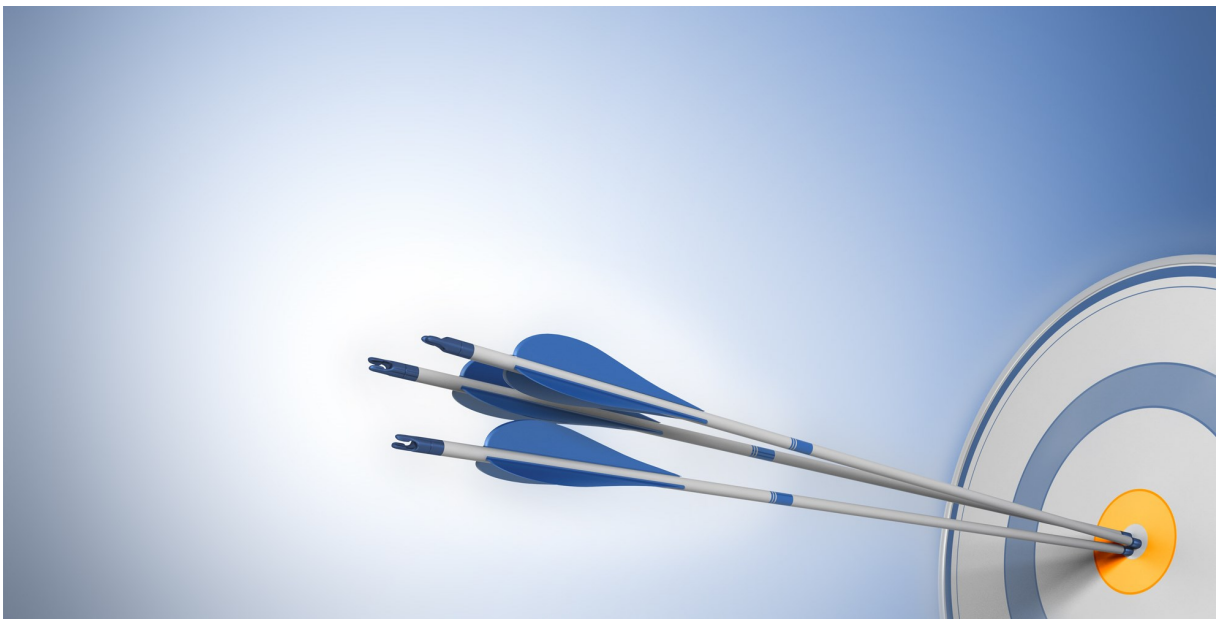


ConSol Software GmbH

Release Notes ConSol CM 6.10.6

Author: Thomas Schall
Phone: Phone: +49 (0)89-458-41-100
Mail: Thomas.Schall@consol.de

Version: 5.0
Date: 28.09.2017
Status: Released

Contents

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 6.10.5 to 6.11.1.
- **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 change 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 6.10.2 to 6.10.5 the instructions of the versions 6.10.3, 6.10.4 and 6.10.5 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 installation instructions” chapter has to be checked for further important notes.

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

1. Version 6.10.6.0 (31.03.2017)

Version 6.10.6.0 includes all ***ConSol CM*** changes and additions of

- 6.10 versions up to 6.10.5.7
- 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 6.10.6.0 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 6.10.5 and earlier versions

This section informs about specific steps when upgrading ***ConSol CM*** from 6.10.5 or earlier versions to version 6.10.6.0.

Oracle JDBC driver issue in data warehouse operations

An issue present in the Oracle JDBC driver could cause the operations following the switch from the ***ConSol CM*** data warehouse ADMIN mode to the LIVE mode to fail. The failure reports an `ORA-01000` error informing about the “maximum open cursors exceeded”. The problem is caused by the Oracle JDBC Driver for Oracle 12.0.1.2 databases.

The presence of this specific problem when encountering the error can be identified by running the following query in a database client:

```
select sql_text, count(*) from v$open_cursor group by sql_text;
```

The driver issue in question caused the error in case this query returns a high number of open cursors for the following originating queries:

```
select 'TABLE' as table_type from dual union select 'VIEW' as table_type from dual
 union select 'SYNONYM' as table_type from dual
```

The problematic behavior can be resolved by **either one** of these measures:

- Install the unaffected Base 12.1.0.1.0 JDBC driver from the Oracle Technology Network website and replace the previous driver. You can get it from this location: http://www.oracle.com/technetwork/database/features/jdbc/jdbc-drivers-12c-download-1958347.html
- Install the patched driver from Oracle support (http://support.oracle.com/). After login search for patch 19632480. Download the patched JDBC driver from this location. You will need a valid login for this procedure.

⚠ Please install an unaffected version of the Oracle JDBC driver, if you are affected by the error!

1.1.2 REST API client modification requirements

Change description.The following REST API enhancements may require to adjust clients accessing ***ConSol CM*** by using the REST API. Eventually the canges can offer a potential for client optimization.

- Activity Control Form support in REST API (#630879)
- Setting specific STRUCT fields via REST (#631055)
- Removing STRUCT rows from a list via REST (#631054)

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 method returning interval with respect to business calendar (#630621)
- Regular display for value from scripted autocomplete fields (#630896)

Please be aware that a method from the first referenced section has been marked as deprecated any may not be present in future releases anymore!

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:

- Explicit deployment timeout setting for ***JBoss EAP*** (#630734)
- Property `MAX_HEADER_SIZE` added to JBoss configuration (#630757)
- Four-byte UTF8 character handling improved for ***MySQL*** databases (#630823)
- JBoss EAP unicast clustering configuration for ***ConSol CM*** (#630839)

No further instructions available.

1.2 New Features

The features newly introduced in this ***ConSol CM*** version 6.10.6.0 are described in detail in this section.

1.2.1 Regular edit display for value in scripted autocomplete fields (#630896)

Scripted autocomplete fields previously were not capable of displaying a value previously entered when editing the ticket later. No matter if the ticket was edited directly or in an Activity Control Form the field always was displayed without value, even if there was on set earlier. This situation can lead to misunderstandings and undesiredly updated/changed field values. Since it is not possible for structural reasons to automatically provide an understandable display value in all circumstances this can only be in the responsibility of the script. The script defines the display values and the stored (key) data and , therefore, it also must define the edit field content as display value.

Please note that this does **not** extend to the normal display of the field value. In view mode the field will still show the key value saved in it. Any other display content still must be achieved by some separate implementation in the respective project context, for example by using a separate custom field!

This edit field display value can now be defined by a new method to be defined in the autocomplete script. This new method `onEditDisplayEntered()` must return the value to display. The logic to achieve this return value is fully in the responsibility of the script author.

```
/**
 * This method controls the value displayed when this field enters edit mode
 * on screen. Can be used to obtain the beautified value based on the current
 * internal value for display purposes.
 *
 * @param pValue the stored value of the field
 * @param pKey fieldKey
 * @param pContext (Ticket/Unit/Resource) holder containing current values
 *           set on the form
 * @return - the string which contain beautified label or one of the domain
 *           objects: Ticket/Unit/Resource,for domain object beautified label
 *           will be taken from standard template of the object.
 */
Object onEditDisplayEntered(String pValue, FieldKey pKey, Context pContext) {
  if (!pValue) return null; // display nothing (original behaviour)
  return "Value set.";
}

/**
 * For custom-fields in ListFields row index is provided as well
 * @param pValue        the stored value of the field
 * @param pKey fieldKey field key
 * @param pRowIndex     row identfication for usage in list fields
 * @param pContext      (Ticket/Unit/Resource) holder containing current values
 *           set on the form
 * @return - the string which contain beautified label or one of the domain
 *           objects: Ticket/Unit/Resource, for domain object beautified label
 *           will be taken from standard template of the object.
 */
Object onEditDisplayEntered(String pValue, FieldKey pKey, int pRowIndex, Context
pContext) {
  if (!pValue) return null;
  return "Value set.";
}
```

The method descriptions above are closely modeled after the built-in template script for auto-complete fields. The second interface variant specifically addresses autocomplete fields inside list field and therefore provided the row index to specify the row for the field in the list.

