly when voCache was enabled (property <code>cmweb-server-adapter</code> , <code>voCacheEnabled</code>). This problem has been solved and the Track user is now updated correctly also when the voCache is enabled.
633457	Mobile layout broken on welcome page of CM/Track V2 The mobile layout of the welcome page of CM/Track V2 was not rendered as expected when only one queue box was displayed. This problem has been solved and the mobile layout now works properly also when only one queue box is present.
633463	Link text changed to “undefined” in the Web Client When adding a link in the Rich Text Editor of the Web Client , the link text was not saved correctly and changed into the text “undefined”. This problem has been solved and the text is now displayed correctly.
633495	Error when accessing deleted customer from the favorites When trying to access a customer which was deleted by another user from the favorites or adding a deleted customer to the favorites in the Web Client , an exception could occur. This problem has been solved and the corresponding error messages are now displayed instead.

Number	Description
633498	Error in the logs when deleting a customer in workflow activity When a customer was deleted in a workflow activity within a ticket which was deleted together with the customer, errors were written to the log files although the deletion worked correctly. This problem has been solved and an entry with log level "INFO" is now written to the log files instead.
633525	Enum values could not be selected in text template The values for enum parameters in a newly created text template could not be selected in the comment or email editor in the Web Client . This problem has been solved and the enum values can be selected now.
633530	Resources could not be deleted in DWH LIVE mode When the LIVE mode of the DWH was activated, it was not possible to delete resources which had relations to customers which were not marked as reportable. This problem has been solved and these resources can be deleted in DWH LIVE mode now.
633532	Script cache not cleared after scene import After importing a scene containing script changes in the Admin Tool , the cache was not cleared automatically resulting in the old version of the script being executed although the new version was displayed in the Admin Tool . This problem has been solved and script changes are correctly applied now.

9. Version 6.11.1.8 (06.09.2018)

Version 6.11.1.8 includes all **ConSol CM** changes and additions of

- 6.11.0 versions up to 6.11.0.6
- 6.10 versions up to 6.10.8.0
- 6.9 versions up to 6.9.4.7
- 6.8 versions up to 6.8.5.8

Newer versions and the respective changes are not included in this **ConSol CM** version.

9.1 Update and installation instructions

Please read all information on update and installation of **ConSol CM** specific for version 6.11.1.8 in this section before performing a system update or installation. Please consult the general product documentation on update and installation procedures as well as earlier version Release Notes, if applicable.

9.1.1 System upgrade from 6.11.0 and earlier versions

This **ConSol CM** release needs specific planning for an update from earlier releases. Please check the instructions for release 6.11.1.0 ([System upgrade from 6.11.0 and earlier versions](#)).

No further instructions available.

9.2 Changes

This section illustrates all relevant software product changes for version 6.11.1.0 of **ConSol CM**.

9.2.1 Usability improvements

History entry localized (#633394)

The entry, which is written to the customer history in the **Web Client** when the **CM/Track** password of the customer is changed, is now localized into German.

9.2.2 Rendering for empty scripted visualization fields introduced for **CM/Track V2** (#633443)

In **ConSol CM** version 6.11.1.6, the possibility to render empty scripted visualization fields was introduced for the **Web Client** (see [Rendering of empty scripted visualization fields introduced \(#632892\)](#)).

The support of this feature has now been extended to **CM/Track V2**. If the annotation **visualize-when-empty** is assigned to a scripted visualization field in the **Admin Tool**, the visualized content is now rendered both in the **Web Client** and in **CM/Track V2** even if the field is empty.

9.3 Bugs fixed

The following defects have been corrected in this **ConSol CM** version.

Number	Description
631075	Default font size not applied correctly in the editor The editor used to draft templates, emails and comments in the Web Client did not apply the default font size correctly to all pieces of text. If the font size of part of the text was changed manually, the font size of the rest of the text was not saved correctly. This problem has been solved and the text is saved with the correct font size now.
633421	Font size changed during validation A validation triggered when trying to save a comment or email could cause the font size of the first word of the text to be different than the rest of the text upon saving. During editing, the correct font size was displayed. This problem has been solved and the text is saved with the correct font size now.
633717	Exception due to missing label A missing label caused an exception when trying to transfer data or delete a contact which belongs to a one-level customer data model with the option <i>Company optional</i> checked. It was not possible to transfer the contact's data or delete the contact. This problem has been solved and the exception does not occur anymore.
633798	Unnecessary compiling of scripts Workflow and task scripts were compiled before each execution which could cause performance problems. This problem has been solved and such scripts are now kept in cache after compiling.

10. Version 6.11.1.9 (12.10.2018)

Version 6.11.1.9 includes all **ConSol CM** changes and additions of

- 6.11.0 versions up to 6.11.0.6
- 6.10 versions up to 6.10.8.0
- 6.9 versions up to 6.9.4.7
- 6.8 versions up to 6.8.5.8

Newer versions and the respective changes are not included in this **ConSol CM** version.

