

DOCUMENT

Release Notes ConSol CM Version 6.10.3

Author: Thomas Schall
Phone: +49 (0) 89 / 45841-100
Mail: thomas.schall@consol.de
Version: 1.1
Date: 09.11.2015
Status: Released

Table of contents

General update and installation instructions.....	3
1 Version 6.10.3.0 (06.11.2015).....	4
1.1 Update and installation instructions	4
1.1.1 Data warehouse schema update necessary (#627716)	4
1.1.2 Data warehouse JMS communication channel not supported anymore	4
1.2 New Features	5
1.2.1 Resource Pool REST API (#625711, #626348)	5
1.2.2 Resource Pool support for external resources (#626591)	18
1.2.3 ACF design and layout improvements (#627892, #628118).....	24
1.2.4 Page customization for e-mail address field visibility (#627777).....	26
1.2.5 Wildcard usage page customization for Quick and Easy Search (#627894).....	28
1.2.6 E-Mail export to EML in Admin-Tool (#628251)	28
1.2.7 Supported Firefox browser version 38 ESR (#627830)	29
1.3 Changes.....	30
1.3.1 Performance improvements.....	30
1.3.2 Layout improvements	30
1.3.3 New result column for resource detail search based on template (#628175)	30
1.3.4 Column sorting and ordering in the customer relation tables (#626279, #627770)	31
1.3.5 Multiple table column settings for different resource types and customer groups (#628236)	31
1.3.6 Clone script engineer assignment modification (#628181)	31
1.3.7 DWH indicator for active tasks restored in Admin-Tool (#627629)	32
1.3.8 System property for enabling/disabling internationalization of static data warehouse tables (#627716).....	32
1.3.9 Server logging for mail sending made consistent (#620198)	33
1.3.10 Obsolete CM.Phone configuration file entries removed (#625957)	33
1.4 Bugs fixed.....	34
1.5 Known Issues	38

General update and installation instructions

For an update of ConSol[®] CM from one version to another two possible ways exist:

- **Distribution installation**
The distribution is installed into the application server. For an update every local configuration, like the data source configuration, has to be saved before and reconfigured afterwards.
This type of update ensures that really every change between the versions is installed. This type of update is recommended for updates of the major or minor version, e.g. for an update from 6.6.3 to 6.7.5.
- **EAR / WAR Update**
For this type of update of the ConSol[®] CM, the EAR (cm6.ear, cmrf.ear) and WAR (cm-track.war) files of the new version have to be installed into the application server. Additionally every installation related changes described in the chapters 'Update and installation instructions' have to be applied manually. The changes have to be applied for every version between your original CM version and the new CM version, e.g. for an update from 6.6.3 to 6.6.7 the instructions of the versions 6.6.5, 6.6.6 and 6.6.7 have to be checked.
This type of update is only recommended for updates within a minor version.

Additionally for every type of update, the 'Update and installations instructions' chapter has to be checked for further important notes.

If available, the solution specific Release Notes have to be checked too.

1 Version 6.10.3.0 (06.11.2015)

Version 6.10.3.0 includes

- 6.10.2 versions up to 6.10.2.1,
- 6.10.1 version 6.10.1.0,
- 6.10.0 version 6.10.0.0,
- 6.9 versions up to 6.9.4.5, and
- 6.8 versions up to 6.8.5.8.

1.1 Update and installation instructions

1.1.1 Data warehouse schema update necessary (#627716)

Changes in the context of the data warehouse static tables internationalization (see also sections 1.3.1 and 1.3.8) have been introduced which make it necessary to modify the database schema of the data warehouse (DWH) database. Update scripts have been provided. Please request those update scripts by contacting the ConSol* CM Support Team.

When using JMS transfer mode previously, please switch to direct mode in advance since the JMS communication channel is not supported anymore (see section 1.1.2 for more details).

Please execute the update script of 6.10.3.0 after the upgrading the ear-file to the version 6.10.3.0 **but before** CMRF (or CM6 with CMRF in a single server) is started.

The script must be executed directly on the DWH database.

1.1.2 Data warehouse JMS communication channel not supported anymore

ConSol* CM does not support the JMS data warehouse communication channel anymore in version 6.9.4 or newer, including this version 6.10.3.0 release. This has been stated explicitly for the JBoss 7 (JBoss EAP 6) application server platform when introducing support for this platform. However, it should be clarified that this also applies to the other supported platforms JBoss 5 and Oracle Weblogic 11g R1. When using a ConSol* CM version 6.9.4 or newer **ONLY** the DIRECT mode data warehouse communication channel is available. Please implement this change when updating from earlier CM6 versions using the JMS communication.

No further instructions available.

1.2 New Features

1.2.1 Resource Pool REST API (#625711, #626348)

The REST API functionality of ConSol[®] CM has been extended to be able to handle resources. This includes all the objects that are necessary to handle the resource pool and the resources themselves. Below the functionality is shown with examples.

Resource Groups

Get all resource groups

```
GET /resourcegroups
```

Example document:

```
curl -u Huber:consol http://localhost:8888/restapi/resourcegroups -H "Accept: application/xml"

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<resourcegroups>
  <resourcegroup
    uri="http://localhost:8888/restapi/resourcegroups/CmCustomer?v=c-P1MTQSpiZx-v-wfV2Yag%3D%3D" name="CmCustomer">
    <orderIndex>0</orderIndex>
  </resourcegroup>
</resourcegroups>
```

Get resource group by name

```
GET /resourcegroups/{name}
```

Example document:

```
curl -u Huber:consol http://localhost:8888/restapi/resourcegroups/CmCustomer -H "Accept: application/xml"

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<resourcegroup
  uri="http://localhost:8888/restapi/resourcegroups/CmCustomer?v=c-P1MTQSpiZx-v-wfV2Yag%3D%3D" name="CmCustomer">
  <enabled>true</enabled>
  <orderIndex>0</orderIndex>
  <resourcetypes>
    <resourcetype uri="http://localhost:8888/restapi/resourcetypes/company?v=c-P1MTQSpiZx-v-wfV2Yag%3D%3D" name="company">
      <orderIndex>1</orderIndex>
    </resourcetype>
    <resourcetype
      uri="http://localhost:8888/restapi/resourcetypes/customer?v=c-P1MTQSpiZx-v-wfV2Yag%3D%3D" name="customer">
      <orderIndex>0</orderIndex>
    </resourcetype>
  </resourcetypes>
</resourcegroup>
```

Resource Types

Get all resource types

```
GET /resourcetypes
```

Example document:

```
curl -u Huber:consol http://localhost:8888/restapi/resourcetypes -H "Accept: application/xml"

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<resourcetypes>
<resource uri="http://localhost:8888/restapi/resourcetypes/company?v=c-PlMTQSpiZx-v-wfV2Yag%3D%3D" name="company">
  <orderIndex>1</orderIndex>
</resource>
<resource uri="http://localhost:8888/restapi/resourcetypes/customer?v=c-PlMTQSpiZx-v-wfV2Yag%3D%3D" name="customer">
  <orderIndex>0</orderIndex>
</resource>
</resourcetypes>
```

Get resource type by name

```
GET /resourcetypes/{name}
```

Example document:

```
curl -u Huber:consol http://localhost:8888/restapi/resourcetypes/customer -H "Accept: application/xml"

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<resource uri="http://localhost:8888/restapi/resourcetypes/customer?v=c-PlMTQSpiZx-v-wfV2Yag%3D%3D" name="customer">
  <enabled>true</enabled>
  <resourceGroup uri="http://localhost:8888/restapi/resourcegroups/CmCustomer?v=c-PlMTQSpiZx-v-wfV2Yag%3D%3D"/>
  <accessMode>INTERNAL</accessMode>
  <orderIndex>0</orderIndex>
  <groups>
    <group uri="http://localhost:8888/restapi/definitions/groups/customerr?v=6Msvdj4jmlqE7MfRDCC3JQ%3D%3D" name="customerr">
      <enabled>true</enabled>
      <sortIndex>0</sortIndex>
    </group>
  </groups>
</resource>
```

Resource and its custom fields

Create a resource

```
POST /resources
```

Example document:

```
curl -u Huber:consol -d "resourceType=customer&customerr.salutationr=mr"
http://localhost:8888/restapi/resources

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<location>
  <uri>http://localhost:8888/restapi/resources/33391</uri>
</location>
```

Get resource by ID

```
GET /resources/{id}
```

Example document:

```
curl -u Huber:consol -d "resourceType=customer&customerr.salutationr=mr"
http://localhost:8888/restapi/resources

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<resource id="33391" uri="http://localhost:8888/restapi/resources/33391">
  <mark>Some text here</mark>
  <resourceType
uri="http://localhost:8888/restapi/resourcetypes/customer?v=OXYBdpgSNKVT82Ndz5A
JnQ%3D%3D"/>
    <accessModeDate>0</accessModeDate>
    <modificationDate>1420631379214</modificationDate>
    <groups>
      <group name="customerr">
        <definition
uri="http://localhost:8888/restapi/definitions/groups/customerr?v=HzQG6EvuPNth5
WgwEFlw5Q%3D%3D"/>
          <fields>
            <field xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="enumFieldElement" class="ENUM"
              name="salutationr" group="customerr">
              <name>salutationr</name>
              <uuid>363b6c69-9663-11e4-87e3-c7ab70def792</uuid>
              <value xmlns:xs="http://www.w3.org/2001/XMLSchema"
xsi:type="xs:string">Mr</value>
              <originalValue>mr</originalValue>
            </field>
          </fields>
        </group>
      </groups>
    </resource>
```

