IMPORTING EXISTING eCTDS

1 Contents

2 Revision History

When Ennov releases a new version of Ennov InSight, they issue Release Notes which explain the new features and updates. The Ennov Business Consulting Team reviews the Release Notes against each Best Practice to determine any impact to the document:

- Impact = Release notes-documented upgrade changes this Best Practice
- No Impact = Release notes-documented upgrade changes do not affect this Best Practice

When Release Notes impact Best Practice documentation, Ennov recommends that clients review the entire Release Notes for a full understanding of all changes associated with this Best Practice documentation.

Software Version	Release/ Revision Date	Summary of Change(s) (Refer to Release Notes for Full Description)
v7.3.4	08 May 2025	Update Best Practice for Ennov rebranding & for v7.3.4 – No Impact
v7.2	20-Jun-2023	Update Best Practice for v7.2 – No Impact
v7.1	13-Jan-2022	Update Best Practice for v7.1 – No Impact
v7.0	25-May-2021	Update Best Practice for v7.0 – Impact

3 Importing Existing eCTDs

eCTD applications created outside of Calyx RIM can be imported into Calyx RIM for Publishing so that future lifecycle sequences can be created from existing Applications. If the Application information already exists in Calyx RIM (Application and Sequence entities), importing additional Sequences can be done at the Application level. The import will match the eCTD Sequences with that information. If that information does not already exist, Application entity must be created for the import process to associate the Sequences with them.

The import process will alert you to any potential issues in the existing Sequences and provides options for how to handle lifecycle conflicts. The Import eCTD function is located on the Applicationpage. The options for import include:

- Import Leaf Elements Only: This option allows for importing only XML leafs found in the eCTD XML, without importing references to the content files. This prevents Calyx RIM from creating and assigning content document elements below each leaf during the import. It can also improve import speed significantly. It also captures all lifecycle operations correctly and prevents republishing of submissions already published.
- Import Orphan Leaf Elements: This option allows the import process to continue if it finds lifecycle conflicts in the eCTD, such as a Leaf being replaced in 2 different Sequences (submit Leaf A in 0000, replace Leaf A in 0001 with Leaf B, replace Leaf A in 0003 with Leaf C, which should have been a replace on Leaf B since A is no longer active). By default, the import process will stop when it encounters conflicts of this nature since the Working view cannot be created in this manner. However, if the Import Orphan Leaf Elements checkbox is selected, Calyx RIM ignores orphan leaf leaf instances and creates a new leaf elements instead. This allows you to create a corrective Sequence at a later time that removes the extraneous Leaf (in the example Leaf B) from the current Application.

The import process does create Assemblies in sequential sequence order, so if you submitted Sequences out of order, you may need to perform an iterative import to ensure the intended lifecycle is preserved. To perform an iterative import, you must move the later Sequences out of the main Application folder to a temporary location so they are not part of the import. After the initial import is complete, you can move the other Sequences back into the main folder and perform a second import. The Sequences with existing Assemblies will be ignored.

Warning - To import files successfully, they must follow the naming convention **filename.extension**. This is not an issue on file systems but is critical in a DMS where the file name may not have the extension appended, but rather, identified as a format type. As an example, a file named index with a format type of **xml** will not import, but **index.xml** will import. The same is true for content files. For example, **cover-letter.pdf** will import but **cover-letter** with a format type of **pdf** will not.