10.1 Update and installation instructions

Please read all information on update and installation of **ConSol CM** specific for version 6.11.1.9 in this section before performing a system update or installation. Please consult the general product documentation on update and installation procedures as well as earlier version Release Notes, if applicable.

10.1.1 System upgrade from 6.11.0 and earlier versions

This **ConSol CM** release needs specific planning for an update from earlier releases. Please check the instructions for release 6.11.1.0 ([System upgrade from 6.11.0 and earlier versions](#)).

10.1.2 Configuration modification requirements

The changes in this release require several configuration items to be reviewed and possibly adjusted. Please see the following sections for details:

- [Configurable date format in CM/Track V2 \(#633393\)](#)

No further instructions available.

10.2 Changes

This section illustrates all relevant software product changes for version 6.11.1.9 of **ConSol CM**.

10.2.1 Usability improvements

Line breaks in localized ACF descriptions (#633710)

Localized ACF descriptions in the **Web Client** can now include line breaks. This improves the readability of the descriptions.

Effective use of screen size when hiding empty groups in the ticket list (#633911)

When using the page customization option *hideEmptyGroups* for the ticket list of the **Web Client**, the screen space is now used efficiently when grouping the ticket list by ticket field. The ticket list uses all available vertical screen space even when some groups are hidden.

Configurable field size for scripted autocomplete fields (#633920)

The annotation `fieldsize` now works for scripted autocomplete fields in the **Web Client**. It can be assigned to the data field in the **Admin Tool** in order to change the size of the field, e.g., to make it larger.

10.2.2 Performance improvements

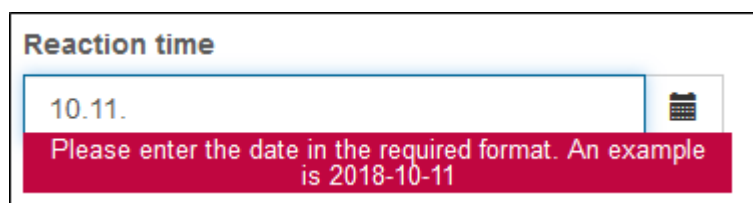
Performance of scenario import improved (#633769)

The performance of the import of scenes using the **Admin Tool** has been improved.

10.2.3 Configurable date format in **CM/Track V2** (#633393)

The annotations `format` and `accuracy` used to define the format for data fields of the type DATE are now applied to date fields in **CM/Track V2**. Previously, they were only applied to date fields in the **Web Client**.

If the date format is defined using the annotation `format`, the format is validated when the user types a date in **CM/Track V2**. If the entered date does not conform to the defined format, an error message indicating the current date in the required format is displayed.

A screenshot of a web form titled "Reaction time". It contains a text input field with the value "10.11.". To the right of the input field is a small calendar icon. Below the input field, a red error message box is displayed with the text: "Please enter the date in the required format. An example is 2018-10-11".

If no `format` annotation is set, the standard German date format (dd.MM.yyyy) is used as a default.

The annotation `accuracy` works in the same way as in the **Web Client**.

10.2.4 Shortcut to jump to the Quick Search (#633819)

The shortcut "F" to jump to the Quick Search has been added to the **Web Client**. This shortcut is available by default, but it can be disabled in the page customization. The new page customization attribute *globalSearchFocusKeyBindingEnabled* has been added to the scope *cmApplicationCustomization* for this purpose. Set its value to "false" to disable the shortcut.

10.3 Bugs fixed

The following defects have been corrected in this **ConSol CM** version.

Number	Description
633410	Error when opening an element in the workspace using the Recently visited link An error could occur when clicking the link in the <i>Recently visited</i> widget to open a ticket, customer or resource which was saved in the workspace. This problem has been solved and all elements in the workspace can now be opened using the Recently visited link.
633526	Resource actions not clickable after adding a relation Resource actions on the resource page of the Web Client were not clickable after adding a relation to a customer or ticket. It was necessary to reload the page after adding the relation. This problem has been solved and actions can now be executed directly after adding a relation.
633592	Image not displayed in the ticket history in CM/Track V2 In some cases, images included in comments were not displayed in the ticket history in CM/Track V2 . This problem has been solved and the images are now displayed correctly.
633649	Workspace did not work properly after using clientGoToResource() An exception occurred when a resource was moved to the workspace after executing an action whose script includes the method <code>clientGoToResource()</code> . This problem has been solved and such an exception does not occur anymore.
633552	Hyphens in CSS styles in incoming email prevented opening the editor If the history of a ticket in the Web Client contained an email with two hyphens (--) in the CSS styles, it was not possible to open the Email and Comment Editor of the ticket. This problem has been solved and such CSS styles do not prevent opening the editor anymore.
633680	Preview of PDF attachments and images did not work when the ticket was created from an email without body content When a ticket was created from an incoming email with no body content but only one or more PDF or image attachments, the ticket preview in the Detailed Search of the Web Client did not work as expected. It was either empty or the attachments were not displayed correctly. This problem has been solved and the PDF and image preview now works as expected for this kind of ticket.
633704	Engineer who added comments to a resource could not be deleted It was not possible to delete an engineer in the Admin Tool if this engineer had added comments to resources in the Web Client . This problem has been solved and the engineer can now be deleted in such a case.

