



CPQ

2205 Breaking Changes and Announcements

Publish Date: January 28, 2022

Contents

- Disclaimer..... 3
- Revision History 3
- Release Dates..... 4
- Breaking Changes and Announcements^[Obj]..... 5
 - Automatic Cancellation of Scripts Exceeding 20 Minute Time Limit 5
 - Actions to Take..... 5
 - Character Limit Introduced for Scripts in 2205..... 6
 - Actions to Take..... 6
 - List as Line Item When Price is Zero Option on Container Attributes in 2205 Release 7
 - Actions to Take..... 7
 - New Business Partner Permissions Feature 7
 - Actions to Take..... 8
 - Try-Except-Continue Construct Forbidden in Scripting 8
 - Actions to Take..... 8
 - Involved Parties Page Breaking Change..... 9
 - Actions to Take..... 9
 - Scripting Message Breaking Change Fix..... 10
 - Actions to Take..... 10
 - HTML Data Binding Changed to Text in Responsive Templates 16
 - Actions to Take..... 17
 - Pricing Attribute Determination Fix for Quote 2.0 17
 - Actions to Take..... 18
 - Quote Error Messages Breaking Change Fix..... 18
 - Actions to Take..... 18

Disclaimer

This document is published and updated before the SAP CPQ releases are deployed, so it is subject to change. To ensure that you have the latest copy of this document, please check the Community for the most up-to-date version.

Revision History

Version	Date	Summary of Changes
1	January 28, 2022	The SAP CPQ 2205 Breaking Changes and Announcements is published.
2	March 18, 2022	<p>The document is updated with additional breaking changes and announcements:</p> <ul style="list-style-type: none">• New Business Partner Permissions Feature• Try-Except-Continue Construct Forbidden in Scripting• Involved Parties Page Breaking Change• Scripting Message Breaking Change Fix <p>In addition, a script is added to the note Character Limit Introduced for Scripts in 2205.</p>
3	April 11, 2022	An edit has been made in the note Involved Parties Page Breaking Change .
4	April 22, 2022	A line of code has been modified in the note Scripting Message Breaking Change Fix .
5	May 10, 2022	<p>The document is updated with additional breaking changes and announcements:</p> <ul style="list-style-type: none">• HTML Data Binding Changed to Text in Responsive Templates• Pricing Attribute Determination Fix for Quote 2.0• Quote Error Messages Breaking Change Fix

Release Dates

All Lower Environments (Development, Sandbox and Test):

- **Saturday, May 7, 2022**

Production:

- **Saturday, May 21, 2022**

Breaking Changes and Announcements

Automatic Cancellation of Scripts Exceeding 20 Minute Time Limit

With the 2205 release of SAP CPQ, IronPython scripts which take more than 20 minutes to execute will be automatically canceled. Consequently, the execution time for scripts will no longer exceed the timeout limit for the HTTP request, which is 20 minutes.

In the period prior to the 2205 release (May 21) starting with January, a warning is recorded in the log every time a script exceeds the 20-minute time limit.

Actions to Take

Please search your log for warnings caused by scripts exceeding the 20-minute time limit now to determine if such scripts exist on your environment.

Go to **Setup > Developer Tools > Log** (alternatively, on the user side of the application, click **Setup > Log**) and filter the log entries the following way:

- **Level: Warn**
- **Labels: Scripts**
- **Title: *Script execution timeout***

Once the filter is applied and the results display, you can find the names of scripts exceeding the time limit in the **Description** column.

If scripts exceeding the time limit exist on your environment, please optimize the script and shorten the execution time before the 2205 release. After the deployment of 2205 code on May 21, scripts which take longer than 20 minutes to execute will be automatically cancelled and will not execute successfully. In that scenario, a warning and an error will be recorded in the **Log**.