The simple real-life example below translates the key from the value to the corresponding engineer display name for a field that holds an engineer identification.

```
Object onEditDisplayEntered(String pValue, FieldKey pKey, Context pContext) {
  if(!pValue) return null;
  return engineerService.getById(Long.valueOf(pValue));
}
```

There is no limitation to the logic of the method to retrieve an object value to be returned as the display value. However, it must be used reasonably to provide a meaningful value for the use case of the autocomplete field taking into consideration that no negative performance impact results from the script. However, it generally is possible to update the field value by a different script changing the referenced object, for example bay a workflow trigger.

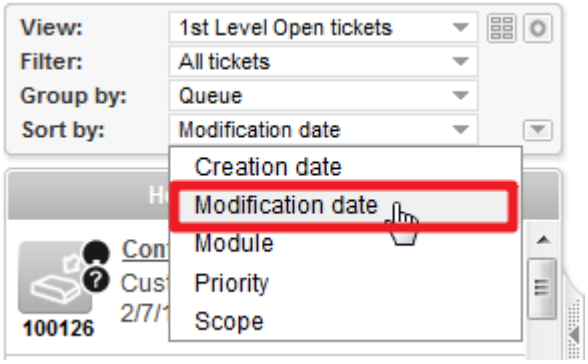
Please be aware that the responsibility for the display value lies completely in the responsibility of the script author! There is no general mechanism beyond executing the method with the scripted logic whatever it may be. Please be aware as well that internal IDs used as keys can change and most probably will do so in system import/exports, for example. In such cases it only is safe to store transfer keys instead of internal IDs.

Existing scripts for autocomplete fields will not be changed during the update. Introducing an edit field display value for these must be made manually after the update. The system will also show the previous behavior and display no value, if the method is not added to the script.

1.2.2 Modification date ticket list sorting option (#630172)

The sorting options for the ticket list have been extended by anew way tot sort the tickets in a grouping section. The tickets can now also be sorted by the last modification date of the ticket. The date and time information used for this are internal data of the system which are also used in the Quick and Easy search. Since these are not from a custom field the information can not be made visible in the ticket list by configuration. There is no configuration for this new option, however, the direction for the sorting can be changed by the small arrow button like for any other sort order.

Please compare the screenshot for the selection of this new option.



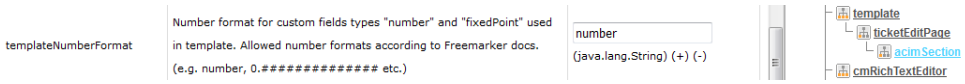
1.2.3 Page customization for number display format (#630170)

The display format for the data from numeric fields used by the placeholders (markers) in text templates was not configurable earlier. This led to situations where the display format presented the actual value in an unwanted way, for example showing thousands separators within an ID number intended to be a sequence of digits only. The format can now be configured by defining the formatting rule in a page customization attribute.

> Please note that this format **only** will apply for the values of **numeric (integer and decimal) fields** which are inserted into a **template text** by using a placeholder (marker). When selecting a template to be added to the Rich Text Editor markers referring to numeric fields are replaced by the value which in turn formatted in the way specified in this page customization attribute.

The new attribute is called *templateNumberFormat* and can be found for the type *template* on all pages which feature the Rich Text Editor with text template insertion. The format is always defined on the lowest scope for which the attribute value is available.

In the example screenshot the entry would apply only to the editor in the history section of the ticket page, since it is in the scope */ticketEditPage/acimSection* of the type *template*.

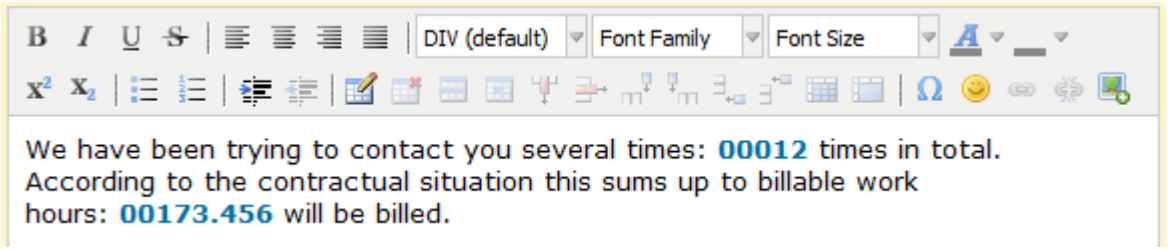


The currently shown value “number” represents the default format and it can be used to return to the default after having a dedicated format configuration previously. The format definition syntax is explained in the public ***Java*** documentation under this URL: https://-docs.oracle.com/javase/7/docs/api/java/text/DecimalFormat.html.

The most important elements are:

- “0” - shows a digit in this place always, will show 0, i f no digit is in this position.
- “#” - shows only, if the number has a digit in this position.
- “.” - decimal separator
- “,” - internal grouping separator for integer/decimal positions

For example “00000.#####” as attribute value will always show five digits before the decimal separator and show up to five digits after the decimal separator, depending on how many are present for decimal numbers. It will always display integers in five digit notation. this effect can be observed in the illustration below. The first number in blue is an integer, the second one is a decimal number with three decimal digits.



The documentation referenced above will elaborate this further with information on how to format numbers in scientific notation, as percentages, currency values, with separate patterns for positive and negative numbers of with special characters as part of the display format. Even more possibilites to influence the format can be found in the Freemarker documentation.

1.2.4 Activity Control Form support in REST API (#630879)

This release features a significant addition to the REST API. Earlier it was already possible to execute activities, but this functionality has been extended with support for Activity Control Forms. This means that now it is possible to execute activities which require to fill an Activity Control Form in the ***Web Client*** by using the REST API.

This new functionality can be used independently which is explained below. However, it is also used by the ***CM.Track V2*** portal client for executing activities with Activity Control Forms, For details on the usage in ***CM.Track V2***, please see the sub-section "Support for Activity Control Forms (#630878)" under CM.Track V2 Improvments.

Executing an activity with form data via REST API requests involves a specific sequence of steps:

1. Get the list of available ticket activities.
2. Get the Activity Control Form information for a specific ticket activity.
3. Get the field (set) information for the form.
4. Executes the activitywith supplying the form data in a PUT request.

The steps are illustrated with real-life examples below. Since the requests are very transparent generic desription of the request structure are not provided here.