Number	Description
633720	Activity conditions not reloaded after change to ticket The condition scripts of the workflow activities which are available for a ticket in the Web Client were not applied correctly after a change to the ticket until the page was reloaded. This could result in the wrong activities being displayed if the ticket change had an impact on the result of the condition script. This problem has been solved and the activity conditions are now applied correctly even if the page is not reloaded.
633780	Password reset via email not possible for CM/Track V2 Customers were not able to reset their passwords using the <i>Change password</i> link in CM/Track V2 . The link to reset the password, which was sent in the email, did not work as expected. This problem has been solved and resetting the password using the link is now possible.
633793	Incorrect model update after changing company After changing the company of a contact in the Web Client , the model was not updated correctly. This resulted in the new company not being available in templates. This problem has been solved and the company is now updated correctly.
633856	Font changed during validation when sending an email A validation triggered when trying to send an email from the Web Client could cause the font type and size of the email text, which were set using the Rich Text Editor, to be lost, so that the email would be sent with the default font type and size, although the text was displayed with the correct font type and size during editing. This problem has been solved and the email is now sent with the correct font type and size.

11. Version 6.11.1.10 (17.10.2018)

Version 6.11.1.10 includes all **ConSol CM** changes and additions of

- 6.11.0 versions up to 6.11.0.6
- 6.10 versions up to 6.10.8.0
- 6.9 versions up to 6.9.4.7
- 6.8 versions up to 6.8.5.8

Newer versions and the respective changes are not included in this **ConSol CM** version.

11.1 Update and installation instructions

Please read all information on update and installation of **ConSol CM** specific for version 6.11.1.10 in this section before performing a system update or installation. Please consult the general product documentation on update and installation procedures as well as earlier version Release Notes, if applicable.

11.1.1 System upgrade from 6.11.0 and earlier versions

This **ConSol CM** release needs specific planning for an update from earlier releases. Please check the instructions for release 6.11.1.0 ([System upgrade from 6.11.0 and earlier versions](#)).

No further instructions available.

11.2 Bugs fixed

The following defects have been corrected in this **ConSol CM** version.

Number	Description
634074	Images not displayed correctly in the ticket history in <i>CM/Track V2</i> If a ticket history entry in <i>CM/Track V2</i> contained more than one image, only the first image was displayed correctly. The other images were either missing or broken. This problem has been solved and all the images are displayed correctly now.

12. Version 6.11.1.11 (26.10.2018)

Version 6.11.1.11 includes all **ConSol CM** changes and additions of

- 6.11.0 versions up to 6.11.0.6
- 6.10 versions up to 6.10.8.0
- 6.9 versions up to 6.9.4.7
- 6.8 versions up to 6.8.5.8

Newer versions and the respective changes are not included in this **ConSol CM** version.

12.1 Update and installation instructions

Please read all information on update and installation of **ConSol CM** specific for version 6.11.1.11 in this section before performing a system update or installation. Please consult the general product documentation on update and installation procedures as well as earlier version Release Notes, if applicable.

12.1.1 System upgrade from 6.11.0 and earlier versions

This **ConSol CM** release needs specific planning for an update from earlier releases. Please check the instructions for release 6.11.1.0 ([System upgrade from 6.11.0 and earlier versions](#)).

No further instructions available.

12.2 Bugs fixed

The following defects have been corrected in this **ConSol CM** version.

Number	Description
634016	Message displayed twice after executing an ACF in the <i>Web Client</i> Messages displayed using the <code>client.showMessage()</code> method in activity scripts executed after confirming an ACF could be displayed twice in some cases. This problem has been solved and these messages are only displayed once now.
634131	Resources folder not sufficiently protected against misuse The mechanism to provide custom files using the <code>resources</code> folder, which was implemented for scripted field visualization, was not sufficiently protected against misuse. This problem has been solved.

13. Version 6.11.1.12 (21.12.2018)

Version 6.11.1.12 includes all **ConSol CM** changes and additions of

- 6.11.0 versions up to 6.11.0.6
- 6.10 versions up to 6.10.8.0
- 6.9 versions up to 6.9.4.7
- 6.8 versions up to 6.8.5.8

Newer versions and the respective changes are not included in this **ConSol CM** version.

13.1 Update and installation instructions

Please read all information on update and installation of **ConSol CM** specific for version 6.11.1.12 in this section before performing a system update or installation. Please consult the general product documentation on update and installation procedures as well as earlier version Release Notes, if applicable.