Character Limit Introduced for Scripts in 2205

As of the 2205 release of SAP CPQ, the size of IronPython scripts will be limited to 40,000 characters. New scripts created after the 2205 release will need to adhere to the introduced limit. Existing scripts will continue to work after the deployment of the 2205 release even if they exceed the character limit. However, if you edit those scripts, you won't be able to save the changes.

Actions to Take

Please review your existing scripts before the 2205 release and make sure that they aren't longer than 40,000 characters. You can use the character count feature of a text editor to review the number of characters in a script. Following that step, modify the scripts so that they don't exceed the character limit by breaking them down into smaller scripts or importing them as modules. In addition, if you plan to further expand any large scripts in the future, please adjust the plan in accordance with the character limit.

The character limit will be the default setting starting with 2205. However, if there is a significant interference with your existing scripts caused by the character limit, please contact the SAP CPQ Support team with your feedback.

You can use the following SQL script to locate scripts originating from any possible script source that are exceeding the character limit:

```
SELECT 'script name: ' + [Name] FROM [SystemEventScript] where LEN([Script]) > 40000
UNION ALL
SELECT 'action name: ' [ACTION_NAME] FROM [ACTIONS] where LEN(CAST([Script] AS NVARCHAR(MAX))) >
40000
UNION ALL
SELECT 'custom calculation name: ' + [Name] FROM [CartEditableMathDefn] where LEN(CAST([Script]
AS NVARCHAR(MAX))) > 40000
UNION ALL
SELECT 'product name: ' + [product_name] FROM [PRODUCTS] where LEN([Script]) > 40000
UNION ALL
select 'product name for attribute trigger: ' + [product_name] from
[attrTrigger] inner join products
on [attrTrigger].product_id = [products].PRODUCT_ID
where LEN(CAST([ActionListC] AS NVARCHAR(MAX))) > 40000
```

List as Line Item When Price is Zero Option on Container Attributes in 2205 Release

As of the 2205 release, when referenced by container attributes, products whose extended price is zero will be listed as line items on the quote only if the checkbox **List as line item when price is zero** is selected in the attribute values section of product attribute settings.

As the new behavior will become active by default in the 2205 release, you can make the necessary adaptations and test the effect of this change on product models in your environments during the 2202 release. This change is introduced only for the Quote 1.0 engine.

Actions to Take

In the 2205 release, the new behavior will become active by default on all tenants using Quote 1.0. However, you can test the new behavior of the checkbox **List as line item when price is zero** on container attributes during the 2202 release, in order to make the necessary adjustments on your environments.

Please contact the SAP CPQ Support team to enable the new behavior on your environment, and make sure to communicate your feedback to the SAP CPQ team.

New Business Partner Permissions Feature

As of the 2205 release, permissions for business partners will no longer be specified for each business partner individually, and the **Permissions** tab will be removed from business partner administration. Instead, you will be able to manage the permissions for multiple business partners at once, by creating rules using the new feature **Business Partner Permissions**.

Any currently used permissions which are defined on individual business partner level will no longer be valid as of the 2205 release, and you will only be able to use the new feature for managing business partner permissions after this time.

In addition, once the new permissions feature becomes available, initially no business partners will be visible to any users. As an initial solution, once 2205 is deployed, you can use the new

feature to create a simple rule as described below to make all business partners available to all users.

Actions to Take

We advise you not to define any additional business partner permissions for individual business partners until the deployment of SAP CPQ 2205, when you can use the new feature **Business Partner Permissions**, because you would need to set those permissions again after the release. Note that existing permissions defined on individual business partner level will no longer be available starting with 2205.

Once the 2205 release is deployed, initially no business partners will be visible to any users. Using the new feature, you will be able create a rule that makes all business partners visible to all users. To do that, create a new rule, enable the option **All business partners are visible to selected groups**, and then select all available permission groups.

Try-Except-Continue Construct Forbidden in Scripting

As of the 2205 release, the following construct will be forbidden in IronPython scripting in order to prevent issues in the system:

```
for j in range(1000):
    try:
        func
    except:
        continue
```

Scripts containing the provided construct will no longer work after the 2205 release.

