Configuration File Changes

Sitecore Experience Platform 8.0 Update-3

sitecore®
Own the experience™

Sitecore Experience Platform 8.0 Update-3

1.1 Config file changes in Sitecore 8.0 rev. 150427 (Update-3) compared to Sitecore 8.0 rev. 150223 (Update-2).

This update requires you to make changes to the following files:

- Web.config
- App_Config\Include\Sitecore.ExperienceEditor.config
- App_Config\Include\Sitecore.MvcExperienceEditor.config
- App_Config\Include\ContentTesting\Sitecore.ContentTesting.config

Depending on search provider, you should also update the appropriate files below:

- Sitecore solution configured to use Lucene:
  - App_Config\Include\Sitecore.ContentSearch.Lucene.DefaultIndexConfiguration.config
  - App_Config\Include\ContentTesting\Sitecore.ContentTesting.Lucene.IndexConfiguration.config
- Sitecore solution configured to use Solr:
  - App_Config\Include\Sitecore.ContentSearch.Solr.DefaultIndexConfiguration.config.example

In this update the following file was added:

- App_Config\Include\Sitecore.Diagnostics.config

The clean configuration files for Sitecore 8.0 rev. 150427 (Update-3) are provided in the following packages:

- Web.config_80rev150427.zip
- app_config_files_80rev150427.zip

  To download these files, on the Sitecore Experience Platform 8.0 Update-3 page, in the **Upgrade options** section, click *Config files for upgrade*.

1.1.1 Changes to Web.config

Tip: Most customers have only applied a small number of changes (if any) to the `Web.config` file. In this case, we recommend that you download the clean config file using the link above and re-apply any changes that you have made to save time and to reduce the risk of making mistakes when adding new settings, pipelines, etc. to the file.

The following changes have been made to the `Web.config` file of Sitecore 8.0 rev. 150427 (Update-3) as compared to the `Web.config` file of Sitecore 8.0 rev. 150223 (Update-2).

1. In the **<appSettings>** section add the following key after `key="Lucene.Net.FSDirectory.class"` key:

```
<add key="PageInspector:ServerCodeMappingSupport" value="Disabled" />
```

2. In the **<sitecore>** section, add the following node (after the **<itemManager>** node):

```
<!-- TEMPLATES -->
    <templateManager defaultProvider="default">
      <providers>
        <clear />
        <add name="default" type="Sitecore.Data.Managers.TemplateProvider,
Sitecore.Kernel" />
      </providers>
    </templateManager>
```

3. In the **<scheduling>** section, in the agent of type `"Sitecore.Tasks.CleanupAgent"` change the following node into the `<files hint="raw:AddCommand" … />` from:

```
<remove folder="$(dataFolder)/logs" pattern="log.*.txt" maxCount="20"
minAge="7.00:00:00" />
```

   to:

```
        <remove folder="$(dataFolder)/logs" pattern="*log.*.txt" maxAge="30.00:00:00" />
```

4. In the `<settings>` section, add the following setting element after the `<setting name="RecycleBinActive" … />`:

```
        <!-- RENDERING - HTML ENCODED FIELD TYPES
             Specifies a pipe-separated list of field types that should be HTML encoded
when rendered by the <renderField> pipeline.
             Default value: text|single-line text
        -->
        <setting name="Rendering.HtmlEncodedFieldTypes" value="text|single-line text" />
```

1.1.2   Changes to App_Config\Include\Sitecore.ExperienceEditor.config

Tip: Most customers have only applied a small number of changes (if any) to the `Sitecore.ExperienceEditor.config` file. In this case, we recommend that you download the clean config file using the link above and re-apply any changes that you've made to save time and to reduce the risk of making mistakes when adding new settings, pipelines, etc. to the file.

The following changes have been made to the `App_Config\Include\Sitecore.ExperienceEditor.config` file of Sitecore 8.0 rev. 150427 (Update-3) as compared to the `App_Config\Include\Sitecore.ExperienceEditor.config` file of Sitecore 8.0 rev. 150223 (Update-2).