13.1.1 System upgrade from 6.11.0 and earlier versions

This **ConSol CM** release needs specific planning for an update from earlier releases. Please check the instructions for release 6.11.1.0 ([System upgrade from 6.11.0 and earlier versions](#)).

13.1.2 REST API client modification requirements

The following REST API enhancements may require to adjust clients accessing **ConSol CM** by using the REST API. The changes might offer a potential for client optimization.

- [Localized description of ACFs added to the REST API and CM/Track \(#634170\)](#)

13.1.3 Script modification requirements

This release features some interface changes in API methods. **Please be aware that scripts using the respective methods may need adjustments!** The affected methods are listed in the following sections of this document:

- [Unit ID available in autocomplete scripts in customer forms \(#633820\)](#)
- [Support for sending emails to many recipients \(#634174\)](#)



Please be aware that some methods were deprecated. Please ensure before an update that the scripts in your installation are not affected by these removals!

13.1.4 Configuration modification requirements

The changes in this release require several configuration items to be reviewed and possibly adjusted. Please see the following sections for details:

- [Precedence of visibility annotations changed \(#633871\)](#)

No further instructions available.

13.2 New features

The features newly introduced in this **ConSol CM** version 6.11.1.12 are described in detail in this section.

13.2.1 Development mode for dashboard widgets (#633970)

Dashboard widgets in the **Web Client** can now be switched to a new development mode. In the development mode, a validation of the widget script is performed and any error messages are displayed on the **Web Client** GUI. This allows debugging the widget script with runtime data.

The development mode is available for widgets of the types *chart* and *table*. It can be activated in the page customization, which has been enhanced with the attribute *developmentMode* available in the scopes *chartWidget* and *tableWidget*. The value is set to “false” by default. Set it to “true” to activate the development mode for the widget. Then, a JSLint validation is performed on the widget script.

If the development mode is active, the widget is displayed with a red border and the text *Development mode* is shown in the upper left corner:



If there are errors in the widget script, the error messages are displayed instead of the widget:

Development mode
line 1, character 321 reason: 'falses' was used before it was defined.

```
{chart:{type:'column'},credits:{enabled:false},exporting:{url:null,buttons:{contextButton:{menuItems:null,onclick:function()
{this.print();}}}},series:[{name:'All',data:[127]},{name:'Own',data:[0]},{name:'Unassigned',data:[108]}],title:{text:'1st Level Open
tickets'},tooltip:{headerFormat:""},yAxis:{allowDecimals:falses}}
```



When the development mode is active, it is shown to all users and not limited to the administrator.

13.3 Changes

This section illustrates all relevant software product changes for version 6.11.1.12 of **ConSol CM**.

13.3.1 Unit ID available in autocomplete scripts in customer forms (#633820)

The unit ID is now available in the context of an autocomplete script which is called within a customer action form. It can be retrieved in the autocomplete script using the `pContext` variable:

- all customers: `pContext.getUnit().id`
- contacts: `pContext.getContact().id`
- companies: `pContext.getCompany().id`

The retrieved ID allows to perform additional actions using the **ConSol CM** API.

13.3.2 Precedence of visibility annotations changed (#633871)

The behavior of the visibility annotations, which can be assigned to data fields in the **Admin Tool**, has been changed. Now, it is first checked if the data field should be visible according to the annotation `visibility`, if set. The annotation `visualize-when-empty` is only evaluated for data fields with a field visualization script if the field is visible in view mode.

13.3.3 Home screen of the **Admin Tool** adapted (#634166)

The home screen of the **Admin Tool** has been adapted for security reasons. The Java VM version of the **ConSol CM** server is now displayed only after the administrator has logged in to the **Admin Tool**.

13.3.4 Localized description of ACFs added to the REST API and **CM/Track** (#634170)

The **ConSol CM** REST API has been extended to provide the localized descriptions of ACFs. They are now included in the response when retrieving an ACF using the `activityformfieldssets/{name}` endpoint.

Therefore, localized ACF descriptions are now displayed in **CM/Track** as well.

13.3.5 Support for sending emails to many recipients (#634174)

The support for sending emails to many recipients using the **Web Client** or a script has been improved. Previously, an error could occur when the recipients' names contained special characters. Now, special characters are allowed in names, but not in the email addresses.

The method `getChunkedAttribute()` has been added to the class `ContentEntry` to set the recipient when sending emails using a script. The following example shows how to set the recipients in the `To` attribute for the email object `mail`:

```
mail.getChunkedAttribute('To')
```

The previously available method `getAttribute()` of the class `ContentEntry` is now deprecated and should not be used anymore.

13.4 Bugs fixed

The following defects have been corrected in this **ConSol CM** version.