Actions to Take

You need to replace this construct with a workaround solution whenever it occurs. The appropriate workaround is to replace `continue` with `pass`. If this workaround isn't viable for implementation, the construct from the following example can be used instead:


```
continueFlag = False
try:
    something
except:
    continueFlag = True
if continueFlag:
    continue
```

Involved Parties Page Breaking Change

Users will be provided with an optimized way to search for existing business partners in the **Involved Parties** quote tab in the 2205 release. Consequently, the options to add new or existing business partners to an involved party will be available in separate dialog windows. By selecting the option **Add Existing Business Partner** users will be able to search for business partners using all standard business partner fields as parameters. In addition, it will be possible to manually go through the field values and select them.

As a result of this improvement, the option to add an existing business partner will no longer be available in the **Add Business Partner** dialog. It will be only possible to create a new business partner via this specific dialog. If customizations have been created for the **Involved Parties** page, or the page was frozen to a previous version, the new option **Add Existing Business Partner** won't be accessible after the deployment of 2205 code. Since the previously existing option to add existing business partners via the **Add Business Partner** dialog will be removed, the option to add business partners won't be available in this scenario. Because of that, you need to implement a fix to retain the option to search for existing business partners when using frozen or customized templates.

However, if no customizations have been created for this page, there will be no issues regarding the changes to the dialog.

Actions to Take

If customizations have been created for the **Involved Parties** page before the 2205 release or the page was frozen to a previous version, you need to make sure to set default partner functions for your quotes before the release. By setting default partner functions, you will be able to search for existing business partners via the search field available next to the partner

functions. You can set default partner functions in partner function definition in the **Setup** by enabling the option **Show by default on new quotes**.

Scripting Message Breaking Change Fix

As of the 2205 release, scripting messages will be visible on all quote tabs in both quote engines, except Inline Frame custom tabs, where they are not supported. Prior to 2205, scripting messages were visible only on the **Quotation** tab.

If no customizations were created for any quote tabs or the **Sections/Notifications** template, and the **Cart** page wasn't frozen to the previous version, scripting messages will work correctly on all quote tabs after the 2205 version is live. However, if any tabs or the **Sections/Notifications** template have been customized, or the **Cart** page has been frozen, you need to implement a fix before the deployment of 2205 code to ensure there are no issues with the responsive templates after the deployment.

Actions to Take

If you have previously customized any of these pages: **CartDocumentPage**, **DealViewPage**, **InvolvedPartiesPage**, **TeamsPage**, **ShippingInfoPage**, **CustomFieldsPage**, you need to add the provided line of code to the customized template:

- `<scripting-messages></scripting-messages>`

The provided line of code should be copied to a specific location at the beginning of the template code. The beginning of these templates is provided here with the addition of the provided line, which is highlighted. When applying the fix, please add the line in the same location as here.

[CartDocumentsPage](#)

```
@using WebSite.Controllers
<div id="main" class="w960 row">
    <main class="col-xs-12 col-md-12 col-sm-12 user-page-container cart-documents-container fiori3-cart-tab-container">
        <div class="fiori3-documents-page clearfix approvals cart-content"
            id="cartDocumentsContainer" data-bind="visible: isLoading" style="display: none">
            <scripting-messages></scripting-messages>
            @Html.PartialWithNameAsComment("/Views/Cart/Sections/Tabs.cshtml")
            <div id="documentsTablesContainer" class="col-md-12 cart-documents-container-table">
```