Get resource by criteria

```
GET /resources
```

Example document:

```
curl -u Huber:consol http://localhost:8888/restapi/resources -H "Accept: application/xml"
curl -u Huber:consol http://localhost:8888/restapi/resources?resourceType=customer -H "Accept: application/xml"
curl -u Huber:consol http://localhost:8888/restapi/resources?resourceType=company -H "Accept: application/xml"
curl -u Huber:consol http://localhost:8888/restapi/resources?customerr.salutationr=mr -H "Accept: application/xml"
curl -u Huber:consol http://localhost:8888/restapi/resources?modifiedBefore=21.10.2014 -H "Accept: application/xml"

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<resources>
  <lastPageNumber>0</lastPageNumber>
  <pageNumber>0</pageNumber>
  <pageSize>1</pageSize>
  <totalNumberOfElements>1</totalNumberOfElements>
  <resource uri="http://localhost:8888/restapi/resources/33391" id="33391"/>
</resources>
```

Possible parameters:

- pattern,
- resourceType,
- resourceGroup,
- modifiedBefore,
- modifiedAfter,
- accessBefore,
- accessAfter,
- order,
- orderDesc,
- maxId,
- minId,
- ids,
- excludeIds,
- resourceTypeIds
- resourceGroupIds
- active
- permission
- resourceRelations

All date parameters use the format: dd.MM.yyyy

Available permission parameter values:

- READ,
- WRITE,
- CREATE,
- DELETE,
- CONTENT_READ,
- CONTENT_WRITE,
- CONTENT_DELETE,
- ACT,
- DISABLE

resourceRelations parameter examples:

```
http://Huber:consol@localhost:8888/restapi/resources?resourceRelations=[{"occurrence":"SHOULD","endpoint":"ANY","resourceId":"160"}]
```

```
curl -G -u admin:consol "http://localhost:8888/restapi/resources" --data-urlencode "resourceRelations=[{"occurrence":"SHOULD","endpoint":"ANY","resourceId":"160"}]"
curl -G -u admin:consol "http://localhost:8888/restapi/resources?resourceRelations=[{"occurrence":"SHOULD","endpoint":"ANY","resourceId":"160"}]"
```

The value of that parameter must be a JSON object. It has to contain a subset of those parameters:

- resourceId,
- ticketId,
- unitId,
- externalId,
- occurrence,
- endpoint,
- resourceTypeId,
- definitionAllAnonymous,
- definitionId

The user can search for one of four different cases:

- by resourceId - required parameters: occurrence, endpoint, resourceId
- by ticketId - required parameters: occurrence, ticketId
- by externalId - required parameters: occurrence, externalId, resourceTypeId
- by unitId - required parameters: occurrence, unitId

Possible values of parameters:

- occurrence: MUST, SHOULD
- endpoint: ANY, SOURCE, TARGET

The parameter definitionId is an ID of the ResourceRelationDefinition.

Update Resource

```
PUT /resources/{id}
```

Example document:

```
curl -u Huber:consol -X PUT -d "active=false"
http://localhost:8888/restapi/resources/33391

HTTP/1.1 200 OK
```

Resource comments and attachments

Create a resource comment

```
POST /resources/{id}/comments
```

Example document:

```
curl -u Huber:consol -X POST -d "comment=text"
http://localhost:8888/restapi/resources/33391/comments

HTTP/1.1 200 OK
```

Get a resource comments – get all or get comments by criteria (comment text, date, engineer)

```
GET /resources/{id}/comments
```

Example document:

```
curl -u Huber:consol http://localhost:8888/restapi/resources/33391/comments -H
"Accept: application/xml"

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<comments>
  <lastPageNumber>0</lastPageNumber>
  <pageNumber>0</pageNumber>
  <pageSize>1</pageSize>
  <totalNumberOfElements>1</totalNumberOfElements>
  <comment creationDate="2015-01-07T14:42:51.972+01:00"
    uri="http://localhost:8888/restapi/resources/33391/comments/33409">
    <active>true</active>
    <engineer uri="http://localhost:8888/restapi/engineers/Huber?v=0"/>
    <text>text</text>
  </comment>
</comments>
```

Possible parameters:

- comment,
- date,
- engineer,
- order,
- orderDesc

Delete a resource comment

```
DELETE /resources/{id}/comments/{id}
```

Example document:

```
curl -u Huber:consol -X DELETE
http://localhost:8888/restapi/resources/53805/comments/53898

HTTP/1.1 200 OK
```

Create a resource attachment

```
POST /resources/{id}/attachments
```

Example document:

```
curl -u Huber:consol -X POST -F "file=@pom.xml" -F "description=some desc"
http://localhost:8888/restapi/resources/33391/attachments

HTTP/1.1 200 OK
```

Get resource's attachments - get all or get attachments by criteria (description, date, engineer)

```
GET /resources/{id}/attachments
```

Example document:

```
curl -u Huber:consol http://localhost:8888/restapi/resources/53805/attachments
-H "Accept: application/xml"
curl -u Huber:consol
http://localhost:8888/restapi/resources/53805/attachments?engineer=Huber -H
"Accept: application/xml"

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<attachments>
  <lastPageNumber>0</lastPageNumber>
  <pageNumber>0</pageNumber>
  <pageSize>1</pageSize>
  <totalNumberOfElements>1</totalNumberOfElements>
  <attachment
file="http://localhost:8888/restapi/resources/33391/attachments/33413/pom.xml"
    creationDate="2015-01-07T15:36:35.285+01:00"

uri="http://localhost:8888/restapi/resources/33391/attachments/33413">
    <active>true</active>
```

```
<engineer uri="http://localhost:8888/restapi/engineers/Huber?v=0"/>
<description>some desc</description>
<filename>pom.xml</filename>
<mimeType>application/xml</mimeType>
<size>37748</size>
</attachment>
</attachments>
```

Possible parameters:

- description,
- date,
- engineer,
- order,
- orderDesc

Get the attachment data

```
GET /resources/{id}/attachments/{id}/{filename}
```

Example document:

```
curl -u Huber:consol
http://localhost:8888/restapi/resources/33391/attachments/33413/pom.xml
```

Delete a resource's attachment

```
DELETE /resources/{id}/attachments/{id}
```

Example document:

```
curl -u Huber:consol -X DELETE
http://localhost:8888/restapi/resources/53805/attachments/53928

HTTP/1.1 200 OK
```

Resource relations configuration

Get all relation definitons

```
GET resources/relations/definitions
```

Example document:

```
curl -u Huber:consol
http://localhost:8888/restapi/resources/relations/definitions -H "Accept:
application/xml"

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<resourcerelationdefinitions>
  <resourcerelationdefinition transferKey="1b7d4748-73d9-11e4-85c4-
75c8c7d3164a" id="32793">
```

```

    <name>ResourceTicket</name>
    <orderIndex>626</orderIndex>
  </resourcerelationdefinition>
  <resourcerelationdefinition transferKey="cd4f9476-73ef-11e4-914c-
b98d2364ad2b" id="32818">
    <name>related</name>
    <orderIndex>949</orderIndex>
  </resourcerelationdefinition>
</resourcerelationdefinitions>

```

Get relation definitions by id

```
GET resources/relations/definitions/{id}
```

Example document:

```

curl -u Huber:consol
http://localhost:8888/restapi/resources/relations/definitions/32818 -H "Accept:
application/xml"

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<resourcerelationdefinition transferKey="cd4f9476-73ef-11e4-914c-b98d2364ad2b"
id="32818">

  uri="http://localhost:8888/restapi/resources/relations/definitions/32818">
    <name>related</name>
    <enabled>true</enabled>
    <orderIndex>949</orderIndex>
    <sourceResourceType
uri="http://localhost:8888/restapi/resourcetypes/customer?v=OXYBdpgSNKVT82Ndz5A
JnQ%3D%3D"/>
    <multiplicity>ONE_TO_ONE</multiplicity>
    <reportable>false</reportable>
    <editable>true</editable>
    <hasComment>true</hasComment>
    <targetQueues>
      <queue
uri="http://localhost:8888/restapi/queues/HelpDesk_1st_Level?v=c-P1MTQSpiZx-v-
wFV2Yag%3D%3D"
        name="HelpDesk_1st_Level">
          <enabled>true</enabled>
          <name>HelpDesk1st_Level</name>
        </queue>
      </targetQueues>
    </resourcerelationdefinition>

```

Get relation definitions by criteria

```
GET resources/relations/definitions
```

Example document:

```

curl -u Huber:consol
http://localhost:8888/restapi/resources/relations/definitions -H "Accept:
application/xml"
curl -u Huber:consol
http://admin:consol@localhost:8888/restapi/resources/relations/definitions?sour

```

```
ceResourceTypesNames=client&targetResourceTypesNames=server
curl -u admin:consol
http://admin:consol@localhost:8888/restapi/resources/relations/definitions?relationDefinitionType=RESOURCE_RESOURCE
curl -u Huber:consol
http://admin:consol@localhost:8888/restapi/resources/relations/definitions?definitionsIds=14,1

<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<resourcerelationdefinitions>
  <resourcerelationdefinition transferKey="7e96189d-b2b8-11e4-a452-8ddf7f876c22" id="1">
    <name>related</name>
    <orderIndex>1</orderIndex>
  </resourcerelationdefinition>
  <resourcerelationdefinition transferKey="d4ca5541-b6ab-11e4-b8dc-870bc13598eb" id="14">
    <name>related</name>
    <orderIndex>31</orderIndex>
  </resourcerelationdefinition>
</resourcerelationdefinitions>
```

Possible parameters:

- definitionsIds,
- definitionsNames,
- sourceResourceTypesNames,
- sourceResourceTypesIds,
- targetResourceTypesNames,
- targetResourceTypesIds,
- targetQueuesNames,
- targetQueuesIds,
- targetCustomerGroupsNames,
- targetCustomerGroupsIds,
- targetUnitType,
- relationDefinitionType,
- enabled

