# 

## #if (“$!{release.getCustomField(0)}” != “”)<center> Release date: #set($d = $textFormatter.getFieldValue($user, $release, “customField[0]”))$textFormatter.formatDate($user,$d,true) </center>#end

Produced by   
<center>#if (“$!{baseline}” == “”)Exported from <b>HEAD</b>#else Exported Baseline <b> $!{baseline.name}</b>#end</center>  
<p><center>${dateTimeFormatted}</center></p>

http://localhost:8080/cb/images/newskin/header/cblogo-xl.png

## Table of contents

Please right-click and choose "Update Field" to initialize this field.

#\*  
 This is a velocity template  
 Renders the „name” part of the exported entities  
\*#  
#set($tracker=${describable.tracker})  
#if (“$!{previousTracker}” == “” || ${previousTracker.id} != ${tracker.id})  
<h2>${tracker.project.name} - ${tracker.name}</h2>  
#end  
#set($previousTracker=$tracker)  
  
#if ("$!{describable.name}" != "")   
 #set($headingLevel = ${headingLevelProvider.apply(${entity})})  
 #if ("$!{headingLevel}" == "") #set($headingLevel = 3) #end  
 ## Word supports max 9 levels  
 #if (${headingLevel} > 9) #set($headingLevel = 9) #end  
  
 ## HTML heading level’s maximum is 6, in HTML <h7> is not present and ignored   
 #set($htmlHeadingLevel = $headingLevel)  
 #if ($htmlHeadingLevel > 6) #set($htmlHeadingLevel = 6) #end  
  
## Note: the “anchor” is used to jump between the exported issues if they refer (contain a link) to each other  
 #set($nameMarkup = “<a name=””$!{anchor}””></a>$!{requirementId.getRelease()} $!{describable.name}”)  
 #if($headingLevel > 6)  
 ## for deeper hierarchies above 4 levels that must be passed to Word document this way  
 #set($nameMarkup = “<p style=’mso-outline-level:${headingLevel};’>$nameMarkup</p>”)  
 #end  
  
 <h${htmlHeadingLevel}>$!{nameMarkup}</h${htmlHeadingLevel}>  
#end

#\*  
 This is a velocity template renders HTML fragment for Properties of entities  
\*#  
## Notification if issue changed in HEAD revision  
#if ((“$!{baseline}” != “”) && ($changedRequirementIdMap[$describable.id]))  
 <p class=”changed”>This issue has been changed in HEAD revision since $!{baseline.name} was created.<br /><br /></p>  
#end  
  
## load Properties specific macros:  
#parse("/html/mhtml-properties-macros.vm")  
  
#set($requirement = $describable)  
 ## render associations table  
 #if (${incomingAssociationsByIssue.containsKey($requirement.getId())} || ${outgoingAssociationsByIssue.containsKey($requirement.getId())})  
 <b>Associations:</b><br/>  
 <table class="requirement-props">  
 <tr>  
 <td class="field">From</td>  
 <td class="field">Association Type</td>  
 <td class="field">To</td>  
 </tr>  
 #foreach($entry in $outgoingAssociationsByIssue.get($requirement.getId()).entrySet())  
 <tr>  
 <td class="value"><i>&lt;this&gt;</i></td>  
 <td class="value">${entry.getKey()}</td>  
 <td class="value">  
 #foreach($item in $entry.getValue())  
 #if(${item.getKeyAndId()})[${item.getKeyAndId()}] <b>${item.getName()}</b><br/>  
 #else[${item.id}] <b>${item.getName()}</b><br/>#end   
 #end  
 </td>  
 </tr>  
 #end  
 #foreach($entry in $incomingAssociationsByIssue.get($requirement.getId()).entrySet())  
 <tr>  
 <td class="value">  
 #foreach($item in $entry.getValue())  
 #if(${item.getKeyAndId()})[${item.getKeyAndId()}] <b>${item.getName()}</b><br/>  
 #else[${item.id}] <b>${item.getName()}</b><br/>#end  
 #end  
 </td>  
 <td class="value">${entry.getKey()}</td>  
 <td class="value"><i>&lt;this&gt;</i></td>  
 </tr>  
 #end  
 </table>  
 <br/>  
 #end  
  