DealViewPage

```
@using WebSite.Models
@model DealViewTableModel

<div class="w960 cart-width deal-view-container user-page-container fiori3-cart-tab-container"
style="height: calc(100% - 80px)">
  <div class="row cart" id="cartContainer" data-bind="css: { 'no-action-images':
!displayImagesForCartActions() } ">
    <div class="row cart" id="dealViewContainer">
      <div class="row-offcanvas row-offcanvas-left quote-page-main-container collapsed-actions-
visible" data-bind="visible: isInitialized" style="display: none;">
        <div class="col-xs-12 col-md-12 col-sm-12 clearfix" data-bind="css: { 'cart-view-
mode': isInViewMode }">
          <div class="clearfix"></div>
          <div class="cart-content clearfix">
            <scripting-messages></scripting-messages>
            @Html.PartialWithNameAsComment("Sections/Tabs")
          </div>
        </div>
      </div>
    </div>
  </div>
</div>
```

InvolvedPartiesPage

```
<div class="w960 cart-width involved-parties-container user-page-involved-party-container fiori3-
cart-tab-container">
  <div class="row cart" id="involvedPartiesContainer">
    <div class="row-offcanvas row-offcanvas-left quote-page-main-container collapsed-actions-
visible" data-bind="visible: isInitialized" style="display: none;">
      <div class="clearfix" data-bind="css: { 'cart-view-mode': isInViewMode }">
        <div class="clearfix"></div>
        <div class="row cart-content-involved-party clearfix">
          <scripting-messages></scripting-messages>
          @Html.PartialWithNameAsComment("Sections/Tabs")
        </div>
      </div>
    </div>
  </div>
</div>
```

TeamsPage

```
<main id="main" class="w960 row">
  <div class="col-xs-12 col-md-12 col-sm-12 user-page-container cart-documents-container
fiori3-cart-tab-container">
    <div class="clearfix approvals cart-content" id="cartTeamsContainer">
      <div id="cartTabsContainer">
        <scripting-messages></scripting-messages>
        @Html.PartialWithNameAsComment("/Views/Cart/Sections/Tabs.cshtml")
      </div>
    </div>
  </div>
</main>
```

ShippingInfoPage

```
@using WebSite.Controllers
<div class="w960">
  <div class="col-xs-12 col-md-12 col-sm-12 clearfix user-page-container shipping-info-page-
container fiori3-shipping-info-page-container fiori3-cart-tab-container">
    <div id="shippingCustomerContainer" class="shipping-info-container fiori3-shipping-info-
container">
    </div>
  </div>
</div>
```

```
<scripting-messages></scripting-messages>
@Html.PartialWithNameAsComment("/Views/Cart/Sections/Tabs.cshtml")
```

CustomFieldsPage

```
@using WebSite.Controllers
<main class="w960">
    <div class="row user-page-container custom-fields-page-container fiori3-cart-tab-container
    cart-content fiori3-cart-tab-container-with-footer" id="cartContainer">
        <div class="row" data-bind="visible: isInitialized" style="display: none;">
            <div data-bind="css: { 'cart-view-mode': isInViewMode }">
                <div class="clearfix"></div>

                @*Custom fields*@
                <div class="clearfix">
                    <scripting-messages></scripting-messages>
                    @Html.PartialWithNameAsComment("Sections/Tabs")
```

CartPage

If you have previously customized the template **CartPage**, you need to add the provided line of code to the customized template:

- <scripting-messages params="quoteMessages: \$root.quoteMessages"></scripting-messages>

Add the provided line of code in the location highlighted below in the beginning of the template.

```
@using WebSite.Controllers
<main class="w960 cart-width quote-page-container fiori3-cart-page-container fiori3-cart-tab-
container">
    <div class="row cart" id="cartContainer" data-bind="css: { 'no-action-images':
!cartActionsModel().displayImagesForCartActions() } ">
        <div class="row row-offcanvas row-offcanvas-left quote-page-main-container collapsed-
actions-visible" data-bind="visible: isInitialized, css: { 'active': areActionsActive }"
style="display: none;">
            <div class="clearfix cart-wrapper" data-bind="css: { 'cart-view-mode': isInViewMode
}">

                <div class="clearfix"></div>
                @Html.PartialWithNameAsComment("Sections/Promotions")
                <scripting-messages params="quoteMessages: $root.quoteMessages"></scripting-
messages>
                @Html.PartialWithNameAsComment("Sections/Notifications")
                @Html.PartialWithNameAsComment("Sections/InvalidAutocompleteNotification")
```