Possible values for parameter relationDefinitionType:

- ALL,
- RESOURCE_RESOURCE,
- RESOURCE_TICKET,
- RESOURCE_UNIT

Possible values for parameter targetUnitType:

- CUSTOMER,
- CONTACT,
- COMPANY

Resource-resource relations

Create resource-resource relation

Form parameters:

- `resourceId` - target resource
- `comment` - (optional) comment
- `definitionId` - (optional) relation definition, if not provided system will try to find the proper definition
- `definitionTransferKey` - (optional) same as `definitionId`
- `resourceType` - (optional) resource type - *required for external resources*
- `externalId` - (optional) external id - *required for external resources*

```
POST /resources/{id}/resources
```

Example document:

```
curl -u Huber:consol -X POST -d "resourceId=33550&comment=some desc"
http://localhost:8888/restapi/resources/33545/resources

HTTP/1.1 200 OK
```

Get all resource-resource relations

```
GET /resources/{id}/resources
```

Example document:

```
curl -u Huber:consol http://localhost:8888/restapi/resources/33545/resources -H
"Accept: application/xml"

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<resourceresourcerelations>
  <resourceresourcerelation>
    <id>33558</id>
    <targetResource uri="http://localhost:8888/restapi/resources/33558"/>
    <relationDefinition
uri="http://localhost:8888/restapi/resources/relations/definitions/33495"/>
      <sourceResource uri="http://localhost:8888/restapi/resources/33550"/>
      <comment>some desc</comment>
    </resourceresourcerelation>
  </resourceresourcerelations>
```

Delete resource-resource relation

```
DELETE /resources/{id}/resources/{relationId}
```

Example document:

```
curl -X DELETE -u Huber:consol  
http://localhost:8888/restapi/resources/33545/resources/33558  
  
HTTP/1.1 200 OK
```

Resource-ticket relations

Create resource-ticket relation

```
POST /tickets/{name}/resources
```

Example document:

```
curl -u Huber:consol -X POST -d "resourceId=33005&comment=some desc"  
http://localhost:8888/restapi/tickets/SUP-91/resources  
curl -u Huber:consol -X POST -d "resourceId=33005&comment=some  
desc&definitionId=32818" http://localhost:8888/restapi/tickets/SUP-91/resources  
  
HTTP/1.1 200 OK
```

Get all resource-ticket relations

```
GET /tickets/{name}/resources
```

Example document:

```
curl -u Huber:consol http://localhost:8888/restapi/tickets/SUP-91/resources -H  
"Accept: application/xml"  
  
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>  
<ticketresourcerelations>  
  <ticketresourcerelation>  
    <id>33439</id>  
    <targetTicket uri="http://localhost:8888/restapi/tickets/SUP-91"/>  
    <relationDefinition  
uri="http://localhost:8888/restapi/resources/relations/definitions/32818"/>  
      <sourceResource uri="http://localhost:8888/restapi/resources/33005"/>  
        <comment>some desc</comment>  
    </ticketresourcerelation>  
</ticketresourcerelations>
```


Delete resource-ticket relation

```
DELETE /tickets/{name}/resources/{relationId}
```

Example document:

```
curl -X DELETE -u Huber:consol http://localhost:8888/restapi/tickets/SUP-91/resources/33181  
  
HTTP/1.1 200 OK
```

Resource-unit relations

Create resource-unit relation

```
POST /units/{id}/resources
```

Example document:

```
curl -u Huber:consol -X POST -d "resourceId=33550&comment=some desc&definitionId=33583" http://localhost:8888/restapi/units/4004/resources  
  
HTTP/1.1 200 OK
```

Get all resource-unit relations

```
GET /units/{id}/resources
```

Example document:

```
curl -u Huber:consol http://localhost:8888/restapi/units/4004/resources -H "Accept: application/xml"  
  
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>  
<unitresourcerelations>  
  <unitresourcerelation>  
    <id>33597</id>  
    <targetUnit uri="http://localhost:8888/restapi/units/33597"/>  
    <relationDefinition  
uri="http://localhost:8888/restapi/resources/relations/definitions/33583"/>  
      <sourceResource uri="http://localhost:8888/restapi/resources/33550"/>  
      <comment>some desc</comment>  
    </unitresourcerelation>  
</unitresourcerelations>
```

Delete resource-unit relation

```
DELETE /units/{id}/resources/{relationId}
```

Example document:

```
curl -X DELETE -u Huber:consol  
"http://localhost:8888/restapi/units/38376/resources/49485"  
  
HTTP/1.1 200 OK
```

1.2.2 Resource Pool support for external resources (#626591)

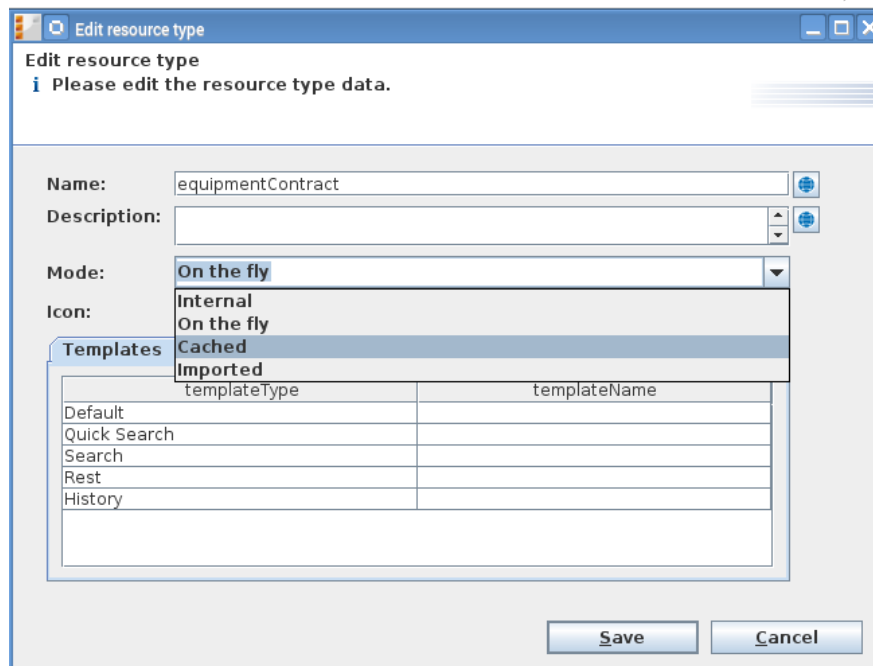
This release features the support for resource data from external sources, shorthand external resources. The external sources to query from ConSol[®] CM may vary and thus the necessary interface must be implemented in a custom CM6 project for several modes. Therefore a custom CM6 project is a prerequisite for using the external resources feature (except for the imported resources mode).

There are three types (modes) of external resources:

- **On-the-fly resources** which are queried from the external source every time they are accessed.
- **Cached resources** which are queried on the first access and then cached in CM6.
- **Imported resources** which are bulk imported into the system and stored in CM6.

This resource mode selection has to be made when modeling the respective resource in the Admin-Tool. The mode cannot be changed anymore when the resource type is in use, i.e. there are actual resource data or relations present. For all these modes it is necessary to create the data model in the Admin-Tool before using them. On-the-fly resources and cached resources also must have the interface implemented and the custom CM6 project built with the interface, before they can be used. The data of the items for imported resources must be bulk imported after the model is created.

The configuration of external resources in the Admin-Tool generally is the same as for internal resources, only with the mode properly set to the correct value. The Web client also relies on this mode setting for displaying the functionalities available for a specific resource.



Edit resource type
Please edit the resource type data.

Name: equipmentContract

Description:

Mode: On the fly

Icon:

Templates

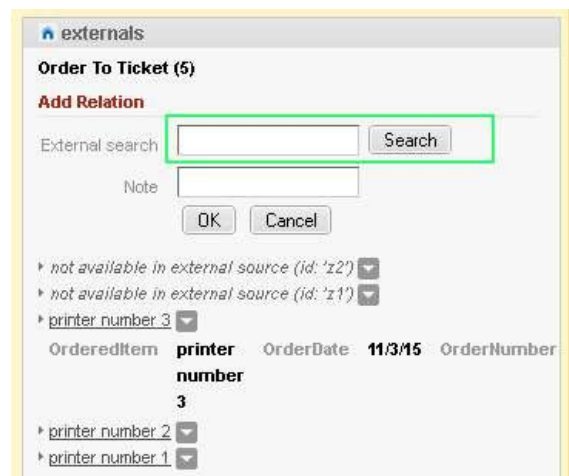
templateType	templateName
Default	
Quick Search	
Search	
Rest	
History	

Save Cancel

None of the external resources can be created from inside ConSol* CM Web Client. Cached and imported resources can be modified in CM6, but such a change cannot be propagated back to the source system. These changes are lost when the data fields are updated from the source system.

On-the-fly resources

On-the-fly resources are volatile to the system. There is no internal data representation of them in the database except for the external ID information. They do not have a detail page and the only place to employ them is in a relation to tickets, data objects or other resources. This external ID also is the only information by which they can be searched for when putting them in a relation. The data fields of such an on-the-fly resource are only displayed in the relations section it is actually used. Therefore, no "Jump to resource" link is shown for such a resource in its context menu there. There is no detail search inside ConSol* CM for on-the-fly-resources.



externals

Order To Ticket (5)

Add Relation

External search Search

Note

OK Cancel

not available in external source (id: 'z2')

not available in external source (id: 'z1')

printer number 3

OrderedItem printer OrderDate 11/3/15 OrderNumber number 3

printer number 2

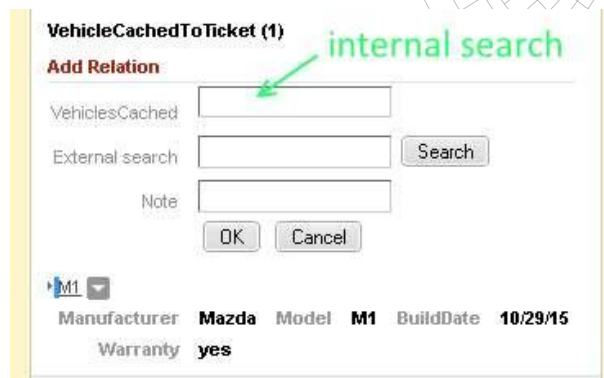
printer number 1

For using the mode of on-the-fly resources it is necessary to implement the interface in a custom project to access the external source.