The first step is a standard HTTP GET request querying the available activities of a ticket in its current workflow state as in the example below. (Please note that the response in the example is highly beautified in order to provide it in a legible way.)

```
curl -u user:password http://localhost:8080/restapi/tickets/100129/activities

{ "activity":
  [
    { "@name"      : "defaultScope/ticket_incoming/close_immediately",
      "@id"        : "114",
      "description": "Close immediately",
      "info"       : "The ticket is closed immediately",
      "name"       : "defaultScope/ticket_incoming/close_immediately" },
    { "@name"      : "defaultScope/ticket_incoming/qualify_ticket",
      "@id"        : "110",
      "acf"        : {
         "@uri" :
"http://localhost:8888/restapi/tickets/100129/activities/110/form" },
      "description": "Accept ticket ...",
      "info"       : "Ticket is qualified - confirmation email",
      "name"       : "defaultScope/ticket_incoming/qualify_ticket" }
  ]
}
```

The response will contain an element “acf” with the corresponding URI for subsequently getting the information about this Activity Control Form, if an activity has a form associated. The only appearance in one of the two available activities from the example above is highlighted in italics.

The information specific to this individual Activity Control Form can be retrieved by sending a GET request using the respective URI from the last response as it can be seen in the next example.

```
 curl -u user:password
http://localhost:8888/restapi/tickets/100129/activities/110/form

{ "@uri"          :
     "http://localhost:8888/restapi/tickets/100129/activities/110/form",
  "@id"           : "1",
  "fieldsSet"     : { "@uri" :
     "http://localhost:8888/restapi/activityformfieldssets/qualify" },
  "validationRule":
  [
    { "group"     : "qualification",
      "name"      : "impact",
      "required"  : "true" },
    { "group"     : "qualification",
      "name"      : "severity",
      "required"  : "true" }
  ]
}
```

The form data in the response on one hand holds the reference to the information about the Activity Control Form fields set with detailed information about the fields of the form. On the other hand this response contains data constraints in the element “validationRule” like mandatory fields in the form. the above example identified two required form fields.

The URI from the previous response again is used to query the fields set for this form by using it in the next request as it is illustrated in the next example.

```
curl -u user:password http://localhost:8888/restapi/activityformfieldssets/qualify
```

The response is not shown here regarding the amount of data it contains even for this small example. However, it includes the information listed in great detail:

- Name
- Group
- Class
- Type
- Localized name
- Sort index
- Potential values for ENUMS
- Annotations, and
- Position

The information from the responses can then be used to craft the request to actually execute the activity selected from the first of this sequence of responses.

The activity is identified by its “@id” from the first response, the required fields are listed in the second one and detailed information on identifying the fields and their potential values are to be taken from the third response. The fields for which values should be submitted are identified by the “@group” and “@name” elements from the third response, ENUM values are taken from the respective “value” element.

The activity itself then is executed like other activities by sending a PUT request. The form values should be sent as payload data in the HTTP message body like the request code below exemplifies. The data sent here contains the activity id and the field values for three fields from two groups:

- `qualification.impact`,
- `qualification.severity`, and
- `helpdesk_standard.module`.

The values supplied in the request are all taken from the respective ENUM value element of the REST response containing the Activity Control Form fields set.

```
curl -u user:password -X PUT -d
 "activity=110&qualification.impact=high&qualification.severity=high&helpdesk_
 standard.module=inventory" http://localhost:8888/restapi/tickets/100129/activities
```

The successful completion of this PUT request will return no content payload, only errors will have content informing about the error and holding a stack trace eventually.

After executing the activity the ticket will be in the following workflow status in a way that is identical to executing the activity with filling the Activity Control Form in the ***Web Client***. Follow-up activities thus can also be made by the REST API liked described or in the ***Web Client*** just as suitable.

1.2.5 New method returning interval with respect to business calendar (#630621)

The was no special functionality previously that allowed to calculate the business time interval between two points in time. A new method has been added to the respective utility class which can be used for this task. The method `getBusinessTime()` uses two timestamps and a business calendar as parameters to calculate the interval which passes only within the business hours defined by the calendar identified. Holidays and out-of-business hours between the two points in time will not be added to the interval. The interval for all the methods described here is a value in milliseconds.

In addition a formerly misnamed method (originally `getBusinessTime()`) in this utility class has been renamed to `getEscalationTime()`. Its previous name/interface is still present but it is marked as deprecated and may be removed from a future release.

Class/Method	Interface	Description
`BusinessCalendarUtil.getBusinessTime()` **NEW**	`getBusinessTime(Date pCurrentDate, Date pEndTime, BusinessCalendar pCalendar)`	Method added, calculates the interval (milliseconds) between two timestamps taking a business calendar into respect
`BusinessCalendarUtil.getEscalationTime()` **NEW**	`getEscalationTime (Date pCurrentDate, long pDurationInMillis, BusinessCalendar pCalendar)`	Replacement for the deprecated method, calculates a timestamp by adding an interval (milliseconds) to another timestamp taking a business calendar into respect
`BusinessCalendarUtil.getBusinessTime()` ***DEPRECATED***	`getBusinessTime(Date pCurrentDate, long pDurationInMillis, BusinessCalendar pCalendar)`	Deprecated method, calculates a timestamp by adding an interval (milliseconds) to another timestamp taking a business calendar into respect

⚠ Please take adequate measures, if scripts in your installation make use of the deprecated method interface!

1.2.6 Database foreign key consistency validation (#630911)

The database foreign keys ensure that the data in the database for different object types is consistent. However, the common years long operation of a database for a ***ConSol CM*** installation with multiple update cycles over many ***ConSol CM*** versions and ongoing development with repeated import/exports of the system can result in few foreign keys being inconsistent. This is mostly harmless as long as normal operation is unimpeded.

The large extent of database changes in the update process to 6.11 version from earlier ones can fail due to foreign key inconsistencies. This problem can only occur for the ***MySQL*** and ***MSSQL*** database products, ***Oracle*** databases are generally unaffected.

A basic test for foreign key consistency has been introduced to the server start process. During server start the presence of the foreign keys is tested now, mean that it exists, that it is trusted/validated and that it is not disabled. Problems which may be detected are logged in lines as shown here:

```
[exec] 2017-03-10 15:28:26,356 INFO  [DatabaseConsistencyCheckerImpl] [-] Checking
foreign keys...
[exec] 2017-03-10 15:28:26,427 WARN  [base.AbstractForeignKeyChecker] [-] Foreign
key is missing. Please execute: alter table cmas_cnt_file add constraint
FK62B8515528645767 foreign key (parent_id) references cmas_cnt_entry;
[exec] 2017-03-10 15:28:26,427 WARN  [base.AbstractForeignKeyChecker] [-] Foreign
key is missing. Please execute: alter table cmas_cnt_file add constraint
FK62B851555E9267A5 foreign key (file_id) references cmas_cnt_entry;
[exec] 2017-03-10 15:28:26,429 WARN  [DatabaseConsistencyCheckerImpl] [-] Foreign
keys not ok
```

A more detailed test for consistency involves the validation of the data of foreign keys referencing rows in other tables and ensuring these are not corrupted or missing. This can be achieved by a builtin command for ***MSSQL***. The execution of this command requires membership in the `sysadmin` fixed server role or the `db_owner` fixed database role. Execution may take several hours on productive databases.

```
DBCC CHECKCONSTRAINTS WITH ALL_CONSTRAINTS
```

A comparable test can be run on ***MySQL*** databases by executing a custom procedure which must be created in an SQL client for the ***ConSol CM*** database as the application database user. Older versions of the procedure should be deleted first with this command:

```
drop procedure if exists check_foreign_keys
```