Sections/Notifications

If any customizations were created for the **Sections/Notifications** page, you need to delete the highlighted lines of code before the deployment of 2205:

```
<!-- ko if: $root.useNewQuoteEngine -->
<div class="row" data-bind="foreach: scriptingMessages">
    <div class="fiori3-alert fiori3-alert-info fiori3-alert-info-border" data-bind="visible:
```

```

scriptingMessage.Type == 1">
  <span class="sap-icon">&#xe202;</span>
  <span data-bind="html: scriptingMessage.Content"></span>
  <span class="sap-icon sap-icon-scripting-messages-remove" data-bind="visible:
!scriptingMessage.IsTemporary, click: function() {
$root.removeScriptingMessage(scriptingMessage)}">&#xe03e;</span>
</div>
  <div class="fiiori3-alert fiiori3-alert-warning fiiori3-alert-warning-border" data-
bind="visible: scriptingMessage.Type == 2">
  <span class="sap-icon">&#xe201;</span>
  <span data-bind="html: scriptingMessage.Content"></span>
  <span class="sap-icon sap-icon-scripting-messages-remove" data-bind="visible:
!scriptingMessage.IsTemporary, click: function() {
$root.removeScriptingMessage(scriptingMessage)}">&#xe03e;</span>
</div>
  <div class="fiiori3-alert fiiori3-alert-danger fiiori3-alert-danger-border" data-bind="visible:
scriptingMessage.Type == 3">
  <span class="sap-icon">&#xe0b1;</span>
  <span data-bind="html: scriptingMessage.Content"></span>
  <span class="sap-icon sap-icon-scripting-messages-remove" data-bind="visible:
!scriptingMessage.IsTemporary, click: function() {
$root.removeScriptingMessage(scriptingMessage)}">&#xe03e;</span>
</div>
  <div class="fiiori3-alert fiiori3-alert-success fiiori3-alert-success-border" data-
bind="visible: scriptingMessage.Type == 4">
  <span class="sap-icon">&#xe203;</span>
  <span data-bind="html: scriptingMessage.Content"></span>
  <span class="sap-icon sap-icon-scripting-messages-remove" data-bind="visible:
!scriptingMessage.IsTemporary, click: function() {
$root.removeScriptingMessage(scriptingMessage)}">&#xe03e;</span>
</div>
</div>
<!--/ko -->
<div class="row">
  <div class="fiiori3-alert fiiori3-alert-danger" data-bind="visible: approval.errorMessage,
html: approval.errorMessage"></div>
  <div class="fiiori3-alert fiiori3-alert-warning" data-bind="visible: messageTitle() ||
approval.conditions().length > 0 || cartUsage().length > 0 || (scriptingMessages().length > 0 &&
!$root.useNewQuoteEngine) || landingMessages().length > 0 || errorMessage()">
    <sap class="sap-icon">&#xe053;</sap>
    <div>
      <!-- ko ifnot: $root.useNewQuoteEngine -->
      <p class="cart-warning-group" data-bind="foreach: scriptingMessages, style: {
display: scriptingMessages().length > 0 ? 'block' : 'none' }">
        <span data-bind="html: scriptingMessage.Content"></span><br />
      </p>
      <!--/ko -->
      <div class="cart-warning-group" data-bind="style : { display: landingMessageTitle() ?
'block' : 'none' }">
        <p data-bind="html: landingMessageTitle"></p>
        <ul data-bind="foreach: landingMessages">
          <li><span data-bind="text: title"></span><span data-bind="if: description">:
</span><span data-bind="text: description"></span></li>
        </ul>
      </div>
      <div class="cart-warning-group" data-bind="style : { display: messageTitle() ?
'block' : 'none' }">
        <p data-bind="html: messageTitle"></p>
        <ul data-bind="foreach: messageItems">
          <li><span data-bind="text: title"></span>: <span data-bind="
text:
description"></span></li>
        </ul>
      </div>
      <div class="cart-warning-group" data-bind="style : { display: errorMessage() ?
'block' : 'none' }">
        <span data-bind="html: errorMessage"></span><br />