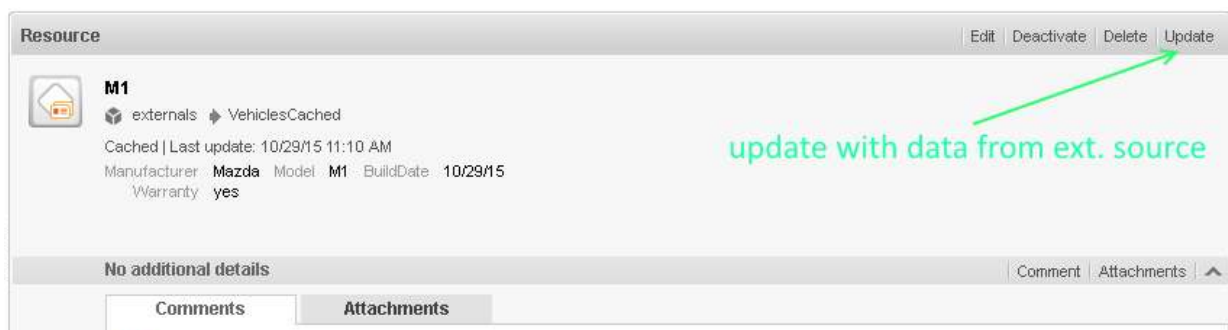
Cached resources

Cached resources are queried from the source system the first time they are used in CM6. They then are stored (cached) for further use. Currently this only is possible when creating a relation with a specific external resource as the target. For a later release it is planned to provide an independent (detail) search and import for external resources to be cached without

immediately using them in a relation. Originally these external resources are not in the CM6 database and the full data set most probably is only present in the external source system. An external resource which is not yet cached in CM6 can only be found in the external system by using the external ID field. The ones already cached can also be found by an internal search as highlighted in the screenshot to the right.



For those resources already cached a detail page is available. These resources can be updated from the source system on demand for each individual resource. The Update can be initiated from the “Update”-link in the tile bar on the detail page



For the mode of cached resources it also is necessary to implement the interface in a custom project to access the external source.

Imported resources

Imported resources in principle work like internal resources. The main difference is that the data are imported, usually by an ETL job, and the resource items can only be created by such an import job. It is not possible to create them in the Web Client even with sufficient privileges. Once an imported resource item is available in the system it can be used and changed like an internal resource. However, a later re-import may revert changes.

Imported resources are the only type of external resources that does not require an interface implementation and a custom CM6 project. However the data import must be available to provide resource items to use in the system.

Service Interface for the binding to an external source

This interface is necessary to access the external source for on-the-fly and cached external resources. In order to integrate it with ConSol* CM a custom CM6 project is mandatory. The interface is called *ResourceExternalSource* and it must be provided in the following directory:

```
web/modules/server/addon/src/main/java/com/consol/cmweb/server/addon/service/impl
```

The interface class has three methods to be implemented:

```
PageResult<Resource> searchByPattern(ResourceType pType, String pPattern, int
pPageSize, int pPageNumber);

Resource importResource(Resource pResource);

Resource getByExternalId(ResourceType pType, String pExternalId);
```

- **searchByPattern:** The method should return the set of search results for the detail search in CM6. Currently it must be implemented, but an empty implementation will suffice, since search and import independent from relation creation are only available in a later release (see above in the section “Cached resources”). Once this search functionality is available working implementation must be present for cached resources. For on-the-fly resources the implementation can generally be empty.
- **importResource:** In two cases this method is being used for cached resources: When creating a relation that targets this resource identified by its external ID. In this case it is required to verify in the CM6 system, if this external resource has not been imported and cached before. The other case the method is used happens when updating a cached resource with current data from the external source. The implementation should care for finding and updating the identified resource in the CM6 system. This method can have an empty implementation for on-the-fly resources since for these no import is done.
- **getByExternalID:** This method is called when the user executes the external ID search while creating a relation which uses the external resource as target. This external ID must be unique to unambiguously identify one single item in the external system. It is assumed that it will return no more than one resource. The method must have a fully functional implementation both for on-the-fly and cached resources.

Below you can find a sample implementation of the interface as a test interface which does not access a real external system. However, this implementation also creates test resources.

```
package com.consol.cmweb.server.addon.service.impl;

import java.util.Date;
import java.util.HashMap;
import java.util.List;
import java.util.Map;

import org.springframework.beans.BeansException;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.context.ApplicationContext;
import org.springframework.context.ApplicationContextAware;
import org.springframework.stereotype.Service;
import org.springframework.transaction.annotation.Transactional;

import com.consol.cmas.common.model.resource.Resource;
import com.consol.cmas.common.model.resource.meta.ResourceType;
import com.consol.cmas.common.model.util.CustomFieldsDeepCopy;
import com.consol.cmas.common.service.resource.ResourceExternalSource;
import com.consol.cmas.common.service.resource.ResourceService;
import com.consol.cmas.common.service.resource.ResourceTypeService;
import com.consol.cmas.common.util.paging.ListPageResult;
import com.consol.cmas.common.util.paging.PageResult;
```

```

/**
 * This file should be copied to
 "web/modules/server/addon/src/main/java/com/consol/cmweb/server/addon/service/impl"
 */

@Transactional
@Service
public class ResourceExternalSourceImpl implements ResourceExternalSource,
ApplicationContextAware {

    private ApplicationContext applicationContext;
    @Autowired private ResourceTypeService resourceTypeService;
    // for special case: update cacheable resource with data from the external source
    private String updateableField = "updateme";
    private String updateableExtId = "ForUpdate";

    @Override
    public PageResult<Resource> searchByPattern(ResourceType pType, String pPattern, int
pPageSize, int pPageNumber) {
        Map<String, Resource> resources = createExternalResources();

        List<Resource> resourceslist = (List<Resource>) resources.values();

        return new ListPageResult<Resource>(pPageSize, pPageNumber, resourceslist.size(),
resourceslist);
    }

    @Override
    public Resource importResource(Resource pResource) {
        // if ID is nonzero it means we got the pResource from CM so
        // we want to update the resource from the details page
        // in this case we need to fetch fresh date from the external source
        if (pResource.getId() != 0) {
            pResource = getByExternalId(pResource.getResourceType(), pResource.getExternalId());
        }
        pResource.setModificationDate(System.currentTimeMillis());
        ResourceService resourceService = (ResourceService)
applicationContext.getBeansOfType(ResourceService.class).values().iterator().next();
        Resource r = resourceService.getByExternalId(pResource.getResourceType(),
pResource.getExternalId());
        if (r == null) {
            return resourceService.create(pResource);
        } else {
            CustomFieldsDeepCopy.deepCopy(pResource, r, false);
            r.setModificationDate(System.currentTimeMillis());
            resourceService.update(r);
            return r;
        }
    }

    @Override
    public Resource getByExternalId(ResourceType pType, String pExternalId) {
        Map<String, Resource> resources = createExternalResources();

        // special case 1: delayed response - provide external id in format:
        // "delayed:[DELAY TIME IN ms]:externalId"
        if (pExternalId.startsWith("delayed") && (pExternalId.split(":").length == 3)) {
            try {
                Thread.sleep(Integer.parseInt(pExternalId.split(":")[1]));
            } catch (Exception e) { }
        }
        Resource r = resources.get(pExternalId);
    }