The foreign key check procedure can then be created by executing the following code:

```
create procedure check_foreign_keys()
  begin
    declare tableSchema, tableName, columnName, constraintName,
            referencedTableSchema, referencedTableName,
            referencedColumnName varchar(64);
    declare done int default false;
    declare c cursor for select table_schema, table_name, column_name,
                                constraint_name, referenced_table_schema,
                                referenced_table_name, referenced_column_name
                           from information_schema.key_column_usage
                          where referenced_table_schema is not null;
    declare continue handler for not found set done = true;
    drop temporary table if exists invalid_foreign_keys;
    create temporary table invalid_foreign_keys(`Table` varchar(256),
                                                `Constraint` varchar(64),
                                                `Where` varchar(512));
    open c;
    l: loop
     if done then
       leave l;
     end if;
     fetch c into tableSchema, tableName, columnName, constraintName,
                  referencedTableSchema, referencedTableName,
referencedColumnName;
     set @query = concat("insert into invalid_foreign_keys select '", tableSchema,
                    ".", tableName, "', '", constraintName, "',
                    concat('", columnName, "=', referring.", columnName,
                    ") from `", tableSchema, "`.`", tableName,
                    "` as referring left join `", referencedTableSchema,
                    "`.`", referencedTableName, "` as referred on (referring.`",
                    columnName, "` = referred.`", referencedColumnName,
                    "`) where referring.`", columnName,
                    "` is not null and referred.`", referencedColumnName,
                    "` is null");
     prepare statement from @query;
     execute statement;
     deallocate prepare statement;
   end loop;
   close c;
   select * from invalid_foreign_keys;
 end
```

The procedure can be executed then by running the following command:

```
call check_foreign_keys()
```

Executing the script returns a list of inconsistencies. After correction of inconsistencies the procedure can be removed again with the drop statements listed above.

After this diagnostic check from either procedure for ***MSSQL*** and ***MySQL*** databases, the findings have to be evaluated each, and measures specifically tailored to every single result must be executed. This can be re-assigning a foreign key value manually or correcting a row with a corrupted data field manually. This follow-up data consistency maintenance must be complete so that there are no more foreign key problems when updating to 6.11 versions of ***ConSol CM***.

1.2.7 JBoss EAP unicast clustering configuration for *ConSol CM* (#630839)

The ***ConSol CM*** configration for cluster communication on multiple ***JBoss EAP*** application servers using unicast connections has been re-validated and extensively documented. This communication channel uses unicast TCP connections in contrast to the default multicast UDP connections. The implementation of this kind of clustering communication should always be made in close coordination with ***ConSol CM*** consulting. Therefore this documentation is not publically available , but will be provided on demand only by your ***ConSol*** representative.

1.3 Changes

This section illustrates all relevant software product changes for version 6.10.6.0 of ***ConSol CM***.

1.3.1 Layout Improvements

Unfold arrow in shortened display for image only entry (#631038)

The arrow to unfold a shortened history entry could have been missing, if the entry consisted only of a picture. This could happen in the basic view configured for short display depending on the user (privileges). The missing arrow has been made available again to unfold and view an image only entry. Please note that this change is the re-implementation of an original change (#630510) implemented in release 6.10.5.5 which had to be reverted in release 6.10.5.6 due to performance issues.

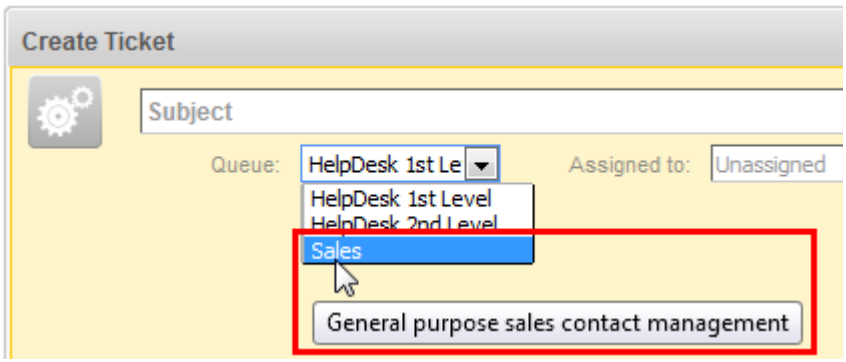
1.3.2 Performance Improvements

Unnecessary access right checks for additional customers removed (#631063)

When accessing any editor component on a ticket related page a check of the access rights for the engineer on all customers was performed. This check in the context of template usage, however is strictly only required for the main customer and not for all additional customers. In cases with high numbers (in the hundreds) of additional customers this could massively slow down the initialization of the editor component. Therefore, the access check has been limited to only the required case of the main customer speeding up editor initialization in these border cases.

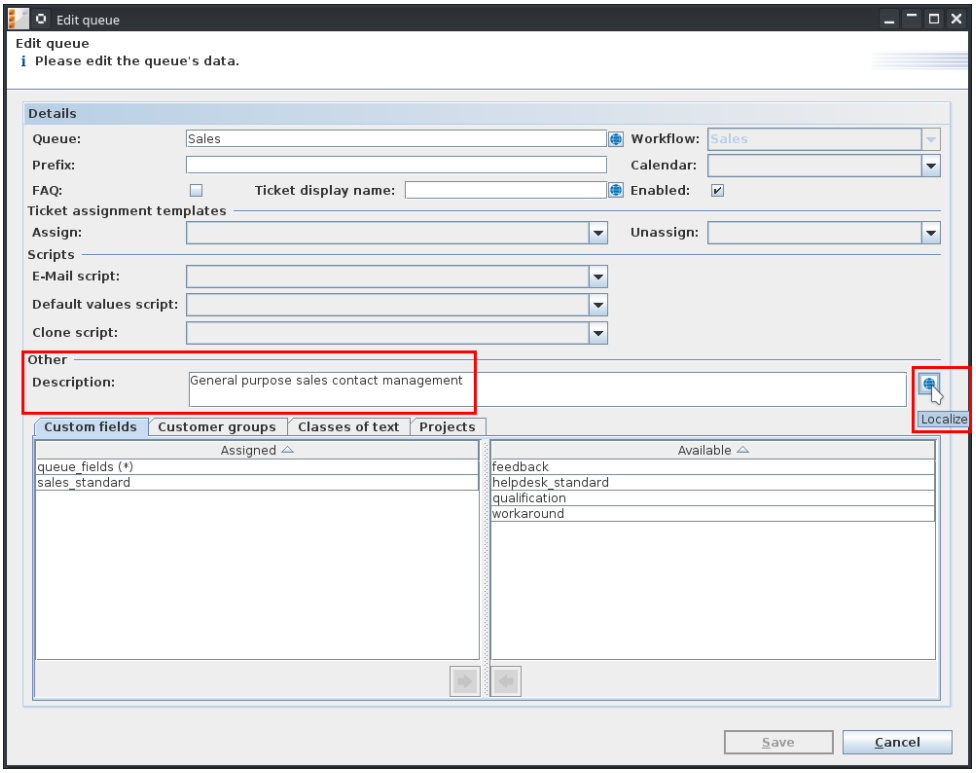
1.3.3 Tooltips for queue selection lists (#630228)

This release introduces additional tooltips for the selection lists for the queue on the ticket create and edit pages. When hovering with the mouse cursor over an entry for a queue in these lists a tooltip will be displayed showing the localized value for the description of the queue defined in the ***Admin Tool***. The tooltip will also be shown, if there is a selector for the queue in an Activity Control Form. There will be no tooltip display, if there is no value defined in the browser language for the description. It will now be shown in general for the queue selection on the detail search page, the engineer profile page, the text template administration and the bindings section of the text template create and edit pages.