```

```

        </div>
        <div class="cart-warning-group" data-bind="style : { display:
approval.conditions().length > 0 ? 'block' : 'none' }">
            <!-- ko template: { name: 'ApprovalTemplate', data: approval } -->
            <!--/ko -->
        </div>
        <div class="cart-warning-group" data-bind="style : { display: cartUsage().length > 0
? 'block' : 'none' }">
            <p>@Translation.Get("Cart.RecentlyOpened")</p>
            <ul data-bind="foreach: cartUsage">
                <li><span data-bind="text: user"></span><span data-bind="
text:
openedMinutesAgo"></span></li>
            </ul>
        </div>
    </div>
</div>
</div>

@Html.PartialWithNameAsComment("Templates/ApprovalTemplate")

```

Cart Page Frozen to Previous Version

If the **Cart** page has been frozen to the previous version, issues will occur with the **Quotation** tab on the quote. To prevent this from happening, you need to create a customization for the **Sections/Notifications** and **CartPage** templates. However, if you have already customized these templates, you need to add and delete the lines of code as described in the previous sections.

If you are adding a new customization, provided here are the complete new templates which you can use.

Cart Page – Complete Template

```

@using WebSite.Controllers
<main class="w960 cart-width quote-page-container fiori3-cart-page-container fiori3-cart-tab-
container">
    <div class="row cart" id="cartContainer" data-bind="css: { 'no-action-images':
!cartActionsModel().displayImagesForCartActions() } ">
        <div class="row row-offcanvas row-offcanvas-left quote-page-main-container collapsed-
actions-visible" data-bind="visible: isInitialized, css: { 'active': areActionsActive }"
style="display: none;">
            <div class="clearfix cart-wrapper" data-bind="css: { 'cart-view-mode': isInViewMode
}">
                <div class="clearfix"></div>
                @Html.PartialWithNameAsComment("Sections/Promotions")
                <scripting-messages params="quoteMessages: $root.quoteMessages"></scripting-
messages>
                @Html.PartialWithNameAsComment("Sections/Notifications")
                @Html.PartialWithNameAsComment("Sections/InvalidAutocompleteNotification")
                <div class="cart-content fiori3-cart-content clearfix">
                    @Html.PartialWithNameAsComment("Sections/Tabs")
                </div>
            </div>
        </div>
    </div>
    <div id="cartSectionsWrapper" class="cart-sections-wrapper">
        @Html.PartialWithNameAsComment("Sections/Proposal")
        @{}
    </div>