Number	Description
633496	Attachment selector not displayed in basic visibility setting The attachment selector, which is available in the Web Client when writing an email for a ticket with attachments, was not displayed if the visibility of the ticket history was set to <i>Basic</i> . This problem has been solved and the attachment selector is now displayed in all visibility settings.
633694	Remember me option did not work The <i>Remember me</i> option which is available on the login page of the Web Client did not work properly. Although the user checked this option, a new session was not created automatically after a session timeout occurred. This problem has been solved and the creation of new sessions after a session timeout now works as expected.
634036	Exception when deploying a workflow which deletes an ACF prefill script An exception could occur when deploying a workflow in which an ACF prefill script, which was used by more than one ACF, was deleted. This problem has been solved and such a workflow can be deployed now.
634135	Paging did not work in task scripts Paging for processing data in a task script did not work properly resulting in the first page being loaded recursively. This problem has been solved and paging can now be used in task scripts.
634190	Incorrect date displayed in the Attachments section The <i>Attachments</i> section of the Web Client showed an incorrect date for attachments which were imported from a scene. In this case, the date of the scene import was displayed instead of the date when the attachment was uploaded. This problem has been solved and the upload date from the original scene is now used.
634215	Ticket list not refreshed when ticket is open in the edit mode When a ticket is opened in the edit mode in the Web Client , the ticket list was not refreshed. This problem has been solved and the ticket list is now refreshed while editing a ticket.

14. Version 6.11.1.13 (18.01.2019)

Version 6.11.1.13 includes all **ConSol CM** changes and additions of

- 6.11.0 versions up to 6.11.0.6
- 6.10 versions up to 6.10.8.0
- 6.9 versions up to 6.9.4.7
- 6.8 versions up to 6.8.5.8

Newer versions and the respective changes are not included in this **ConSol CM** version.

14.1 Update and installation instructions

Please read all information on update and installation of **ConSol CM** specific for version 6.11.1.13 in this section before performing a system update or installation. Please consult the general product documentation on update and installation procedures as well as earlier version Release Notes, if applicable.

14.1.1 System upgrade from 6.11.0 and earlier versions

This **ConSol CM** release needs specific planning for an update from earlier releases. Please check the instructions for release 6.11.1.0 ([System upgrade from 6.11.0 and earlier versions](#)).

No further instructions available.

14.2 Bugs fixed

The following defects have been corrected in this **ConSol CM** version.

Number	Description
634304	Links with line breaks in incoming emails cannot be clicked in the ticket history If an incoming email in the ticket history of the Web Client included a link which contained a line break, the link could not be clicked. This problem has been solved and such links are clickable now.
634468	Labels of empty data fields visible The label of a data field which is annotated with <code>visibility = "view"</code> was visible in the view mode even if the field was empty. This problem has been solved and labels are not visible in this case now.

15. Version 6.11.1.14 (11.03.2019)

Version 6.11.1.14 includes all **ConSol CM** changes and additions of

- 6.11.0 versions up to 6.11.0.6
- 6.10 versions up to 6.10.8.0
- 6.9 versions up to 6.9.4.7
- 6.8 versions up to 6.8.5.8

Newer versions and the respective changes are not included in this **ConSol CM** version.

15.1 Update and installation instructions

Please read all information on update and installation of **ConSol CM** specific for version 6.11.1.14 in this section before performing a system update or installation. Please consult the general product documentation on update and installation procedures as well as earlier version Release Notes, if applicable.

15.1.1 System upgrade from 6.11.0 and earlier versions

This **ConSol CM** release needs specific planning for an update from earlier releases. Please check the instructions for release 6.11.1.0 ([System upgrade from 6.11.0 and earlier versions](#)).

15.1.2 Configuration modification requirements

The changes in this release require several configuration items to be reviewed and possibly adjusted. Please see the following sections for details:

- [Script cache invalidation enhanced \(#634409\)](#)

No further instructions available.

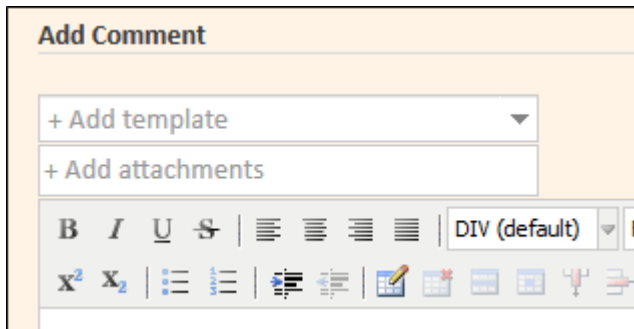
15.2 Changes

This section illustrates all relevant software product changes for version 6.11.1.14 of **ConSol CM**.

15.2.1 Usability improvements

Styling of the attachments selector improved (#633492)

The styling of the attachments selector, which is displayed in the **Web Client** when writing a comment or email for a ticket, has been improved. It now matches the template selector regarding the initial width and the plus icon.

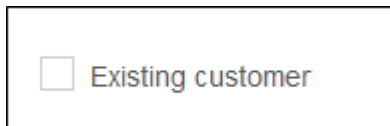


Required date fields (#634240)

In **CM/Track V2**, required fields of the type *date* are now marked with an asterisk symbol (*) which indicates that the field is required.