The text length for the tool tip is unlimited for the ***Firefox*** browser. When using ***Microsoft Internet Explorer*** only first 512 characters of the description will be shown as tooltip. This is a limitation of the ***Internet Explorer*** browser.

The text shown as tooltip is the localized description for the queue defined in the ***Admin Tool***. The description for the main language can be set in the create/edit queue dialog, other localization texts must be defined in the dialog opnend by the localize button. These two controls are highlighted in the screenshot below.



There is no specific configuration for these tooltips besides the localized description value.

1.3.4 Link disabled for URLs added through REST API (#631095)

URLs added to a ticket for example are rendered as clickable links in general only as long as they are added by a trusted application. However, erroneously URLs added by using the REST API were lately rendered as clickable links as well even though the external clients accessing the REST API cannot be considered trusted in general. this behavior has been aligned again with the general rule so that URLs added via REST API will not be clickable anymore.

Trusted applications in this context are the ***ConSol CMWeb Client***, ***Admin Tool***, and ***Process Designer***. These can be used to add URLs manually, via templates or by using scripts, and they will be shown a clickable links. Untrusted applications in comparison are clients with potentially external exposure like ***CM.Track*** or custom clients accessing the REST API. The eventually external origin of data from these clients offers potential for malicious entries so URLs entered this way are not trust-worthy and therefore will not be rendered clickable.

1.3.5 Four-byte UTF8 character handling improved for *MySQL* databases (#630823)

Incoming emails with a subject line containing four-byte UTF8 characters could not be handled by some installations using the ***MySQL*** database engine. The reason is the encoding/collation configuration of the database using a two-byte BMP 0 plan which cannot be changed in some installations for technical reasons. Other database engines were unaffected.

Mails with this encoding clash could not be imported into the system at all earlier. In order to accommodate this issue a new system property for configuration was added. In the module `cmas-core-server` the new property `strict.utf.bmp.eanbled` can take boolean values. Setting it to "true" will filter out all four-byte UTF8 characters before any database interaction, so the problems mentioned above will not occur anymore. The property value is "true" by default for ***MySQL*** databases, and "false" for any other database where it should not be necessary at all. Change it for a ***MySQL*** database only, if the settings positively will support four-byte characters.

1.3.6 File type meta data for email attachments case insensitive (#630743)

The attachments of an incoming e-mail were previously rejected, if the their file type information was sent in upper case. This is not prohibited by the respective RFCs even though conventionally this information is transmitted in lower case. The validation of admissible file types has been changed to be case insensitive so that admissible file attachments are accepted, even if their file type description is communicated in upper case.

Please take note of the closely related defect #630632 listed in the section Bugs fixed below.

1.3.7 Index synchronization method in cluster modified (#630949)

The synchronization of the index in a cluster method has been changed in order to address cases with bigger amounts of information to be synchronized. Formerly the information was transmitted between the nodes in the header of an HTTP GET request. This caused issues when bigger amounts of information had to be synchronized which exceeded the header size limitations of an installation. This kind of problem will be avoided now by using POST request with the information payload in the request content section.

Besides this method modification additional logging on the indexer slave node has been introduced regarding the node communication in index synchronization. This extends to normal communication, broken synchronization links and the case when the slave takes over and becomes the new indexing master, however it may only reflect in the log file `index.log` with the log level set to "DEBUG".

1.3.8 Property `MAX_HEADER_SIZE` added to JBoss configuration (#630757)

The property `org.apache.coyote.http11.Http11Protocol.MAX_HEADER_SIZE` has been added to the configuration for ***JBoss*** type application servers to all system setups. It has been set to a value of "65535". The explicit definition of a value for this property is specifically supporting

cluster and SSO configurations. 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 installation where reasonable.

1.3.9 Explicit deployment timeout setting for *JBoss EAP* (#630734)

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 modifiying it in `cm6.xml`, for example, as long as it 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 installation where reasonable.

1.3.10 Page customization attribute for view level renamed (#630736)

The page customization attribute for setting the character limit when displaying a history entry in the basic (shortest) view in the ***Web Client*** originally did not follow this naming convention. It was called wrongly *standardViewCharactersLimit* and now has been renamed to the correct *basicViewCharactersLimit* (found on the ticket page type *acimSection* in the scope *ticketEditPage/acimSection*). The name change is done automatically during the update and it will keep the previous value intact.

1.3.11 Setting specific STRUCT fields via REST (#631055)

Setting the content for a STRUCT row in a list only was possible earlier, if the data field in question already had a value. As long as the field was empty there was no method to set the value of a data field inside a STRUCT. The REST API has been extended to provide a mechanism to specifically set a field value inside a STRUCT independent of the previous state. The usage of this method is consistent with the existing methods to handle list values. A field in a STRUCT can now be addressed with this extended syntax: `groupName.fieldName.struct_UUID` resulting in the following ways to set and unset values:

```
... groupName.fieldName.123-123-123-123=Value
... groupName.fieldName.123-123-123-123=
```

The first example is used to set the field value of the field `groupName.fieldName` in the STRUCT row with the transfer key `123-123-123-123` to "Value" independent of the previous status if the field contained a value or if it was empty. The second example without a specific value can be used to delete an existing STRUCT element.

1.3.12 Removing STRUCT rows from a list via REST (#631054)

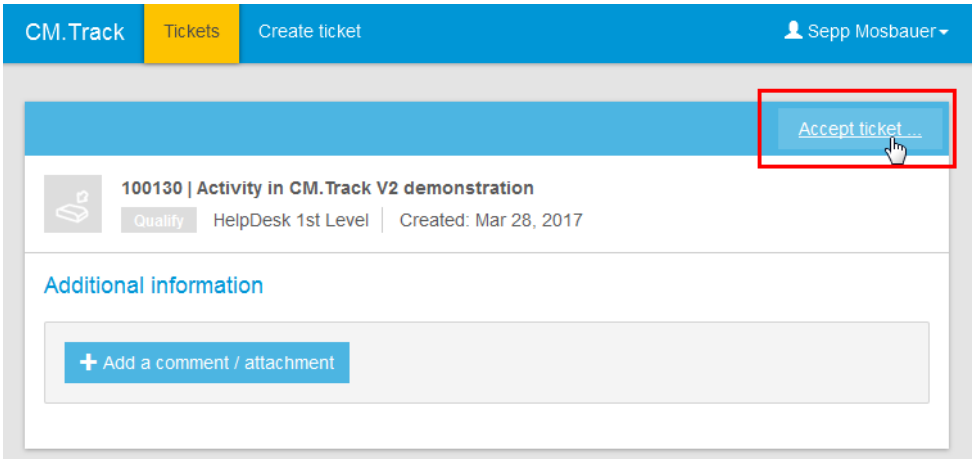
The REST API as documented is capable of deleting values in a list. This worked previously for simple lists and for fields inside a STRUCT forming one row of a list. Thus, earlier it was not possible to delete the STRUCT itself, but only discrete field values. the capabilities have been extended, so that now it also is possible to delete STRUCT rows from list fields with the method explained in the ***ConSol CM*** REST API documentation.