```

```

        var i = 0;
        IList<string> sections = CartController.GetSections();
        while (i < sections.Count)
        {
            Html.RenderPartialWithNameAsComment("Sections/" + sections[i]);
            i += 1;
        }
    }
</div>
@if (CartController.CanAccessCartDebugger())
{
    @Html.PartialWithNameAsComment("Debugger")
}
</div>
@Html.Partial("~/Views/Favourites/AddToFavoritesDialogTemplate.cshtml")

<!-- ko if: promosInfo.shouldShowPromotionsInfo -->

@Html.PartialWithNameAsComment("/Views/Cart/Dialogs/PromotionDialogTemplate.cshtml")
<!-- /ko -->
<!-- ko with: filterEditor -->
<div style="position: fixed; left: 0; right: 0; top: 0;bottom: 0"></div>

    <div id="filter-editor-content" class="quote-items-filter-editor fiori3-cart-
filter slide-in-from-left" data-bind="affixTop: { disableScroll: true }">
        <div class="quote-filter-header" style="border-bottom: 1px solid #ededed">
            <h4 data-bind="text: $root.filtersLabel"></h4>
            <span class="close-filter-editor">
                <i tabindex="0" class="fa fa-times" data-bind="click:
$parent.toggleFilterEditor, event: { keyup: function() { $parent.toggleFilterEditorOnEnter(event)
} }, tooltip: { title: '@Translation.GetJS("Msg.Close")' }, attr: { 'aria-label':
'@Translation.GetJS("Msg.Close")' }"></i>
            </span>
        </div>
        <div id="quoteFilterContainer" class="quote-list-editors cpq-scroll" data-
bind="foreach: filters">
            <div data-bind="template: 'Filter_' + template" class="fiori3-input-
group"></div>
        </div>
        <div class="quote-list-tab-footer">
            <button class="btn btn-primary fiori3-btn-primary" data-bind="click:
applyFilters, enable: actionEnabled(), executeOnEnter: function() { applyFilters();
}">@Translation.Get("QuoteList.ApplyFilter")</button>
            <button class="btn btn-tertiary fiori3-btn-tertiary" data-bind="click:
$parent.toggleFilterEditor, event: { keyup: function() { $parent.toggleFilterEditorOnEnter(event)
} }">@Translation.Get("QuoteList.Cancel")</button>
        </div>
    </div>
<!-- /ko -->
</div>
</div>
</main>

@Html.PartialWithNameAsComment("Templates/CellTemplates")
@Html.PartialWithNameAsComment("Templates/QuoteFilterTemplates")
@Html.PartialWithNameAsComment("Templates/TotalsTemplate")
@Html.PartialWithNameAsComment("/Views/CartList/Dialogs/ReassignCartDialogTemplate.cshtml")
@Html.PartialWithNameAsComment("/Views/Cart/Dialogs/PricebookMissingForMarketDialogTemplate.cshtml")
@Html.PartialWithNameAsComment("/Views/Cart/Dialogs/CreatePartialOrderDialogTemplate.cshtml")
@Html.PartialWithNameAsComment("Dialogs/UpgradeProductsProgressBarTemplate")
@Html.PartialWithNameAsComment("SliderTemplate")

```

Sections/Notifications – Complete Template

```
<div class="row">
```

```

    <div class="fiori3-alert fiori3-alert-danger" data-bind="visible: approval.errorMessage,
html: approval.errorMessage"></div>
    <div class="fiori3-alert fiori3-alert-warning" data-bind="visible: messageTitle() ||
approval.conditions().length > 0 || cartUsage().length > 0 || landingMessages().length > 0 ||
errorMessage()">
        <sap class="sap-icon">&#xe053;</sap>
    <div>
        <div class="cart-warning-group" data-bind="style : { display: landingMessageTitle() ?
'block' : 'none' }">
            <p data-bind="html: landingMessageTitle"></p>
            <ul data-bind="foreach: landingMessages">
                <li><span data-bind="text: title"></span><span data-bind="if: description">:
</span><span data-bind="text: description"></span></li>
            </ul>
        </div>
        <div class="cart-warning-group" data-bind="style : { display: messageTitle() ?
'block' : 'none' }">
            <p data-bind="html: messageTitle"></p>
            <ul data-bind="foreach: messageItems">
                <li><span data-bind="text: title"></span>: <span data-bind="
description"></span></li>
            </ul>
        </div>
        <div class="cart-warning-group" data-bind="style : { display: errorMessage() ?
'block' : 'none' }">
            <span data-bind="html: errorMessage"></span><br />
        </div>
        <div class="cart-warning-group" data-bind="style : { display:
approval.conditions().length > 0 ? 'block' : 'none' }">
            <!-- ko template: { name: 'ApprovalTemplate', data: approval } -->
            <!--/ko -->
        </div>
        <div class="cart-warning-group" data-bind="style : { display: cartUsage().length > 0
? 'block' : 'none' }">
            <p>@Translation.Get("Cart.RecentlyOpened")</p>
            <ul data-bind="foreach: cartUsage">
                <li><span data-bind="text: user"></span><span data-bind="
openedMinutesAgo"></span></li>
            </ul>
        </div>
    </div>
</div>
</div>
@Html.PartialWithNameAsComment("Templates/ApprovalTemplate")