1. Add the following node in **<commands>** section (you can add it at any place in the section, but in a clean Sitecore installation the new node is placed after the `webedit:editlink` command):

```
    <command name="webedit:clearlink"
type="Sitecore.Shell.Applications.WebEdit.Commands.ClearLink, Sitecore.ExperienceEditor"/>
```

1.1.3   Changes to App_Config\Include\Sitecore.MvcExperienceEditor.config

Tip: Most customers have only applied a small number of changes (if any) to the `Sitecore.MvcExperienceEditor.config` file. In this case, we recommend that you download the clean config file using the link above and re-apply any changes that you've made to save time and to reduce the risk of making mistakes when adding new settings, pipelines, etc. to the file.

The following changes have been made to the `App_Config\Include\Sitecore.MvcExperienceEditor.config` file of Sitecore 8.0 rev. 150427 (Update-3) as compared to the `App_Config\Include\Sitecore.MvcExperienceEditor.config` file of Sitecore 8.0 rev. 150223 (Update-2).

1. Add the following processor (you can add it at any place in the section, but in a clean Sitecore installation the new node is placed after the `Sitecore.Mvc.ExperienceEditor.Pipelines.Response.CustomizeRendering.SkipIfDesigning` processor):

```
    <processor
type="Sitecore.Mvc.ExperienceEditor.Pipelines.Response.CustomizeRendering.SelectVariation,
Sitecore.Mvc.ExperienceEditor"
patch:before="processor[@type='Sitecore.Mvc.Analytics.Pipelines.Response.CustomizeRendering.Se
lectVariation, Sitecore.Mvc.Analytics']" />
```

1.1.4   Changes to App_Config\Include\ContentTesting\Sitecore.ContentTesting.config

Tip: Most customers have only applied a small number of changes (if any) to the `Sitecore.ContentTesting.config` file. In this case, we recommend that you download the clean config file using the link above and re-apply any changes that you've made to save time and to reduce the risk of making mistakes when adding new settings, pipelines, etc. to the file.

The following changes have been made to the `App_Config\Include\ContentTesting\Sitecore.ContentTesting.config` file of Sitecore

8.0 rev. 150427 (Update-3) as compared to the
`App_Config\Include\ContentTesting\Sitecore.ContentTesting.config` file of Sitecore
8.0 rev. 150223 (Update-2).

1. In the **<settings>** section, add the following setting (you can add it at any place in the section, but in a clean Sitecore installation the new node is placed after the `ContentTesting.AutomaticContentTesting.Enabled` setting):

```
<!-- CONTENT TESTING - CACHING - ACTIVE TEST CACHE MAXSIZE
     Specifies the maximum size of the ActiveTestCache cache.
     Default value: 512KB
-->
<setting name="ContentTesting.Caching.ActiveTestCache.MaxSize" value="512KB" />
```

2. In the **<settings>** section, add the following setting (you can add it at any place in the section, but in a clean Sitecore installation the new node is placed after the `ContentTesting.MinimumDuration` setting):

```
<!-- CONTENT TESTING - PHANTOM JS - ENABLE DISK CACHE
     Determines whether the PhantomJS tool should use disk caching for web
resources.
     Default value: true
-->
<setting name="ContentTesting.PhantomJS.EnableDiskCache" value ="false"/>
```

3. In the **<settings>** section, remove the following setting:

```
<!-- CONTENT TESTING - SCREENSHOT DOMAIN
     The domain to use when generating screenshots if running outside a request
and if the item's site doesn't have a target hostname.
     -->
<setting name="ContentTesting.ScreenshotDomain" value="http://localhost"/>
```

4. In the **<settings>** section, add the following setting (you can add it at any place in the section, but in a clean Sitecore installation the new node is placed after the `ContentTesting.PhantomJS.LockNavigation` setting):

```
<!-- CONTENT TESTING - PHANTOM JS - TIMEOUT
     The timeout in milliseconds used when executing calls to PhantomJS
     Default value: 60000
-->
<setting name="ContentTesting.PhantomJS.Timeout" value ="60000"/>
```

5. In the **<events>** section, add the following event in the end of the section:

```
<!-- ACTIVE TEST CACHE CLEARER
     This event handler clears data from the ActiveTest cache
-->
<event name="publish:end">
  <handler type="Sitecore.ContentTesting.Events.ActiveTestCacheClearer,
Sitecore.ContentTesting" method="Process"/>
```