1.3.13 ***CM.Track V2*** Improvements

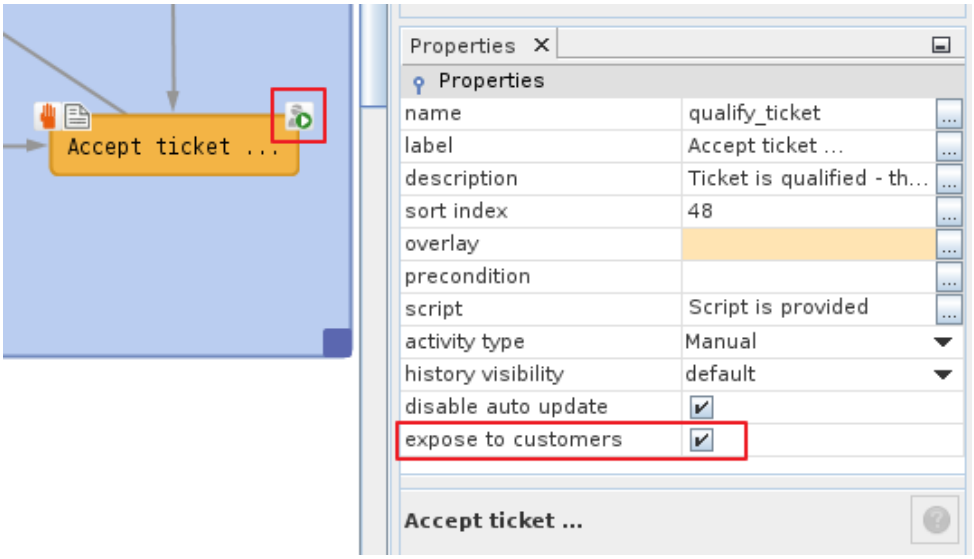
Workflow activities availability in ***CM.Track V2*** (#630310)

This release features avaiabililty of workflow activities in ***CM.Track V2***. This enhancement open up a whole set of new use cases for this portal client. Activities are not immediately available out of the box, but rather must be made available.

The available activities for the current ticket will be shown in the header bar above the ticket when displaying a single ticket as highlighted below. Clicking the activity link will execute the activity and put it in the followup workflow state which will bee displayed immediately. If the activity has an Activity Control Form associated this will be shown and must be filled before the activity executes. The Activity Control Forms are explained in more detail in the next sub-section.



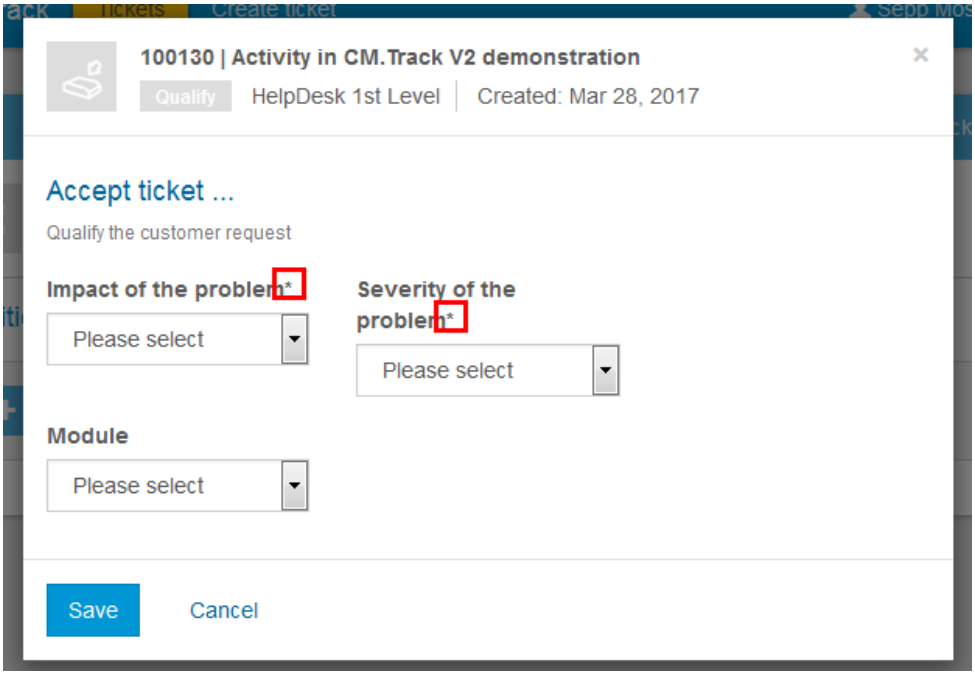
Any manual activity can be made available for ***CM.Track V2*** users inside the ***Process Designer***. This is done by checking the check *expose to customers* among the properties for manual activities. A manual activity with this property set will be marked with a symbol on the activity on the canvas. The symbol is a person with green a play symbol overlay on the top right corner of the activities' box. In the most basic case this is sufficient to have a workflow activity in ***CM.Track V2***.



Such an activity will only be shown, if the ticket is in the correct workflow state to offer the activity. Existing pre-condition scripts apply as well for the portal client. Each activity in ***CM.Track V2*** has it own ID for CSS styling derived from its name. Please refer to the documentation about customizing this client.

Support for Activity Control Forms (#630878)

The new activity execution functionality described above is enhanced by general support for Activity Control Forms when executing these activities. The forms are intended to be used for data modifications when executing the associated activity.



The forms support custom as well as data object fields so that a customer could change his own data. It is also possible to modify, add and remove rows in single column or STRUCT list fields. Required fields are marked by an asterisk and must be filled as expected. Pre-fill scripts defined for the form will also work and set initial values. The form dialog layout follows the Activity Control Form layout defined in the ***Admin Tool*** for it. An appropriate message will be displayed, if the tickets has been modified by another person in the meantime. The same applies to the case an error is encountered.

However, there is a number of **limitations** for the use of Activity Control Forms in ***CM.Track V2***:

- Queue and engineer fields are not supported, thus not available.
- Dependent ENUM fields are not supported.
- Autocomplete features for fields are not supported.
- No post-activity flow modifications like `postActivityExecutionHandler` scripts are possible.
- Custom field annotations from the group `cmweb-common` which are mostly relevant for the ***Web Client*** ticket list configuration are not taken into account.

The majority of the limitations should generally not be relevant considering the central use case for which is allowing customers to interact with tickets when needed in the process. Therefore, process elements not intended for customer interaction need not be available.

FAQ ticket list aligned with regular ticket list (#630933)

The ticket list shown for FAQ listings earlier was simplified in ***CM.Track V2*** compared to the regular ticket list of the customer. It did not show different ticket symbols or colors and provided no scope and date information. The ticket information shown now is aligned with the regular ticket list showing all this information which is also the listing present in the previous ***CM.Track*** product.

FAQ search inconsistencies eliminated (#630934)

The ***CM.Track V2*** FAQ search page showed several aspects of behavior which were inconsistent.

In some cases a FAQ search was performed even though the button was not clicked. This happened for example when switching between ticket and FAQ pages or between the result pages of the FAQ. Besides the search could yield different results when executed on the first or later result page. The filter dropdown list was reset after clicking the search button effectively showing another value than the one used for the search returning the results displayed.