```

HTML Data Binding Changed to Text in Responsive Templates

Data binding for the product description field has been changed from HTML to text in the following templates:

- **Catalog > CatalogPage**
- **CompareProducts > CompareProductsTemplate**
- **Favorites > FavoritesPage**
- **ProductDetails > ProductDetailsTemplate**

This change improves the overall security of SAP CPQ and prevents potential cross-site scripting.

Depending on whether administrators have customized these templates before this release or not, they need to take the following actions:

- If these templates were customized before this release, there will be no breaking changes in the model as the HTML binding will continue to work as before this release. However, we strongly advise that administrators manually change the binding to text, as explained in [Actions to Take](#).
- If HTML was used in the product description field, there will be a breaking change in the model after the 2205 release. To prevent breaking the model, administrators need to manually change the binding from text to HTML, as explained in [Actions to Take](#).
- If these templates weren't customized before, nor was HTML used in the product description field, administrators don't need to take any action.

Actions to Take

To change the data binding of the description field, open each template (**Catalog > CatalogPage, CompareProducts > CompareProductsTemplate, Favorites > FavoritesPage and ProductDetails > ProductDetailsTemplate**) and find where data binding is defined for the description. Then replace `html` with `text`.

Old Binding	New Binding
<code><div data-bind="html: description"></div></code>	<code><div data-bind="text: description"></div></code>

Pricing Attribute Determination Fix for Quote 2.0

The logic behind the determination of pricing attributes is modified in the Quote 2.0 engine for the 2205 release. When using a Quote 2.0 environment in which the Variant Configuration integration is enabled, it is necessary to implement a fix before the deployment of 2205 code, to

ensure that the information in quotes remains correct. The behavior remains unchanged on environments using Quote 1.0.

Actions to Take

If the Variant Configuration integration is enabled on a Quote 2.0 environment, prior to 2205, it is necessary to review the pricing attributes that are used on a created quote in **Setup > Providers > Providers > SAP General Attribute Mappings**. The administrator needs to make note of all the quote custom fields and quote item custom fields referenced in the **Value** column of pricing attribute definition.

Following that step, it is necessary to click **Set Determination Events** and move all the fields that are referenced in the **Value** column of pricing attribute definition to the column **Selected Fields/Partner Functions**. As of 2205, the **Set Determination Events** option will become available on all Quote 2.0 environments using the Variant Configuration integration.

By performing this fix, the administrator ensures that pricing attribute determination is triggered every time any of those fields is edited. Consequently, the quote context remains up to date and the pricing information remains correct.


Quote Error Messages Breaking Change Fix

More than one error message can now display on the quote at the same time when using the Quote 2.0 engine. Due to this improvement, administrators who have previously created any kind of quote messages using IronPython scripting need to implement the fix described below for the messages to continue working correctly.

Actions to Take

If any messages appearing on the quote were created using scripting, it is necessary to replace the enumeration `MessageType` with `MessageLevel` in all these scripts. Scripts used for quote messages that are using the enumeration `MessageType` will no longer work starting with the 2205 release.

The following example shows how to create a script for adding a quote message with the correct enumeration `MessageLevel`:



The screenshot shows a web-based interface titled "Script Workbench". At the top, there is a dark blue header with the text "Script Workbench". Below the header, there is a blue button with a play icon and the text "Run". To the right of the button is a label "Mode" followed by a dropdown menu currently set to "Standard". Below these controls is a code editor area with a light gray background. The code editor contains two lines of Python code:

```
1 from Scripting.Quote import MessageLevel
2 context.Quote.AddMessage("Info Msg", MessageLevel.Info, True)
```