6. In the **<events>** section comment all the events, except the `publish:end` event in the end of the section. After this, the whole events section should look like:

```
<events>
     <!-- PRE-EMPTIVE SCREENSHOT GENERATION - LOCAL
          Enable the following event handler to generate screenshot files as items
change thus reducing wait time
          within the UI when screenshots are displayed.
          This handler only works on CM instances. Never enable it on any other
Sitecore instance.
     -->
     <!--<event name="item:saved">
       <handler type="Sitecore.ContentTesting.Events.GenerateScreenshot,
Sitecore.ContentTesting" method="OnItemSaved">
          <excludeFields hint="list:ExcludeFieldFromComparison">
            <created>  created</created>
            <createdby>__created by</createdby>
            <updated>__updated</updated>
            <updatedby>__updated by</updatedby>
            <revision>__revision</revision>
            <sortorder>  sortorder</sortorder>
            <validfrom>  valid from</validfrom>
            <workflow>__workflow</workflow>
```

```
                    <workflowstate>__workflow state</workflowstate>
                    <lock>  lock</lock>
                    <pageleveltestsetdefinition>  Page Level Test Set
Definition</pageleveltestsetdefinition>
                </excludeFields>
            </handler>
        </event>-->
        <!-- PRE-EMPTIVE SCREENSHOT GENERATION - REMOTE
            Enable the following event handler to generate screenshot files as remote
items change thus reducing wait time
            within the UI when screenshots are displayed.
            This handler only works on CM instances. Never enable it on any other
Sitecore instance.
        -->
        <!--<event name="item:saved:remote">
            <handler type="Sitecore.ContentTesting.Events.GenerateScreenshot,
Sitecore.ContentTesting" method="OnRemoteItemSaved">
                <excludeFields hint="list:ExcludeFieldFromComparison">
                    <created>  created</created>
                    <createdby>  created by</createdby>
                    <updated>  updated</updated>
                    <updatedby>__updated by</updatedby>
                    <revision>__revision</revision>
                    <sortorder>  sortorder</sortorder>
                    <validfrom>  valid from</validfrom>
                    <workflow>  workflow</workflow>
                    <workflowstate>__workflow state</workflowstate>
                    <lock>__lock</lock>
                    <pageleveltestsetdefinition>  Page Level Test Set
Definition</pageleveltestsetdefinition>
                </excludeFields>
            </handler>
        </event>-->
        <!-- ACTIVE TEST CACHE CLEARER
            This event handler clears data from the ActiveTest cache
        -->
        <event name="publish:end">
            <handler type="Sitecore.ContentTesting.Events.ActiveTestCacheClearer,
Sitecore.ContentTesting" method="Process"/>
        </event>
    </events>
```

7. In the `<getCurrentTestCombination>` node of the **<pipelines>** section, remove the following processor:

```
    <processor
type="Sitecore.ContentTesting.Pipelines.GetCurrentTestCombination.GetQueryStringCombination,
Sitecore.ContentTesting"/>
```

8. In the **<pipelines>** section, remove the following node:

```
    <!-- GET ITEM TESTS
            Inspect an item and get tests for it
        -->
        <!--<getItemTests>
            <processor
type="Sitecore.ContentTesting.Pipelines.GetItemTests.GetPageLevelTest,
Sitecore.ContentTesting"/>
            <processor
type="Sitecore.ContentTesting.Pipelines.GetItemTests.GetComponentTest,
Sitecore.ContentTesting"/>
            <processor
type="Sitecore.ContentTesting.Pipelines.GetItemTests.GetPersonalizationTest,
Sitecore.ContentTesting"/>
            <processor type="Sitecore.ContentTesting.Pipelines.GetItemTests.GetReferrers,
Sitecore.ContentTesting"/>
        </getItemTests>-->
```

1.1.5 Changes to App_Config\Include\Sitecore.ContentSearch.Lucene.DefaultIndexConfiguration.config

Tip: Most customers have only applied a small number of changes (if any) to the `Sitecore.ContentSearch.Lucene.DefaultIndexConfiguration.config` file. In this case, we recommend that you download the clean config file using the link above and re-apply any changes that you've made to save time and to reduce the risk of making mistakes when adding new settings, pipelines, etc. to the file.

The following changes have been made to the `App_Config\Include\Sitecore.ContentSearch.Lucene.DefaultIndexConfiguration.config` file of Sitecore 8.0 rev. 150427 (Update-3) as compared to the `App_Config\Include\Sitecore.ContentSearch.Lucene.DefaultIndexConfiguration.config` file of Sitecore 8.0 rev. 150223 (Update-2).