These items have been made consistent in their use in themselves and with the rest of this portal client, so that results are constant and the search is only executed after clicking the button. Additionally specific FAQ search criteria are only displayed after a page switch, if a search using them actually was executed by clicking the button. Otherwise the criteria will be reset.

Opening child tickets inoperational (#631044)

In earlier versions of ***CM.Track V2*** it was not functional to open a child ticket created in the ***Web Client*** before. Trying to access it returned to the ticket list instead of showing the ticket information. This deficit has been changed so that child tickets are accessible in general now.

Display of long string fields enabled (#631062)

Previously it was not possible to display fields of the data type long string in ***CM.Track V2***. This inconsistency has been removed so that this data type can be used normally now.

Time only entry ignored (#630940)

The value set for a datetime field when only entering a time value used as date “01.01.1970”. This is undesired especially since the ***Web Client*** ignores such an entry completely. This behavior has been aligned so that the entry is ignored in ***CM.Track V2*** as well in order to avoid confusing values and to align behavior across the clients.

Timezone offset of datetime fields in view mode (#630810)

The display value of date fields with a datetime accuracy and a format setting for entering and displaying eventually showed one or more hours offset due to unintended timezone miscalculations. This undesired offset has been eliminated and the time value entered is the values displayed now.

Error interceptor configurable (#629470)

The error interceptor in ***CM.Track V2*** has been made configurable so that a customized version can react differently on response status codes than the standard distribution. Please see the documentation for ***CM.Track V2*** customization for more details.

Custom field label translations (#630572)

The labels of custom fields did not change the label language when the browser language was changed. All other labels were unaffected and changed the language as desired. This inconsistency has been changed so that custom fields will change the label language along with other labels.

Browser language switch in Internet Explorer only partial (#630818)

When trying to switch the browser language in Microsoft ***Internet Explorer*** only custom field labels changed the language, other labels were unaffected. This has been changed so that all labels can switch the language, however, this is only applied when reloading the page by pressing the `F5` keyboard button or restarting the browser.

1.4 Bugs fixed

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

Number	Description
628863	**Contact data not refreshed when ticket is closed by workflow activity** The contact data of a ticket was not refreshed, if the ticket was closed by some activity, even though these contact data could have been modified by scripts executed for this activity. This could result in the display of outdated information for the contact. This defect has been corrected so that this situation should not cause the display of information which has changed in the meantime.
628984	**Ticket change not visible in *Web Client* because of URL reference to a specific ticket version** On some occasions a change in a ticket could not be seen immediately in the ***Web Client*** because the URL in the browser referenced a specific ticket version before this change. A manual page refresh could not change this either. An example case was closing a ticket by a timer trigger in the meantime after it was changed by an Activity Control Form. This overly specific ticket version reference has been changed to the ticket only reference, so that changes reflect as desired again.
630111	**Exception when entering a value for a non-indexed string field in an indexed STRUCT list** Trying to enter a value in a string field which is not indexed but part of a STRUCT list which is indexed caused an exception on the customer creation page. This problem was corrected and data entry works in this case.
630147	**Text template issues with text blocks** There were several minor issues with the use of text blocks in text templates. Adding a second template after adding a template with several text blocks in a ticket entry did not work. Pressing the Enter key twice after setting the cursor at the end of the last line or the beginning of the first line of the template text deleted the template text, if the text originated from a template using a text block. The template selector was partly not correctly updated in these cases. The issues described here have been corrected and the interaction with templates containing text blocks works as expected now.
630390	**PDF preview problems when ticket was created from mail without body content** When a ticket was created from an incoming mail with no body content but only one or more PDF attachments the ticket preview in the detail search did not work as expected, especially when using Internet Explorer. It did not initially show the PDF or only showed it in a misshaped rectangle area, but after trying to switch attachments the preview showed correctly. This problem was fixed and the PDF should preview as expected immediately for this kind of ticket.

Number	Description
630461	**Database transaction timeout on saving user preferences with multiple browser tabs** A change in the user preferences which had to be stored could cause a database error, for example an `ORA-02049: timeout: distributed transaction waiting for lock` on Oracle databases, when there were multiple tabs open with potential preference changes. This situation has been changed so that even when multiple tabs are used this kind of database timeout will not happen anymore.
630632	**Email MIME type identification was case-sensitive** The recognition of MIME types for the email content was case sensitive only properly identifying lower case letter descriptions, for example “text/html”. Upper case descriptions like “text/HTML” were not recognized and treated wrongly as “text/plain”. This erroneous behavior was fixed and upper case descriptions are handled correctly now as well.
630634	**Context ticket object not reliable for resource relation search in activity control form autocomplete field script** When using the current ticket with the method `pContext.getTicket()` in the script for an autocomplete field of anActivity Control Form in order to retrieve resources with a specific relation to the ticket (as criteria for `resourceRelationService.getByCriteria()`) for selection the result was larger than desired because the object identification did not work as expected and returned all resources with the defined relation to any ticket. However, using the method `ticketService.getByName(pContext.getTicket().getName())` did yield the desired results. The problem with the shorter method call has been resolved and this case will now return the desired result set.
630639	**ETL resource ID lookup step exact match too unspecific** The exact match condition for the resource ID lookup step of an ETL transformation did not work exactly as expected. If two resources had values where one was the beginning substring of the other, both were considered a match when strictly searching for the substring only. This is unwanted since the longer name cannot be seen as an exact match and works correctly in the corresponding resource output step. It has been corrected and both steps return exact matches as expected now.
630646	**Square brackets not usable in rich text editor for comments and emails** The usage of square brackets in the rich text editor for comments and email was not possible because entering an opening bracket was immediately interpreted as a placeholder which then was added automatically, replacing the bracket. Additionally, after adding a template which ended in a placeholder any subsequent text was treated as if it was part of the placeholder. This has been changed so that brackets are usable again in the editor **Please note** that the original behavior is still present in the editor when editing text templates themselves! In this case this behavior is rather wanted and thus has not been changed!

Number	Description
630670	**Field values of text templates did not show in the editor text** Field values from engineer data in a text template which was added to the editor did not show up in the editor text although they even were shown in the preview. This undesired behavior was altered and now the values are present in the editor text as well.
630681	**Attachments from emails without text body were not attached to a ticket** If an incoming email did have attachments but no mail body with text, which is perfectly legal, the attachments were not attached to both existing and new tickets. For new tickets the ticket was not even created in such a case. This defect was addressed and emails with attachments only but no body will be handled as desired adding the attachments to a ticket.
630717	**Layout definitions for dashboards did not work with spaces between the elements** The layout definitions of dashboards did not work, if the page customization attribute value contained whitespace characters to separate the individual elements of the complex value. Only values with no spaces outside of string values were working. This faulty behavior was recently introduced with tabbed dashboards. It has been addressed and layout definitions structured with space characters are working again.
630724	**Dynamic horizontal sizing of selection lists caused values to be cut off on the right** Selection lists for a field with many entries showing a scrollbar to access all entries cut off the value labels on the right size due to their dynamic width calculation. Thus, the choices were not completely visible which could impair the selection. This has been corrected and now the list is as wide as the field for which it is shown and, if there is a longer entry in the list, a horizontal scrollbar will be shown on the bottom of the select list.
630779	**Busy indicator message deficit when deleting tickets in *Admin Tool*** When deleting tickets in the ***Admin Tool*** the busy indicator showed a message with a placeholder instead of the valid information. This happened for all cases and localizations. The deficit has been removed and now for all messages in this context the placeholder is replaced as desired.
631081	**Protocol uppercase name causing invalid URL** Entering an URL to an URL field with an upper case protocol name “HTTP” caused the value to have a lower case protocol name added at the front resulting in a value like “http://HTTP://some.address.com/”. This error has been corrected and an uppercase “HTTP” will not result in an invalid URL anymore.