Display of BOOLEAN data fields in **CM/Track V2** improved (#634541)

The display of data fields of the type *boolean* in **CM/Track V2** has been improved. It now works in the same way as in the **Web Client**, i.e., there is a label and a checkbox which can be selected.

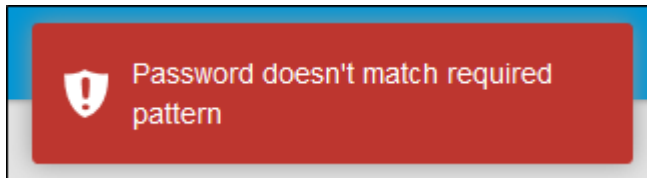


15.2.2 Remember filter settings on customer pages (#633523)

The selected filters (*Show* and *Status*) in the *Tickets* section of contact and company pages in the **Web Client** are now saved to the user's preferences for each customer group. They are applied automatically when the user opens a contact or company page.

15.2.3 Error handling regarding password change in **CM/Track V2** improved (#634305)

When the user tries to change his password for **CM/Track V2** to a password which does not match the required password pattern (determined by the system property `cmas-core-security.policy.password.pattern`), an error message informing the user about the problem is displayed and the user has to enter another new password.



15.2.4 Script cache invalidation enhanced (#634409)

It is now possible to change the time period during which unused scripts remain in the cache. The system property `cmas-core-server, script.evict.unused.after.hours` has been introduced for this purpose. The default value is "24" meaning that unused scripts remain in the cache for 24 hours. After this time, the compiled class of the script is removed. The **ConSol CM** server checks for scripts to evict every hour.

15.2.5 Logging for authentication exceptions added for the REST API (#634701)

The **ConSol CM** REST API has been enhanced with a more detailed logging of authentication exceptions. In the log levels WARN and DEBUG, messages as the following are written to the log files:

```
[exec] 2019-02-25 15:12:47,060 WARN [esource.AuthenticationResource] [--]  
Authentication exception: Incorrect credentials
```

15.3 Bugs fixed

The following defects have been corrected in this **ConSol CM** version.

Number	Description
634552	Special characters are escaped when removing an attachment When removing an attachment, whose name contains special characters, in the Web Client , the special characters were escaped in the context menu entry <i>Remove</i> . This problem has been solved and any special characters included in the file name are correctly displayed now.
634552	Content of scripted field visualization fields not escaped The content of a data field with the annotation visualization was not escaped in the Web Client . This problem has been solved and the field content is escaped now.
634690	Different tasks execute the same source code If two dynamic tasks with different source code were executed in the same millisecond in the same task group using the Task Execution Framework, both tasks executed the same source code. This problem has been solved and each task now executes its own source code.
634711	Resource type permissions not retrieved correctly Permissions to resource types were not retrieved correctly resulting in the menu item <i>Resource pool</i> not being available in some cases. This problem has been solved and the menu item is now always available if the user has permissions to at least one resource type.
634731	Performance problem for views with many queues The performance of the Web Client was slow when a view containing many queues was selected in the ticket list and the page customization attribute <i>hideEmptyGroups</i> was set to "true". This problem has been solved and the performance has been improved for this use case.

16. Version 6.11.1.15 (19.07.2019)

Version 6.11.1.15 includes all **ConSol CM** changes and additions of

- 6.11.0 versions up to 6.11.0.6
- 6.10 versions up to 6.10.8.0
- 6.9 versions up to 6.9.4.7
- 6.8 versions up to 6.8.5.8

Newer versions and the respective changes are not included in this **ConSol CM** version.

16.1 Update and installation instructions

Please read all information on update and installation of **ConSol CM** specific for version 6.11.1.15 in this section before performing a system update or installation. Please consult the general product documentation on update and installation procedures as well as earlier version Release Notes, if applicable.

16.1.1 System upgrade from 6.11.0 and earlier versions

This **ConSol CM** release needs specific planning for an update from earlier releases. Please check the instructions for release 6.11.1.0 ([System upgrade from 6.11.0 and earlier versions](#)).

No further instructions available.