```

```

        if (r != null && r.getResourceType().getName().equalsIgnoreCase(pType.getName())) {
            return r;
        }
        return null;
    }

    private Map<String, Resource> createExternalResources() {
        Map<String, Resource> resources = new HashMap<String, Resource>();
        Resource resource;
        String extIdPrefix = "c";
        // create some Cached resources "VehiclesCached"
        ResourceType rType = resourceTypeService.getByName("VehiclesCached");
        for (int i = 0; i < 20; i++) {
            resource = new Resource(rType);
            resource.set("VehiclesCachedCFG.Manufacturer", "Mazda");
            resource.set("VehiclesCachedCFG.model", "M" + i);
            resource.set("VehiclesCachedCFG.BuildDate", new Date());
            resource.set("VehiclesCachedCFG.Warranty", i/2==0);
            resource.setExternalId(extIdPrefix + i);
            resources.put(extIdPrefix + i, resource);
        }

        resource = new Resource(rType);
        resource.set("VehiclesCachedCFG.Manufacturer", "Lexus");
        resource.set("VehiclesCachedCFG.model", updateableField);
        resource.set("VehiclesCachedCFG.BuildDate", new Date());
        resource.set("VehiclesCachedCFG.Warranty", false);
        resource.setExternalId(updateableExtId);
        resources.put(updateableExtId, resource);

        // create some OnTheFly resources "OrderHistoryOnTheFly"
        rType = resourceTypeService.getByName("OrderHistoryOnTheFly");
        extIdPrefix = "p";
        for (int i = 0; i < 20; i++) {
            resource = new Resource(rType);
            resource.set("OrderHistoryOnTheFlyCFG.OrderedItem", "printer number " + i);
            resource.set("OrderHistoryOnTheFlyCFG.OrderDate", new Date());
            resource.set("OrderHistoryOnTheFlyCFG.OrderNumber", i);
            resource.setExternalId(extIdPrefix + i);
            resources.put(extIdPrefix + i, resource);
        }

        // create some "delayed" resources
        resource = new Resource(rType);
        extIdPrefix = "delayed:1000:1";
        resource.set("OrderHistoryOnTheFlyCFG.OrderedItem", "printer delayed");
        resource.set("OrderHistoryOnTheFlyCFG.OrderDate", new Date());
        resource.set("OrderHistoryOnTheFlyCFG.OrderNumber", -1);
        resource.setExternalId(extIdPrefix);
        resources.put(extIdPrefix, resource);
        return resources;
    }

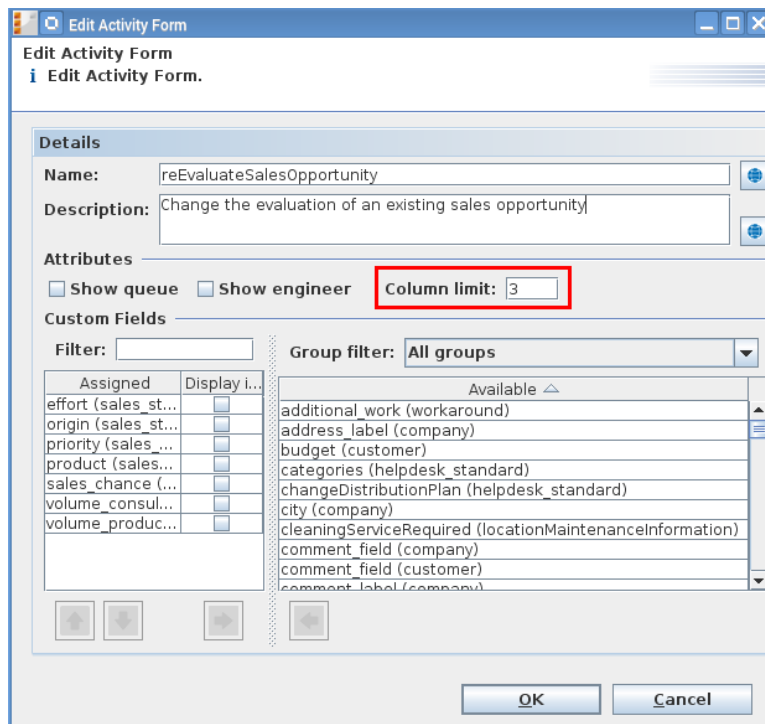
    @Override
    public void setApplicationContext(ApplicationContext pApplicationContext) throws
        BeansException {
        applicationContext = pApplicationContext;
    }
}

```


1.2.3 ACF design and layout improvements (#627892, #628118)

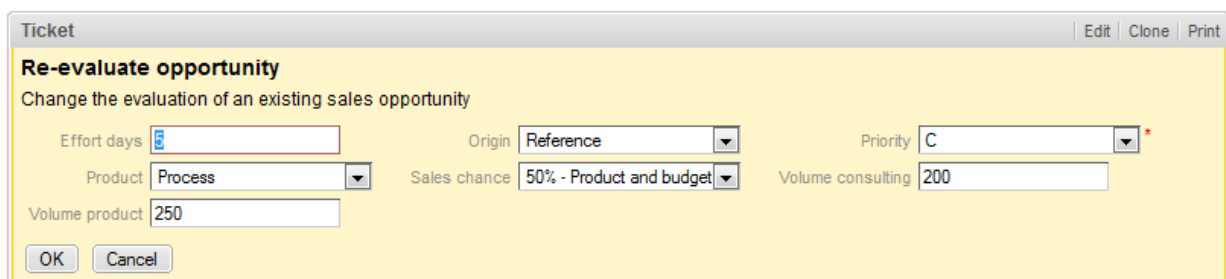
The possibilities to design and layout Activity Control Forms have been improved in several ways. Whereas earlier only the screen resolution decided, if a one column layout or a two column layout was chosen, this release offers easy configuration options to influence the general form design.

The Admin-Tool dialog to configure an ACF not offers to set the column limit. This is the number of columns to be used when rendering the form fields. It will apply independent of the screen resolution.



The screenshot shows the 'Edit Activity Form' dialog box. The 'Details' section contains 'Name: reEvaluateSalesOpportunity' and 'Description: Change the evaluation of an existing sales opportunity'. The 'Attributes' section has checkboxes for 'Show queue' and 'Show engineer', and a 'Column limit' field set to '3'. The 'Custom Fields' section shows a list of fields with checkboxes for 'Assigned' and 'Display i...'. The 'Group filter' is set to 'All groups'. The 'Available' list includes various fields like 'additional_work (workaround)', 'address_label (company)', 'budget (customer)', etc. The 'OK' and 'Cancel' buttons are at the bottom.

The example value of “3” as shown above will display the fields of the form in three columns as can be seen below.

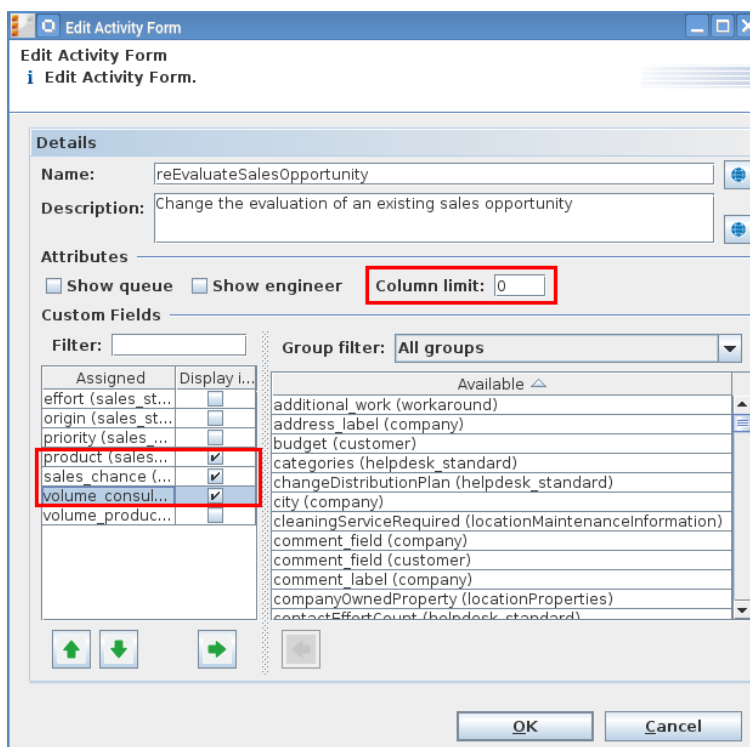


The screenshot shows the 'Ticket' form titled 'Re-evaluate opportunity'. The form displays fields for 'Effort days' (5), 'Product' (Process), 'Volume product' (250), 'Origin' (Reference), 'Sales chance' (50% - Product and budget), 'Priority' (C), and 'Volume consulting' (200). The 'OK' and 'Cancel' buttons are at the bottom.

Setting this value to “1” will show all fields in one single column putting one under the other.

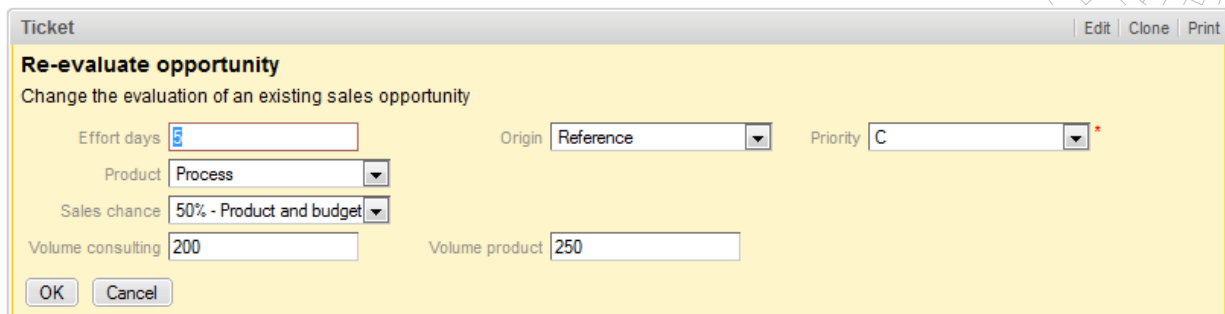


The value “0” has a special meaning. It does not set a column limit which means all fields would be shown next to each other in one single row. However, to reasonably control the layout in this case the property “Display in new row” for individual form fields can be set. Checking this property causes this field to be displayed in the first column of a new row, effectively allowing manual layouts with arbitrary column numbers. Such a configuration is illustrated in the next Admin-Tool screenshot.



Assigned	Display in new row
effort (sales_st...)	<input type="checkbox"/>
origin (sales_st...)	<input type="checkbox"/>
priority (sales ...)	<input type="checkbox"/>
product (sales...)	<input checked="" type="checkbox"/>
sales_chance (...)	<input checked="" type="checkbox"/>
volume_consul...	<input checked="" type="checkbox"/>
volume_produc...	<input type="checkbox"/>

The corresponding manual layout can be observed in the following screenshot of the Web Client which clearly shows the effect of the property “Display in new row” for a ACF field.