2. Version 6.10.6.1 (11.09.2017)

Version 6.10.6.1 includes all ***ConSol CM*** changes and additions of

- 6.10 versions up to 6.10.6.0
- 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 6.10.6.1 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.

This release does not feature an updated ***CM.Track V2*** version. Please continue using the unchanged 6.10.6.0 version.

2.1.1 System upgrade from 6.10.6.0 and earlier versions

This section informs about specific steps when upgrading ***ConSol CM*** from 6.10.6.0 or earlier versions to version 6.10.6.1.

2.1.2 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:

- REST API customer access to object restriction enforcement (#632194)

No further instructions available.

2.2 Changes

This section illustrates all relevant software product changes for version 6.10.6.1 of ***ConSol CM***.

2.2.1 REST API customer access to object restriction enforcement (#632194)

The REST API when used with valid customer credentials did allow accessing the data of other unrelated objects, if a REST request with a manipulated structure and a valid ID was crafted manually and issued. This access is undesired for most use cases. Usage of REST calls with engineer credentials will generally apply the desired access restrictions.

A new system property has been added which controls REST customer data access with customer credentials. It can be found in the module `cmas-restapi-core` and is called `security.restrict.unit.access.to.own.data`. It will be introduced automatically by the update with its value defaulting to “true”.

The new restricted unit object data access policy applied by the property value “true” will activate an additional check for requested customer data. The requested information will then only by returned, if either

- the requested item is the company for the customer logged in or
- the requested item is another contact of the company for the customer logged in.

Requests for other unit object data will get a response status `403 FORBIDDEN` in return. This policy is enforced for all requests for customer data, no matter if they are requested by ID or by search criteria.

Setting the property value to *false* will allow the less restricted data access as previously for backwards compatibility.

2.2.2 Text escaping improved in ticket history (#632195)

The text in a history entry like a comment or an email generally gets escaped to avoid execution of script code in the texts. The code in effect is not executable anymore after this change. The escaping was not fully applied specifically to the content of a collapsed section, which in theory could be exploited to execute script code within the browser page display. The text escaping in the ticket history has been revised to fully escape all content of history entries in all display modes, so that this potential injection vector conceptually cannot succeed anymore, no matter how the script code has been added to the ticket history.

3. Version 6.10.6.2 (26.09.2017)

Version 6.10.6.2 includes all ***ConSol CM*** changes and additions of

- 6.10 versions up to 6.10.6.1
- 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 6.10.6.2 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 6.10.6.1 and earlier versions

This section informs about specific steps when upgrading ***ConSol CM*** from 6.10.6.1 or earlier versions to version 6.10.6.2.

3.1.2 REST API client modification requirements

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

- List of STRUCTs field filtering in the REST API (#632273)

No further instructions available.

3.2 Changes

This section illustrates all relevant software product changes for version 6.10.6.2 of ***ConSol CM***.

3.2.1 List of STRUCTs field filtering in the REST API (#632273)

The availability of fields which are accessed via the REST API with a customer login is controlled by the annotation `customer exposure`. Setting its value to “none”, for example, should not return the field to any customer accessing the system over the REST API for any use case. This filtering did not work as expected in the case of lists of STRUCTs shown in Activity Control Forms. In such a specific context all STRUCT fields were shown undesiredly instead of only those exposed to customers. This deficit has been removed so that in the context of Activity Control Forms lists of STRUCTs only show fields explicitly exposed to customers.

Please be aware to adjust any REST API client explicitly dealing with the unexpected behavior described here!

4. Known Issues

The following table lists known issues in the 6.10.6 releases. Please note that the column “Since” does not specify the earliest release this issue became known, if it was known before version 6.10.6.0.

Number	Since	Resolved	Description
621068	6.10.6.0 and earlier	obsolete in 6.11.0.0	**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	6.10.6.0 and earlier	open	**Wrong queue name displayed in ticket history** The current queue name is for some cases replacing the name of the queue a ticket was formerly assigned to. This is rendering such a queue change entry useless since it wrongly names the same queue for previous and current queue association.
622836	6.10.6.0 and earlier	resolved in 6.11.0.0	***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	6.10.6.0 and earlier	will be resolved in 6.11.2.0	**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	6.10.6.0 and earlier	resolved in 6.11.0.0	**Workflow activity dysfunctional after switching to a newly created contact** A workflow activity cannot be executed immediately after changing the contact which was just created in this step. After a page refresh the activity is available again.
626156	6.10.6.0 and earlier	will be resolved in 6.11.1.0	***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.
626675	6.10.6.0 and earlier	open	**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.

Number	Since	Resolved	Description
626847	6.10.6.0 and earlier	will be resolved in 6.11.1.0	**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	6.10.6.0 and earlier	resolved in 6.11.0.0	**Deficits in manifest files** Fields for ***ConSol CM***-Version and Build-Date are missing in the manifest.
627117	6.10.6.0 and earlier	open	**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.
627286	6.10.6.0 and earlier	open	**Issues with inline images in pasted text** Pasting (inline) images together with HTML text can still cause the images not being displayed, if the image link cannot be resolved later on a different computer. In this special case images are not included but linked and depending on the client computers specific network accessibility the link address may in some cases not be accessible from this machine. In a future release improved handling of this kind of image pasting will be implemented.
627768	6.10.6.0 and earlier	open	**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	6.10.6.0 and earlier	resolved in 6.11.0.0	***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.
628739	6.10.6.0 and earlier	resolved in 6.11.0.0	**Task execution framework exception handling issues** The task status of a task encountering an exception is not updated properly and thus shows an unexpected status.
629133	6.10.6.0 and earlier	open	**Data warehouse transfer error for long field names** A data warehouse data transfer without overwriting fails after a successful initialization with overwriting enabled. This happens for data field with long names which caused an unrecoverable error.

Number	Since	Resolved	Description
629704	6.10.6.0 and earlier	open	**Customer data model object deactivation inoperational** The functionality offered in Admin Tool to deactivate objects or elements in a customer data model is not working. Deactivated entries will still be available in their context.
630493	6.10.6.0 and earlier	open	**Two licenses consumed on clients using Windows 10 and Internet Explorer 11 with Single Sign-on** In the specific configuration of clients using Windows 10 and the Internet Explorer 11 in a Kerberos Single Sign-On environment normal access of ***ConSol CM*** consumes two client licenses instead of just one as on other platforms.
631148	6.10.6.0 and earlier	will be resolved in 6.10.7.0	**REST responses with stack traces** In case of an error while processing a request to the REST API there are cases when the response contains the stack trace of the exception. This information should never be in the response, but should only be logged.
631224	6.10.6.0 and earlier	open	**Setting a contact newly created from an Activity Control Form as main customer fails** After filling in an Activity Control Form with data for a new customer and creating this contact in the corresponding activity, setting this new contact as the main contact for the ticket within the same script fails with an exception. The exception results in a message in the ***Web Client***. The steps leading to this point succeed, however.

5. 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)	Certifcate expiry date
6.10.6.0 and higher	28.10.2019

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***!