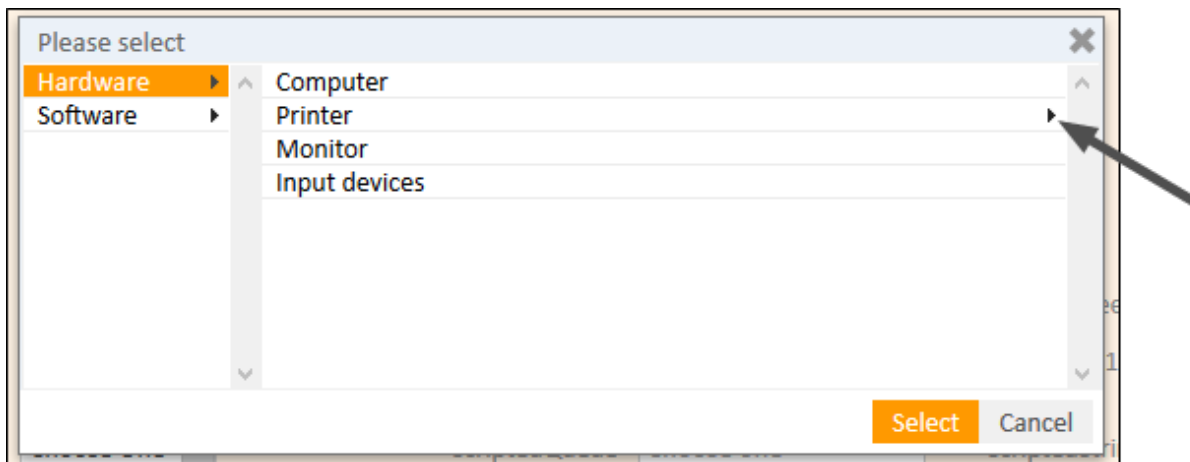
16.2 Changes

This section illustrates all relevant software product changes for version 6.11.1.15 of **ConSol CM**.

16.2.1 Usability improvements

Arrow to indicate MLA levels added (#632329)

The little arrow which is displayed to indicate that an MLA entry has sublevel entries has been added again.



Consistency of date formats in **CM/Track V2** improved (#632697)

The consistency of the displayed dates in **CM/Track V2** has been improved. The date format used for the ticket history now matches the date format used in the basic ticket data and the ticket fields. It depends on the browser locale which date format is used.

16.2.2 Wildcard behavior changed for customer and engineer auto-complete search (#634405)

A wildcard is now applied automatically after entering one character in the autocomplete fields to select a customer or engineer in the **Web Client**. In this way, the user already sees the suggestions for the first character of each search term he enters.

16.2.3 Handling of required fields in **CM/Track V2** improved (#634751)

The annotation **required** now works for all types of data fields in **CM/Track V2** in the same way as in the **Web Client**. An asterisk is displayed next to a required field and a validation message is shown if the user tries to save without providing a value. Previously, the annotation did not work for data fields of the types BOOLEAN, LIST, MLA or STRUCT in **CM/Track V2** and the asterisk was also missing for the subject and comment fields.

16.3 Bugs fixed

The following defects have been corrected in this **ConSol CM** version.

Number	Description
632748	<p>Unnecessary whitespace at the beginning of CM/Track configuration file</p> <p>The example configuration for CM/Track V2 in the Admin Tool (section <i>Clients -> Configurations</i>) contained a whitespace at the beginning of the config.json file, which prevented the configuration from loading. This whitespace has been removed.</p>
634508	<p>Resource page of a resource which is related to a contact without company could not be opened</p> <p>If the last settings for the table which shows the related customers of a resource on a resource page contained the <i>Company</i> column, it was not possible to open the resource page of any resource which is related to a contact without company. This problem has been solved and the resource page can be opened in this case now.</p>
634756	<p>Changes to company not visible in ticket</p> <p>When the VO cache is enabled and company data is changed, the changes were not visible in the company's tickets until the VO cache was refreshed. This problem has been solved and company changes are directly visible in tickets now.</p>
634984	<p>DWH transfer failed if a list contained a field with a name including "field"</p> <p>The transfer of data to the DWH failed if a field of the type LIST contained a field whose name included the word "field". This problem has been solved and the transfer to the DWH does not fail anymore in this situation.</p>
634995	<p>Session timeout in the Admin Tool did not work in DWH LIVE or ADMIN mode</p> <p>If the DWH was running in the modes LIVE or ADMIN, session timeouts in the Admin Tool did not work correctly. It could happen that no session timeout occurred. This problem has been solved and session timeouts work correctly now.</p>
635296	<p>CM/Track credentials cleared when changing another annotation</p> <p>When changing any annotation of a customer field which contains the user name or password for CM/Track (annotations contact authentication, username or contact authentication, password), the values of the field were deleted from the ConSol CM database. This problem has been solved and field values are only cleared now when changing the contact authentication annotations.</p>
635444	<p>Broken layout when assigning an additional customer to a ticket in the Web Client</p> <p>In some cases, the layout of the <i>Customers</i> section of the ticket was broken when adding an additional customer to the ticket. The <i>Cancel</i> button was missing and there was no orange background color in the edit mode. This problem has been solved and the section is displayed correctly now.</p>

17. Version 6.11.1.16 (27.09.2019)

Version 6.11.1.16 includes all **ConSol CM** changes and additions of

- 6.11.0 versions up to 6.11.0.6
- 6.10 versions up to 6.10.8.0
- 6.9 versions up to 6.9.4.7
- 6.8 versions up to 6.8.5.8

Newer versions and the respective changes are not included in this **ConSol CM** version.