Ticket Edit Clone Print

Re-evaluate opportunity
Change the evaluation of an existing sales opportunity

Effort days Origin Priority

Product

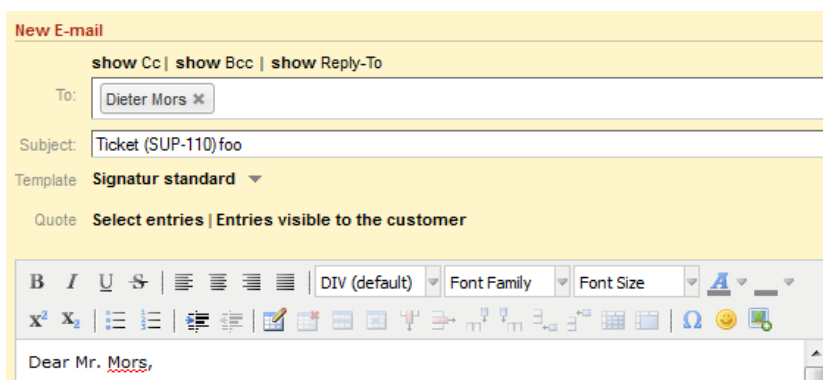
Sales chance

Volume consulting Volume product

OK Cancel

1.2.4 Page customization for e-mail address field visibility (#627777)

The address fields when writing e-mails shown by default were fixed until now. The *To:* and *Reply-to:* fields were displayed. By introducing a set of three new page customizations this is open to configuration now. This allows to model different usage scenarios so that no other addressees of the e-mail are visible for a regular recipient (or all of them).



New E-mail

show Cc | show Bcc | show Reply-To

To:

Subject:

Template: **Signatur standard** ▼

Quote: **Select entries** | Entries visible to the customer

Dear Mr. Mors,

The only address field generally shown now is the *To:* field. All other address fields are hidden by default now. They can be accessed by the user when clicking the corresponding “show”-link above the *To:* address field: “show Cc”, “show Bcc”, and “show Reply-To”.

showBcc	Show Bcc field.	<input type="text" value="false"/> (boolean) (+) (-)	
showCc	Show Cc field.	<input type="text" value="false"/> (boolean) (+) (-)	
showReplyTo	Show reply to field.	<input type="text" value="false"/> (boolean) (+) (-)	

In this standard case the default value *false* is used for all three new page customization attributes: *showBcc*, *showCc*, and *showReplyTo* which are defined in the top level scope *mailTemplate*.

showBcc	Show Bcc field.	false (boolean) (+) (-)	
showCc	Show Cc field.	true (boolean) (+) (-)	
showReplyTo	Show reply to field.	false (boolean) (+) (-)	

Each one of these three fields can be shown by default when opening the e-mail editor by setting the respective attribute value to *true*. Then the address field is displayed and the link to open it will not be rendered any more. The other fields are unaffected.

Setting all three attributes to *true* leads to all address fields to be shown immediately when the e-mail editor is opened.

showBcc	Show Bcc field.	true (boolean) (+) (-)	
showCc	Show Cc field.	true (boolean) (+) (-)	
showReplyTo	Show reply to field.	true (boolean) (+) (-)	

Additionally the validation of the e-mail addressees has been refined. Whereas it previously required at least one valid address in the *To:* field, it now is only required that there is one valid e-mail address in either the *To:*, the *Cc:*, or the *Bcc:* field.

1.2.5 Wildcard usage page customization for Quick and Easy Search (#627894)

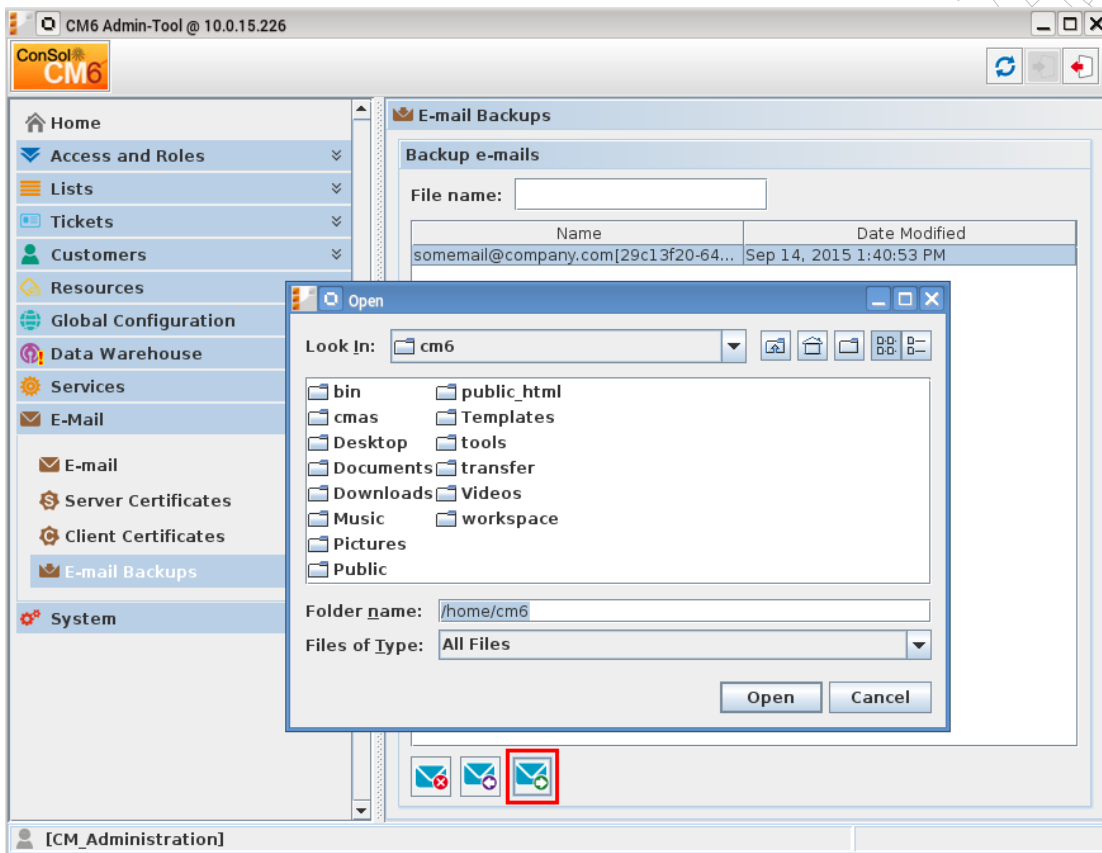
A new page customization has been introduced to control whether the wildcard “*” should be appended automatically to the search term of the Quick and Easy Search.

This has been done generally in the past, therefore the default now is to append the wildcard. Thus, no change in configuration is necessary to maintain the application behavior constant. The new customization attribute is called *appendWildcardAutomatically* and must be set in the top level scope *globalSearchField* so that it applies uniformly on every page. Setting the value to *false* changes the behavior, the wildcard will not be appended automatically to every search term, effectively narrowing down the result list. This way only exact matches for the term will be listed as result, others will be excluded.

Attribute name	Description	Value	
appendWildcardAutomatically	When enabled the wildcard is appended to every search phrase	<input type="text" value="true"/> (boolean) (+) (-)	<ul style="list-style-type: none"> cmApplicationCustomization navigationLinks welcomePage globalSearchField

1.2.6 E-Mail export to EML in Admin-Tool (#628251)

The individual e-mails listed under E-mail-Backup in the Admin-Tool are stored in the database when using NIMH for incoming e-mail processing. For this reason the mails were difficult to retrieve for detailed problem analysis and reproduction of errors. A new functionality has been introduced in the Admin-Tool to ease this kind of task. The e-mail message corresponding to an entry in the list on the page E-mail-Backup can now be saved as a standard EML file to disk. A third button has been added below the list marked in the screen shot below. This button is available when an entry in the list is selected, otherwise it is inactive. When clicking this button a standard directory chooser dialog appears to save the selected message as an EML file. The file name is the name shown in the list with the extension “.eml”.



The function also is available in the context menu of the selected list entry. Clicking the menu entry “Export selected E-Mail to EML” will have the same effect as clicking the button. The file then can be used for the above mentioned purposes for example.



1.2.7 Supported Firefox browser version 38 ESR (#627830)

Starting with CM release 6.10.3.0 version 38 of the Firefox browser Extended Support Release (ESR) is supported by CM6. The support for the previous ESR version 31 of Firefox is no longer actively maintained.

1.3 Changes

1.3.1 Performance improvements

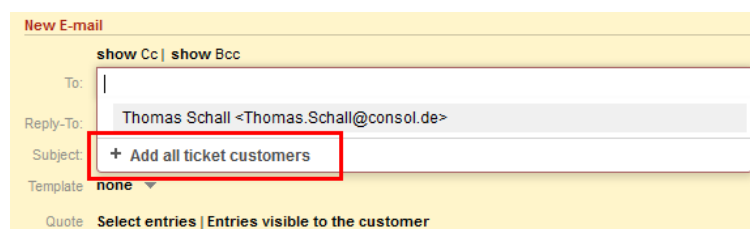
- **Ticket search improvements/performance restoration (#627005):** Some specific cases of ticket search degraded with addition of the resource pool. These cases have been improved so that the performance is now as expected again.
- **Optimization of DWH database update duration (#627716):** Internationalization for static data warehouse tables introduced changes which caused the database update during a DWH update to run significantly longer. This was mentioned in section 1.1.3 of the release notes for CM version 6.10.1.0. This release features optimizations to reduce this step's duration again. This internationalization feature can be turned off completely restoring the previous behavior. This is the default setting now. Please see section 1.3.8 for more information.

1.3.2 Layout improvements

- **Replacement of annotation icons in Admin-Tool (#627775):** The icons used on buttons for manipulating custom field annotations have been replaced by new, simpler ones with a more modern appearance which may be better to memorize. The same was done for annotations of data object fields and resource field and all corresponding field groups.