 ## render references table  
#if($issueReferences)  
 #set($incomingRefs = $issueReferences.getIncomingReferences($requirement))   
 #set($outgoingRefs = $issueReferences.getOutgoingReferences($requirement))  
 #if (! ($incomingRefs.isEmpty() && $outgoingRefs.isEmpty()))  
 <b>References:</b><br/>  
 <table class="requirement-props">  
 <tr>  
 <td class="field">From</td>  
 <td class="field">Field Name</td>  
 <td class="field">To</td>  
 </tr>  
 #foreach($ref in ${outgoingRefs.entrySet()})  
 #set($dto = ${ref.getValue()})  
 <tr>  
 <td class="value"><i>&lt;this&gt;</i></td>  
 <td class="value">${ref.key}</td>  
 <td class="value">  
 #foreach($version in $ref.value)  
 #set($dto = $version.dto)  
 #if ("${dto.id}" != "")[$!{dto.id}]#end <b>$!{dto.name}</b>  
 #end  
 </td>  
 </tr>  
 #end  
 #foreach($ref in $incomingRefs)  
 #set($dto = ${ref.dto})  
 <tr>  
 <td class=”value”>#if („${dto.id}” != „”)[$!{dto.id}]#end <b>$!{dto.name}</b></td>  
## Note: if you want a link then replace the content above:  
## #set ($url = „${cbBaseUrl}/${exporter.getUrlLink(${dto})}”)  
## <a href=”${url}”>${dto.name</a>  
 <td class="value">#set($sep=””) #foreach($referer in ${dto.referers}) ${sep} ${referer.label} #set($sep=”, ”) #end</td>  
 <td class="value"><i>&lt;this&gt;</i></td>  
 </tr>  
 #end  
 </table>  
<br/>  
#end  
#end ## issue’s references ENDS here  
## Render item/issue/requirement’s history if selected on the export GUI  
#if (${exportOptions.exportItemHistory})  
<div class=”exportItemHistory”><b>History:</b><br/>  
#set($historyRenderer = $springDispatcherServletContext.getBean(“trackerItemHistoryTableRenderer”))  
$!{historyRenderer.renderTrackerItemHistory($request, $user, $requirement)}  
</div>  
#end

|  |  |
| --- | --- |
| **CodeBeamer reference:** | <w:p><w:r>  <w:rPr>  <w:rStyle w:val="Hyperlink" />  <w:rFonts w:ascii="Arial" w:eastAsia="Times New Roman" w:hAnsi="Arial" w:cs="Arial"/>  <w:sz w:val="17"/>  <w:szCs w:val="17"/>   </w:rPr>  <w:fldChar w:fldCharType="begin" />  <w:instrText>HYPERLINK "$!{cbBaseUrl}/${urlLink}"</w:instrText>  <w:fldChar w:fldCharType="separate" />  <w:t>#if("$!{describable.keyAndId}" != "") $!{describable.keyAndId} #else ${interwikiLink} #end</w:t>  <w:fldChar w:fldCharType="end" /> </w:r></w:p> |
| **Revision:** | value |
| **Priority:** | Must have |
| **Business Value:** | Must have |
| **Severity:** | Must have |
| **Status:** | value |
| **Submitted at:** | <w:p><w:r>   <w:rPr>  <w:rFonts w:ascii="Arial" w:eastAsia="Times New Roman" w:hAnsi="Arial" w:cs="Arial"/>  <w:sz w:val="17"/>  <w:szCs w:val="17"/>   </w:rPr> <w:t> #if(“$!{describable.submittedAt}” != “”)${textFormatter.formatDate($user,$!{describable.submittedAt})}#end</w:t> </w:r></w:p> |
| **Submitted by:** | <w:p><w:r>   <w:rPr>  <w:rStyle w:val="Hyperlink" />  <w:rFonts w:ascii="Arial" w:eastAsia="Times New Roman" w:hAnsi="Arial" w:cs="Arial"/>  <w:sz w:val="17"/>  <w:szCs w:val="17"/>   </w:rPr>  #if(“$!{describable.submitter}” != “”) <w:fldChar w:fldCharType="begin" />  <w:instrText>HYPERLINK "$!{cbBaseUrl}/${describable.submitter.urlLink}"</w:instrText>  <w:fldChar w:fldCharType="separate" />  <w:t>${describable.submitter.name}</w:t>  <w:fldChar w:fldCharType="end" /> #end </w:r></w:p> |
| **Modified at:** | <w:p><w:r>   <w:rPr>  <w:rFonts w:ascii="Arial" w:eastAsia="Times New Roman" w:hAnsi="Arial" w:cs="Arial"/>  <w:sz w:val="17"/>  <w:szCs w:val="17"/>   </w:rPr> <w:t> #if(“$!{describable.modifiedAt}” != “”)${textFormatter.formatDate($user,$!{describable.modifiedAt})}#end</w:t> </w:r></w:p> |

#\*  
 This is a velocity template  
 Renders the „description” part of the exported entities  
\*#  
$wikiMarkupProcessor.transformToHtml(${describable.description}, ${describable.descriptionFormat}, false, false, $wikiContext)  
  
#\*   
 ## sample anchor for jumping to top of the current issue  
 #if ("$!{anchor}" != "") <div><a href="#$!{anchor}">Jump to top</a></div>#end  
\*#

cbContent

#\*   
 This is a velocity template  
 Allows customization of decoration around HTML parts  
\*#  
${defaultTemplate}

#\*  
 This is a velocity template  
 Allows customization of the CSS used in HTML parts  
\*#  
${defaultTemplate}  
  
#\*   
 An example: uncomment this below customizing color and font in tables and lists  
\*#  
#\*  
table td {  
 font-size: 32px !important;  
 font-weight: bold !important;  
 color: red !important;  
 font-family: courier !important;  
}  
  
ul li {  
 font-size: 32px !important;  
 font-weight: bold !important;  
 color: green !important;  
 font-family: courier !important;  
}  
\*#  
p.changed {  
 color: red !important;  
 font-weight: bold !important;  
 padding: 0 !important;  
 margin: 0 !important;  
}