Document number	Revision
UMA11497	1

Document subtypes and templates

1. Background

This article will show you how to create and maintain document subtypes and templates in Highstage.

2. User requirements for administering subtypes

2.1. General security

The user administering (creating/modifying) subtypes must have TrustRead, TrustModify and TrustCreate permissions for the basetype ts_subtype.

In order to administrate a specific subtype in Highstage, the user maintaining need proper access. The users should have [TrustRead] permissions for specific base type ts_subtype, so that they have access to see the subtype.

2.2. Template document Security

The template file may exist in the Templates folder defined by the Templates TemplatesPath parameter, or it may reside in a document.

To create new subtype using with a template in Templates folder the user has to have NTFS modify permissions to Templates folder on the server.

To create new subtype with a template in template document the user must be an author of the template document to be able to modify the template folder and schema XML file.

3. Create a new document subtype

Decide the id (short name) for the new subtype. It is vital that is short, intuitive and descriptive. As an example, "RS" may be used as id for type "Requirements Specification". It is essential that the id is long-term durable since it is challenging and cumbersome to change later, due to traceability and especially when the distribution of documents have taken place.

The ID must be unique, meaning that it can not be used if another base type or subtype already uses it.

Best is to use types that resemble entities or names already in place in the organization. If none of these is available, then inspiration can be found here:

- https://www.turbostage.net/ts_public/ts/view.aspx?t=doc&o=WP10014
- https://www.slac.stanford.edu/grp/md/stds/ds31001.html
- https://en.wikipedia.org/wiki/IEC_61355

To create the new subtype execute the following steps:

1. In the menu go under the desired basetype Eg. (Documents) and click [Create Type].
2. Enter the id (short name) as described in the previous section and click `Create`.

4. Descriptive information

The descriptive information is useful for users that need to create a new item. It may sometimes be difficult for users to decide which subtype to use, so descriptive information to help the user is beneficial.

1. Give the new subtype (ex. **RS**) and a descriptive Name (ex. **Requirements Specification**).
2. If possible/feasible an optional note for users about the purpose and use of the subtype.
3. The Class may be used to group documents types, in for example Quality, Production etc.

5. NextID

Use -1 for using global NextID counter defined by parameter NextID.

Alternatively set a value to make the type have its own NextID counter.

6. Extra configuration for ts_subtype

Extra columns can be added to the ts_subtype form. See general schema documentation for this.

Columns such as the class is by default free text as a string type. but predefined values may be specified by adding an xml snippet to custom.schema.xml, a sample is here:

```
  <type name="ts_subtype">
    <column name="class" values="Management;HR;Logistics;Production;R&amp;D;Sales;Quality" />
  </type>

```

7. Template folder

If left blank, then an empty folder will be created. However normally a folder template file(s) is required to provide user creating a document a suitable initial file or set of files.

Observe that Linux and Mac OS is case sensitive, file extension should be lower case to support these operating systems. For non-critical templates that do not require version control, the templates folder may be used. Alternatively, the template folder can reside in a document and the template folder is then under revision control.

7.1. Using Template folder

If no TemplateDocument have been chosen, the global template path defined by the parameter is used. The field TemplateFile can then be used select the specific file in the global template folder.

7.2. Using existing document folder as template folder

This option allows a template to be revision controlled and be used as a template for one or more types in Highstage. In the TemplateDocument field, enter the ID of the item which has the template.

Now use the TemplateFolder to locate the folder that holds the template.

Use the TemplateFile, to select the primary file that should be used as the primary template document.

If no primary file is chosen, all files in the template folder will copied over.

> **Important!**
> Make sure to not use Dashes (-) in the file name, because Highstage uses this one to show revisions.

8. Defining schema for a subtype

A custom definition for the subtype can be specified in a schema, which follows the general schema design principles. Choose the schema file in this field and reset the web app by pressing the restart button in the header, to take it into action.

9. Document files, refinery and properties

Refinery is the process that stamps document files with various information when the item switches status (Working, Freeze, Review, Approval, Approved).

Document refinery by default is able to process file formats from Microsoft Office applications. Examples are provided that show how this is done for office applications.

Refinery can be highly customized by code-behind so that new properties are added. Refinery is also able to generate various alternative rendition file formats. Please contact the Highstage support for further information about custom refine-job for a subtype.

10. Add document templates

There are too ways of implementing templates:

1. In specific directory. Only one file. No revision control of the template document.
2. Using the document handling system, and the templates would be under revision control.

10.1. Setup of Document templates with no revision control

In principle, all document types need a template document. As a template any type of file can be used, .ex Excel spreadsheet, a document, a txt file etc.

The root directory for templates files which are not part of the revision control, is found by looking in Highstage: **System**>**Management**>**Files**>**TemplatesPath**

In principle, does every **Document** type need a directory with the same name as the document type, and with a template file with the same name

10.2. Example

Document type: **DD**
Template directory **\DD\DD.doc**

It is possible to use other directory name, but still the filename and the directory name should be the same.
As an example:

Document type **DD**

Template directory **\STANDARD\STANDARD.doc**

11. Reviewers, Approvers and Distribution list

These fields give the possibility of having pre-filled users or user-groups for the specific document type. These settings can be found under the Options section on the subtype.

Set the **ReviewBy** to set to true, a default set of reviewers for the Subtype.

If the **ReviewByManager** field is set to true, the workforce manager will be added default to the lists of reviewers.

If the **ReviewEnforcement** field is set to true, Enforce that review is done, and that at least one role of a list of reviewers must review, and that the reviewers include a non-author.

Set the **ApproveBy** to set to true, a default set of reviewers for the Subtype.

If the **ApproveByManager** field is set to true, the workforce manager will be added default to the lists of reviewers.

If the **ApproveEnforcement** field is set to true, Enforce that review is done, and that at least one role of assigned reviewers must review, and that the reviewers include a non-author.

12. Additional subtype options

Additional options to configure subtype behavior can be specified. These settings can be found under the Options section on the subtype.

If the TemplateReference field is set, a reference to the template document that the document was created from is made by default.

A template for the review can be specified on the type by setting ReviewTemplate to object that contains the desired template files.

13. Security

Security to create a certain subtype can be controlled by going into the Subtype setting and change the [TrustCreate] field, this field can specify both users and VectorRoles.

14. Activation

To enable creating items of the subtype the Active attribute must be checked.

The subtype may later be inactivated to depreciate the subtype, preventing new items of this type to be created. Deactivating a subtype will not affect already created items of the subtype.

Highstage