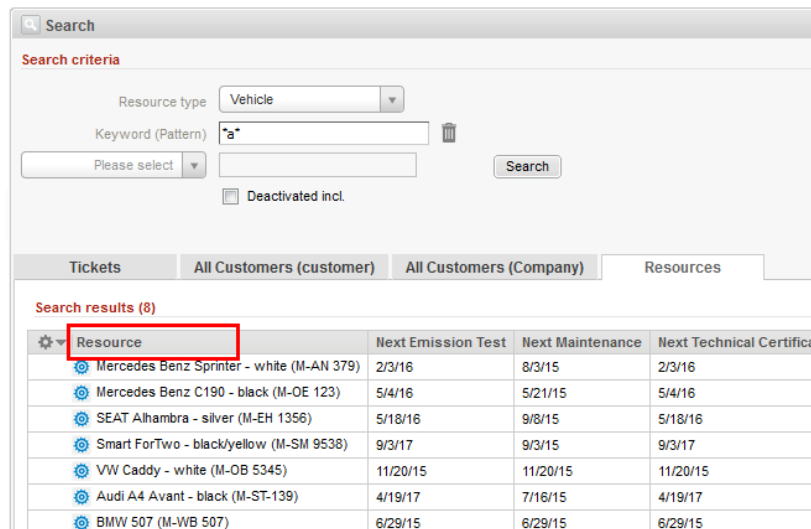


- **Improved visual presentation of "Add all ticket customers option" (#627915):** The plus icon of the function to add all ticket customers when selecting the e-mail recipients has been improved together with minor layout adjustments for the option.



1.3.3 New result column for resource detail search based on template (#628175)

A new column has been added to the result table of the resource detail search page. This column lists the resource name based on the template configured for this resource type and displays the corresponding icon. The column cannot be removed like other result table columns, thus it behaves like the contact result table column in the customer detail search. The column heading label by default shows "Resource", however this will change if the label for resources is changed in the Admin-Tool.



The screenshot shows a search interface with the following elements:

- Search criteria:**
 - Resource type: Vehicle
 - Keyword (Pattern): *a*
 - Please select: (dropdown menu)
 - Deactivated incl. (checkbox)
 - Search button
- Tabs:** Tickets, All Customers (customer), All Customers (Company), Resources
- Search results (8):**

Resource	Next Emission Test	Next Maintenance	Next Technical Certification
Mercedes Benz Sprinter - white (M-AN 379)	2/3/16	8/3/15	2/3/16
Mercedes Benz C190 - black (M-OE 123)	5/4/16	5/21/15	5/4/16
SEAT Alhambra - silver (M-EH 1356)	5/18/16	9/8/15	5/18/16
Smart ForTwo - black/yellow (M-SM 9538)	9/3/17	9/3/15	9/3/17
VW Caddy - white (M-OB 5345)	11/20/15	11/20/15	11/20/15
Audi A4 Avant - black (M-ST-139)	4/19/17	7/16/15	4/19/17
BMW 507 (M-WB 507)	6/29/15	6/29/15	6/29/15

1.3.4 Column sorting and ordering in the customer relation tables (#626279, #627770)

On the customer page the relations to other contacts and companies could not be sorted by clicking the column header until now. Furthermore, the column order could not be changed as expected. These problems have been resolved and the columns can be sorted and reordered like for other tables now just like expected.

1.3.5 Multiple table column settings for different resource types and customer groups (#628236)

The tables presenting the resources and customer had only one column configuration setting previously for all resource types or customer groups, respectively. Only the last modified table was kept as setting. This behavior could be observed for the detail search results and on the resource type pages. It is only partially useful since the field configurations may be very different between resource types and customer groups. This shortcoming has been eliminated and now different table column settings are stored for each resource type and customer group, so that for each one there can be an individual column configuration. The way to change the columns display for these tables has not been changed and can be done just like before.

1.3.6 Clone script engineer assignment modification (#628181)

It was not possible to change the engineer assignment in a ticket clone script of a queue until now. The new ticket always had exactly the same engineer assigned as the source ticket. Changing this in a clone script had no effect. Such a change is working with this release which also includes unassigning the new ticket. An example script is the following code:


```
import com.consol.cmas.common.model.customfield.Unit;
import com.consol.cmas.common.model.ContactTicketRole;
import com.consol.cmas.common.model.customfield.ContactReferenceField;

ticket.setEngineer(engineerService.getById(39));
ticket.setQueue(queueService.getById(7));
ticket.setSubject("cloned");

Unit unit = unitService.getById(120);
ContactReferenceField field = contactReferenceFieldService.getMain(ticket);
field.getValue().setUnit(unit);
```

1.3.7 DWH indicator for active tasks restored in Admin-Tool (#627629)

With the redesign of the Admin-Tool navigation the functionality to display active DWH tasks in the icon bar disappeared with the old icon bar. This functionality has been restored in the new navigation panel now.

A small red exclamation point will be shown next to the navigation group “Data Warehouse”, if there are data warehouse tasks. The number of tasks will be shown in brackets after the label “DWH Tasks” of the corresponding navigation item within the group.



1.3.8 System property for enabling/disabling internationalization of static data warehouse tables (#627716)

Internationalization for static data warehouse tables introduced changes which caused the database update during a DWH update to run significantly longer. This fact was described in section 1.1.3 of the release notes for CM version 6.10.1.0. While this release features optimizations to reduce this step's duration this step still take significantly longer. In a real life system the time was reduced by 40%, however it still could take twice as long as before.

Since the necessity of internationalization for static data warehouse tables is limited a new system property has been introduced: *cmrf.localization.enabled*. The default value of this property is *false*, effectively keeping the previous behavior without internationalized static data tables.

The internationalization now has to be actively enabled meaning that the longer runtime must be willingly accepted. It can be done by supplying this parameter on the command line at the server start, for example on a Windows system:

```
standalone.bat -c=cm6-cmrf.xml -b=10.20.30.40 -Dcmrf.localization.enabled=true
```

Upon (re-) start of the server the database tables are changed and updated accordingly, if the value changed. The relevant localization columns are added or dropped depending on the property value. When setting the value to *false* (or not providing the property value at all) no further action is necessary. When setting it to *true* a data warehouse update is needed to fill the newly created columns.

When using a MySQL database the interaction with the locale definitions from the property *cmrf.mysqlLocales* introduced in CM release 6.10.1.0 (see section 1.3.11 of the Release Notes for this version) works as it could be expected.

1.3.9 Server logging for mail sending made consistent (#620198)

The log messages when sending mail were not uniform and consistent when sending mail in different cases. Log entries differed depending, if they were sent from the web client or from a workflow activity. These differences have been removed so that the logging of sending mail is uniform now.

1.3.10 Obsolete CM.Phone configuration file entries removed (#625957)

The configuration file of the CM.Phone tool has been modified so that old and obsolete entries are removed now. Only the relevant entries are still available in the file.

1.4 Bugs fixed

Number	Description
623593	Ticket queue changes were not represented correctly in a DWH table For tickets after a queue change the value of the field "QUEUE_PREV_ID" wrongly was NULL in the data warehouse table "FACT_TICKET_QUEUE_CHG". The change was entered correctly in the table "FACT_TICKET_LOG". This error has been corrected and the values in the data warehouse table "FACT_TICKET_QUEUE_CHG" now are as could be expected.
626279	Column sorting and ordering in the customer relation tables On the customer page the relations to other contacts and companies could not be sorted by clicking the column header. Furthermore, the column order could not be changed as expected. This problem has been resolved and the columns can be sorted and reordered like for other tables now.
626978	Scene export with customer runtime data contained no customer group information The export of a scene that contained only the customer data as runtime data besides configuration data could lead to problems since the customer group information was missing. This issue has been fixed and scene exports with customer runtime data now contain the necessary customer group and model information.
627250	Exception when opening a resource group with history entries relating to a deleted resource type An exception occurred when opening resources when some specific conditions applied. The history had to contain information about a relation to resource of a type which had been deleted in the meantime. It could also appear with a deleted resource group. The error has been fixed and this kind of information in the resource history will not cause such an exception any more.
627417	Lazy loading display and configuration issues There were small issues with lazy loading display and configuration in the most recent releases. The history entries were not shown collapsed, if the filter "Display communication" was active while lazy loading was activated and also minor variation in the headline strings could appear. Additionally when only setting "headHistoryElementsCount" lazy loading was not activated as expected. These issues have been corrected and lazy loading should work as expected again.
627505	No message when trying to add an existing ticket or customer relation to a resource When trying to a relation with a ticket or customer to a resource, which already exists, there was no validation message shown informing why the relation creation is not possible. Such a message has been added and it is shown after clicking the "OK" button now. So the user is being told about the reason for not accepting this relation.
627610	Faulty order of loading localized CSS files The order of loading localized CSS files did not allow to to override general style definitions with localized ones. This order has been changed so that it now is possible to override general by localized style information.
627660	Label data object fields in field group not displayed A data object field annotated as label which belonged to a field group that was not shown in the group section was not displayed in view mode. In contrast the label was displayed in edit mode. The label field had to be in the first row of the group which is a common position for a label. This faulty behavior was changed so that the label is always displayed now.
627661	Resource field labels not displayed in view mode Resource fields with the annotation "label" were not rendered when viewing a resource before. They were only visible when editing the resource. This unwanted behavior has been corrected and the label field is now displayed both in view and edit modes.