1. In the **<fieldNames>** section, add the following node, at the beginning of the section:

```
        <field fieldName="culture"              storageType="YES" indexType="TOKENIZED"
vectorType="NO" boost="1f" type="System.String"
settingType="Sitecore.ContentSearch.LuceneProvider.LuceneSearchFieldConfiguration,
Sitecore.ContentSearch.LuceneProvider">
                        <analyzer
type="Sitecore.ContentSearch.LuceneProvider.Analyzers.LowerCaseKeywordAnalyzer,
Sitecore.ContentSearch.LuceneProvider" />
                </field>
```

2. In the **<fields>** section, add the following node, (you can add it at any place in the section, but in a clean Sitecore installation the new node is placed between the `parsedlanguage` and `site` fields):

```
        <field fieldName="culture"
>Sitecore.ContentSearch.ComputedFields.Culture,Sitecore.ContentSearch</field>
```

3. In the **<settings>** section, change the value of the `ContentSearch.LuceneQueryClauseCount` setting from 1024 to 4096. After the change setting should look like this:

```
        <setting name="ContentSearch.LuceneQueryClauseCount" value="4096" />
```

1.1.6 Changes to App_Config\Include\ContentTesting\Sitecore.ContentTesting.Lucene.IndexConfiguration.config

Tip: Most customers have only applied a small number of changes (if any) to the `Sitecore.ContentTesting.Lucene.IndexConfiguration.config` file. In this case, we recommend that you download the clean config file using the link above and re-apply any changes that you've made to save time and to reduce the risk of making mistakes when adding new settings, pipelines, etc. to the file.

The following changes have been made to the `App_Config\Include\ContentTesting\Sitecore.ContentTesting.Lucene.IndexConfiguration.config` file of Sitecore 8.0 rev. 150427 (Update-3) as compared to the `App_Config\Include\ContentTesting\Sitecore.ContentTesting.Lucene.IndexConfiguration.config` file of Sitecore 8.0 rev. 150223 (Update-2).

1. In the **<fieldNames>** section, add the following analyzer to the **_itemuri** field:

```
        <analyzer
type="Sitecore.ContentSearch.LuceneProvider.Analyzers.LowerCaseKeywordAnalyzer,
Sitecore.ContentSearch.LuceneProvider" />
```

After this the whole **_itemuri** field node should look like this:

```
        <field fieldName="_itemuri" storageType="YES" indexType="TOKENIZED" vectorType="NO"
boost="1f" type="System.String"
```

```
settingType="Sitecore.ContentSearch.LuceneProvider.LuceneSearchFieldConfiguration,
Sitecore.ContentSearch.LuceneProvider">
                                <analyzer
type="Sitecore.ContentSearch.LuceneProvider.Analyzers.LowerCaseKeywordAnalyzer,
Sitecore.ContentSearch.LuceneProvider" />
                              </field>
```

1.1.7 Changes to App_Config\Include\ Sitecore.ContentSearch.Solr.DefaultIndexConfiguration.config.example

Tip: Most customers have only applied a small number of changes (if any) to the `Sitecore.ContentSearch.Solr.DefaultIndexConfiguration.config.example` file. In this case, we recommend that you download the clean config file using the link above and re-apply any changes that you've made to save time and to reduce the risk of making mistakes when adding new settings, pipelines, etc. to the file.

The following changes have been made to the `App_Config\Include\Sitecore.ContentSearch.Solr.DefaultIndexConfiguration.config.example` file of Sitecore 8.0 rev. 150427 (Update-3) as compared to the `App_Config\Include\Sitecore.ContentSearch.Solr.DefaultIndexConfiguration.config.example` file of Sitecore 8.0 rev. 150223 (Update-2).

1. In the **<fields>** section, add the following node, (you can add it at any place in the section, but in a clean Sitecore installation the new node is placed between the **calculateddimension** and **haschildren** fields):

```
        <field fieldName="culture"                              returnType="string"
>Sitecore.ContentSearch.ComputedFields.Culture,Sitecore.ContentSearch</field>
```

1.1.8 Changes to App_Config\Include\Sitecore.Diagnostics.config

The `Sitecore.Diagnostics.config` file was added in Sitecore 8.0 rev. 150427 (Update-3). You should download the clean config file using the link above.