17.1 Update and installation instructions

Please read all information on update and installation of **ConSol CM** specific for version 6.11.1.16 in this section before performing a system update or installation. Please consult the general product documentation on update and installation procedures as well as earlier version Release Notes, if applicable.

17.1.1 System upgrade from 6.11.0 and earlier versions

This **ConSol CM** release needs specific planning for an update from earlier releases. Please check the instructions for release 6.11.1.0 ([System upgrade from 6.11.0 and earlier versions](#)).

17.1.2 Data Warehouse update

When updating the ConSol CM system to version 6.11.1.16, the DWH needs to be updated due to the extension of the supported dates. The update scripts for each database are available from your ConSol representative. Please follow the instructions in [DWH adapted to support dates until 01.01.2100 \(#641871\)](#).

No further instructions available.

17.2 Changes


This section illustrates all relevant software product changes for version 6.11.1.16 of **ConSol CM**.

17.2.1 DWH adapted to support dates until 01.01.2100 (#641871)

The DWH table `dim_date` has been adapted to support dates until 01.01.2100. Previously, only dates until 01.01.2020 were supported.

The DWH database needs to be updated for this change. Please perform the following steps:

1. Before starting the **CMRF**, manually execute the database update script for your database engine. Please ask the ConSol CM support or your ConSol CM consultant for the update script.
2. Start the **ConSol CM** server. If the DWH is in LIVE or ADMIN mode and the **ConSol CM** system contains data fields with dates between 01.01.2020 and 01.01.2100, DWH tasks are created automatically to update these fields in the DWH.
3. Run the created DWH tasks in the section *Data Warehouse* -> *Tasks* of the **Admin Tool**.

 DWH tasks are also created for data fields which are not annotated as “reportable”. During the execution of the DWH tasks, these tasks are handled automatically.

18. Version 6.11.1.17 (23.11.2021)


Version 6.11.1.17 includes all **ConSol CM** changes and additions of

- 6.11.0 versions up to 6.11.0.6
- 6.10 versions up to 6.10.8.0
- 6.9 versions up to 6.9.4.7
- 6.8 versions up to 6.8.5.8

Newer versions and the respective changes are not included in this **ConSol CM** version.

18.1 Update and installation instructions

Please read all information on update and installation of **ConSol CM** specific for version 6.11.1.17 in this section before performing a system update or installation. Please consult the general product documentation on update and installation procedures as well as earlier version Release Notes, if applicable.

 The only change in this version with respect to the previous version 6.11.1.16 is that the certificate of the Java Web Start applications and Java applets has been renewed, please see [Web applications expiry](#).

18.1.1 System upgrade from 6.11.0 and earlier versions

This **ConSol CM** release needs specific planning for an update from earlier releases. Please check the instructions for release 6.11.1.0 ([System upgrade from 6.11.0 and earlier versions](#)).

No further instructions available.

19. Version 6.11.1.18 (10.03.2022)


Version 6.11.1.18 includes all **ConSol CM** changes and additions of

- 6.11.0 versions up to 6.11.0.6
- 6.10 versions up to 6.10.8.0
- 6.9 versions up to 6.9.4.7
- 6.8 versions up to 6.8.5.8

Newer versions and the respective changes are not included in this **ConSol CM** version.

19.1 Update and installation instructions

Please read all information on update and installation of **ConSol CM** specific for version 6.11.1.18 in this section before performing a system update or installation. Please consult the general product documentation on update and installation procedures as well as earlier version Release Notes, if applicable.

 The only change in this version with respect to the previous version 6.11.1.17 is that the certificate of **CM/Doc** has been renewed so that it matches the certificate of the other the Java Web Start applications and Java applets, please see [Web applications expiry](#).

19.1.1 System upgrade from 6.11.0 and earlier versions

This **ConSol CM** release needs specific planning for an update from earlier releases. Please check the instructions for release 6.11.1.0 ([System upgrade from 6.11.0 and earlier versions](#)).

No further instructions available.

20. Web applications expiry

ConSol CM includes several **Java Web Start** tools and **Java applets**. These technologies validate the downloaded programs by validating a certificate. This applies to all recent and supported versions of the Java platform at the time of writing. Any certificate for this purpose includes an expiry date after which it will not validate anymore. After this date a program containing this certificate will not be considered valid anymore as well and an update to a version with a newer certificate is required to have the tool operational.

The **ConSol CM** releases covered in this document include Java Web Start tools and Java applets with the following certificate expiry dates:

Release version(s)	Certificate expiry date
6.11.1.0 - 6.11.1.14	28.10.2019
6.11.1.15 - 6.11.1.16	08.03.2022
6.11.1.17 and higher	05.11.2024

The **Java Web Start** tools and **Java applets** of **ConSol CM** affected by this expiry date are these:

- **Admin Tool**
- **Process Designer**
- **CM/Doc**



Please make sure you always have a **ConSol CM** version with a valid certificate installed to guarantee uninterrupted usage of the **Java Web Start** tools and **Java applets**!