627717	Autocomplete functionality for Enum fields was case-sensitive for umlauts The autocomplete functionality of Enum fields did not find values for upper case umlauts when entering the corresponding lower case umlaut and vice versa. Thus, expected values were not offered for selection. This unwanted behavior has been eliminated and now matching values are found independent of the case of an umlaut letter.
627731	Misleading error message related to resource relations An error message of an exception regarding resource relation creation was misleading and did not inform about the real underlying problem. This message was changed so that it does give proper information and the related property for a resource history entry was fixed and improved.
627756	Exception when trying to sort ticket relations on resource page by status When clicking the header "Status" in the ticket relations on the resource detail page an exception was caused and the user was informed about it. The sort order of the table entries did not change. This problem has been fixed and the click now changes the order of the table entries as desired.
627809	Page customization "companyEditLinkVisible" dysfunctional on company page The page customization "companyEditLinkVisible" had no effect in in the latest releases. Even though its value had been set to "false" the link was available and the company could be edited. This erroneous behavior has been corrected and the customization again works as desired.
627850	Internet Explorer displayed a horizontal scrollbar for quick search results In the latest releases an unwanted horizontal scrollbar was displayed for the quick search results when using an Internet Explorer browser. This especially happened when all results were resources of one type. Attempting to to use the scrollbar made the result list disappear. This undesired display and interaction has been fixed so that the quick search results can be used in Internet Explorer as designed.
627854	Wrong display of fields in groups on the company page after validation In case validation identified an inadmissible value in a field inside a group on the company page the wrong value was displayed. The field showed the last valid value and not the new invalid entry besides the validation message. This wrong display has been changed so that the wrong entry can be seen next to the validation message again.
627884	Company headline repeated when using optional companies When using a customer data model with optional companies, the headline (row positions "0,*") was repeated as first line in the data display when showing a contact with a company. It should be displayed as headline only on ticket and contact pages. This doubled data display has been corrected and these data are shown as headline only now.
627896	Removing the last of several ticket relations on a resource caused an error Removing the last relation caused an exception when a resource had multiple ticket relations and all were removed on the resource detail page. This error has been corrected and now the last ticket relation on a resource can be removed without exception as well.
627918	Admin-Tool displayed "null" for empty descriptions The Admin-Tool showed the text "null" in description field which WHERE intended to be empty. This unwanted display has been changed by correcting the database entries which caused this. "null" should not be displayed any more.
627972	Incorrect mail address selection after selecting "Reply All" When selecting the option "Reply All" on an e-mail which had two plain email addresses without an extra addressee name next to each other these two addresses were shown as one recipient. Thus, they could now be handled separately anymore. This presentation problem has been solved and both plain e-mail addresses can be managed separately again.
627980	Problematic handling of very long field names in column selection Usage of very long field names caused display and interaction problems when selecting these as columns for tables. The width of the selection list adjusted to the field name length which rendered it overly wide. After selection the button for this field also adjusted to the field length which caused it to be cut off on the right end by the display box border. Thus, the symbol to remove the field again could not be accessed. These width problems have been

	resolved and both list and button adjust to the component width now by cutting off the label text. This way it also is possible again to remove the field from the list.
627986	Autocomplete address script type option removed from Admin-Tool The script type "Autocomplete address" has been removed as a selection for scripts from the Admin-Tool again. The type is not required, so it need not be offered. Therefore it has been disabled and will not show in the selection list any more.
628027	Exception when creating an engineer with the user name as password An exception occurred when trying to create a new engineer and setting the password to the same value as the username even though this is prohibited. Trying to save such an engineer profile caused the exception. Now a dialog with the validation message informing about the illegitimate password choice is shown instead of the exception.
628103	Clicking on scrollbar made view list from ticket list disappear in Internet Explorer 11 As soon as the view list in the ticket list was shown with a scrollbar, clicking on this scrollbar in Internet Explorer 11 made the whole list disappear. This happened with more than thirteen views to be listed, since this number caused the display of the scrollbar. This error has been corrected and the list will not vanish any more when clicking its scrollbar in Internet Explorer.
628127	Vertical bar character in e-mail-address made e-mail editor unusable In case an e-mail address value or the template for rendering it contained a vertical bar character the address selector component could not initialize, in turn effectively leaving the e-mail editor unusable. This problem has been solved and a vertical bar character can be used again unrestricted in e-mail address values and templates.
628152	Dependent Enum scripts were not executed in nested contexts Dependent Enum scripts were not executed in contexts where the field was embedded/nested somehow. This was the case for companies of a contact in a two-level customer data model and also for fields in groups with the annotation "show-in-group-section" set to "true". When editing the object field in such a nested context for a dependent Enum field the corresponding script was not executed, so that value selection did not work properly. This error has been resolved and dependent Enum scripts now also work in these nested contexts.
628172	ETL import of MLA values in a list dysfunctional When using MLA values in a list and then importing values for such a list using ETL the imported produced structurally incorrect data. When later displaying these values the Web Client could log out the user with an error. This import problem was fixed, the values are correct now and can be displayed correctly in the Web Client.
628176	Resource groups could not be accessed in a ticket, if too many groups were related Relations to resource groups in the most detailed presentation are shown in tabs. If there were many groups and thus tabs, the ones on the right could not be selected, because they were out of the visible area without controls to access them. This deficit has been fixed and a menu control is now shown after the last tab, if the tabs do not fit into the available space. The menu lists all group tabs.
628236	Multiple table column settings for different resource types and customer groups The tables presenting the resources and customer had only one column configuration setting previously for all resource types or customer groups, respectively. Only the last modified table was kept as setting. This behavior could be observed for the detail search results and on the resource type pages. It is only partially useful since the field configurations may be very different between resource types and customer groups. This shortcoming has been eliminated and now different table column settings are stored for each resource type and customer group, so that for each one there can be an individual column configuration. The way to change the columns display for these tables has not been changed and can be done just like before.
628277	Reply all did not quote the content and caused an error The "Reply all" function on a history entry could cause an exception in the latest release. The desired content entry was not quoted in this case. This error has been corrected and the "Reply all" function works as expected again.

628334	Timed-out REST API user-session caused exception stack trace in the log file When a user session used via the REST API had timed out and it was attempted to use it in a request, the log file showed an exception with a stack trace, even though this is no error but a regular event. This unwanted log entry has been changed, and this event is only logged with one line on DEBUG level.
628350	REST-API user-session invalidated much too soon when using WildFly application server A user session requested via REST API was invalidated within a few seconds after opening when using WildFly as the application server. This highly undesired behavior has been modified so that REST API user sessions are working as configured now for the WildFly application server, too.
628354	Page customization "mailToFixedMail" did not accept multiple addresses In the latest releases the page customization "mailToFixedMail" on the ticket page for example did not accept multiple, comma-separated e-mail addresses anymore like in previous releases. This functional limitation has been removed again and now multiple addresses will work.
628411	Exception in ACF prefill scripts from Groovy variable scope After updating to the latest release some ACF prefill scripts caused exceptions which did not occur before. This was due to Groovy variable scope considerations. This problem has been corrected and now ACF prefill scripts should not cause new exceptions after the update anymore.

1.5 Known Issues

Number	Description
613647	Annotation "order-in-result" dysfunctional The annotation "order-in-result" to define a default field order for search results does not work when used correctly.
621068	Incoming mail headers with umlauts cause error Umlauts and other special characters in the mail header of an incoming e-mail can cause an error in processing the e-mail.
621143	Wrong queue name displayed in ticket history The name of the queue a ticket was formerly assigned to is for some cases replaced by the current queue name rendering such a queue change entry useless.
622836	Admin-Tool role list cut off at the bottom The list of roles in the Admin Tool could be cut off at the bottom, if the list is quite long or the window has been resized. The obstructed last entries usually can be accessed when the window is sufficiently enlarged.
623171	Exception opening a ticket from the workspace after queue change It causes an exception and an empty browser screen when trying to open a ticket from the workspace, if the referenced ticket in the meantime has been moved to another queue for which the engineer has no access.
623767	Workflow activity dysfunctional after switching to a newly created contact A workflow activity cannot be executed immediately after changing the contact which is just created then. After a page refresh the activity is available again.
625571	Removed ticket attachments can be added to e-mails causing an exception on sending A ticket attachment which was removed is still offered for an e-mail when a second attachment has been added in the meantime. When trying to send the e-mail after attaching this one an exception occurs.
626156	Web Client user session not invalidated correctly When instead of properly logging out the login page is accessed with the back button and a different login is successful, the ticket list uses the older login and session (originally occurred in version 6.9.0.0).
626675	REST response missing unit count The REST API response for unit search using a number range lacks the field "Total number of elements" with the result count (originally occurred in version 6.10.0.0).
626847	Struct fields in unit groups cut off on the right side When displaying structs with many fields inside a unit's group field tab the fields on the right get cut off at the right edge of the tab. A necessary horizontal scrollbar is not made available.
626903	Deficits in manifest files Fields for CM-Version and Build-Date are missing in the manifest files (originally occurred in version 6.10.0.0).
627117	Misleading relation transfer message Deleting a resource and trying to transfer a relation to a contact which is already related to the target resource yields a misleading error message about illegal circular relations (originally occurred in version 6.10.0.0).
627223	Custom field annotation "visibility configuration" dysfunctional The annotation "visibility configuration" for custom fields which is intended on which display level field changes should show in the history does work as expected. The different values do not have different effects on the field level.
627434	Creation date ticket search result too large A search for tickets with a specific creation date using the operator "is" will also return results from the next day.

627768	Table export interfering with relation remove undo Removing a relation to another customer on the customer page, then exporting the table and trying to undo the removal afterwards causes an exception.
628060	JBoss cluster cache issues On a JBoss cluster exceptions can occur when only one node is active and queues are deleted then. It also could cause cache exceptions when creating custom fields on one cluster. These issues will be fixed in the context of the platform updates for release 6.11.0.0.
628173	Import with deleting all data fails if a task is scheduled Importing a scene with the option to delete all existing data active fails if there is a task scheduled at the time of import.
628187	Dependent Enum value change in detail search can cause an exception After selecting a value and a dependent value in a dependent Enum in the detail search, then changing the main value and trying to change the dependent value after that an exception occurs.