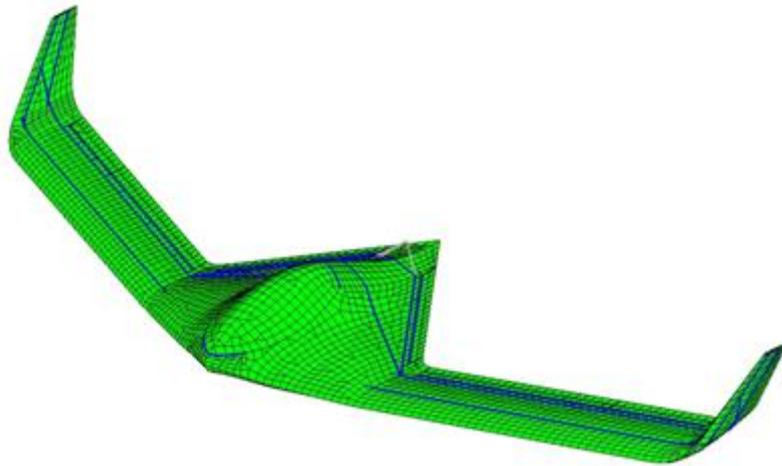




## **Simulation Data Management**



***EDMopenSimDM™***

**Usage Guide (R16)**



## TABLE OF CONTENT

1	Introduction to this document .....	3
2	Introduction to EDMopenSimDM .....	3
3	Use cases .....	4
3.1	Management of data on an active program .....	4
3.1.1	Access control .....	6
3.1.2	Define Product Structures .....	6
3.1.3	Manage files and models .....	7
3.1.4	Baselines .....	8
3.1.5	Folder versions .....	8
3.1.6	Links among files, folders and models .....	9
3.2	Use in customer support organizations .....	9
3.2.1	Browse in EDMopenSimDM client .....	10
3.2.2	Browse in CAE tool .....	11
3.3	Long term archival of program data .....	11
3.3.1	Archival .....	12
3.3.2	Retrieval .....	12
3.4	Overview of supporting EDMopenSimDM functionality .....	12
4	The EDMopenSimDM database .....	14
4.1	Federated AP209 product data model .....	14
4.1.1	Importing a Design Model .....	14
4.1.2	Importing an Analysis Model .....	15
4.1.3	Creating or Importing Structural Test Models .....	16
5	References .....	18

## TABLE OF FIGURES

Figure 1:	Process diagram - Management of data on an active program .....	5
Figure 2:	Process diagram - Details of structural testing .....	6
Figure 3:	Process diagram - Use in customer support organizations .....	10
Figure 4:	Process diagram - Long Term Archival and Retrieval of Simulation Data .....	11

# 1 Introduction to this document

This EDMOpenSimDM Usage Guide describes the typical use and the benefits of the EDMOpenSimDM application. There is also an EDMOpenSimDM Reference Manual, which lists and explains the individual user functions of the application. It is recommended to study this usage guide before reading the reference manual.

EDMOpenSimDM is currently available in release 16, respectively CRYSTAL Platform, version 3. This document relates to the state of the current implementation.

## 2 Introduction to EDMOpenSimDM

It is the objective of EDMOpenSimDM to improve the effectiveness of simulation, analysis and testing programs and the sharing of information among them. Thus, EDMOpenSimDM is an addition to existing engineering analysis tools; it offers special provisions to reuse data across traditional engineering application boundaries.

EDMOpenSimDM is designed for the following use cases, which are described in more detail in section 3 below:

- Management of design and engineering analysis data on an active program
- Use in customer support organizations
- Long term archival of program data.

EDMOpenSimDM is different to other simulation data management tools on the market in offering an interoperable solution by supporting the internationally recognized ISO STEP standard format - specifically ISO 10303-209:2014 [1], which is edition 2 of AP209. The following functionality of EDMOpenSimDM supports data interoperability:

- import of AP209 files in ISO 10303-21 (P21) format;
- merging of subsequent imports of product structure, design, analysis and structural test data into one consistent AP209 product model, also referred to as federated model;
- validation of AP209 data;
- visualization of the graphical aspects of AP209 data;
- engineering queries into AP209 data, including queries into composites data and queries to prepare correlation plots for analysis and structural test data;
- AP209 compliant representation of interactively defined product structure and management data;
- export of interactively defined data into AP209 files in P21 format;
- export of zip-archives of a complete product breakdown structure or of branches of it including the federated product model.

This openness enables the support of the following sub-goals and benefits:

- Design, simulation and test data are made accessible for the support and upgrade of aircraft, land vehicles, and ships:
  - Allows users such as national Ministries of Defence and Air Logistics Centers to have access to original manufacturer information.

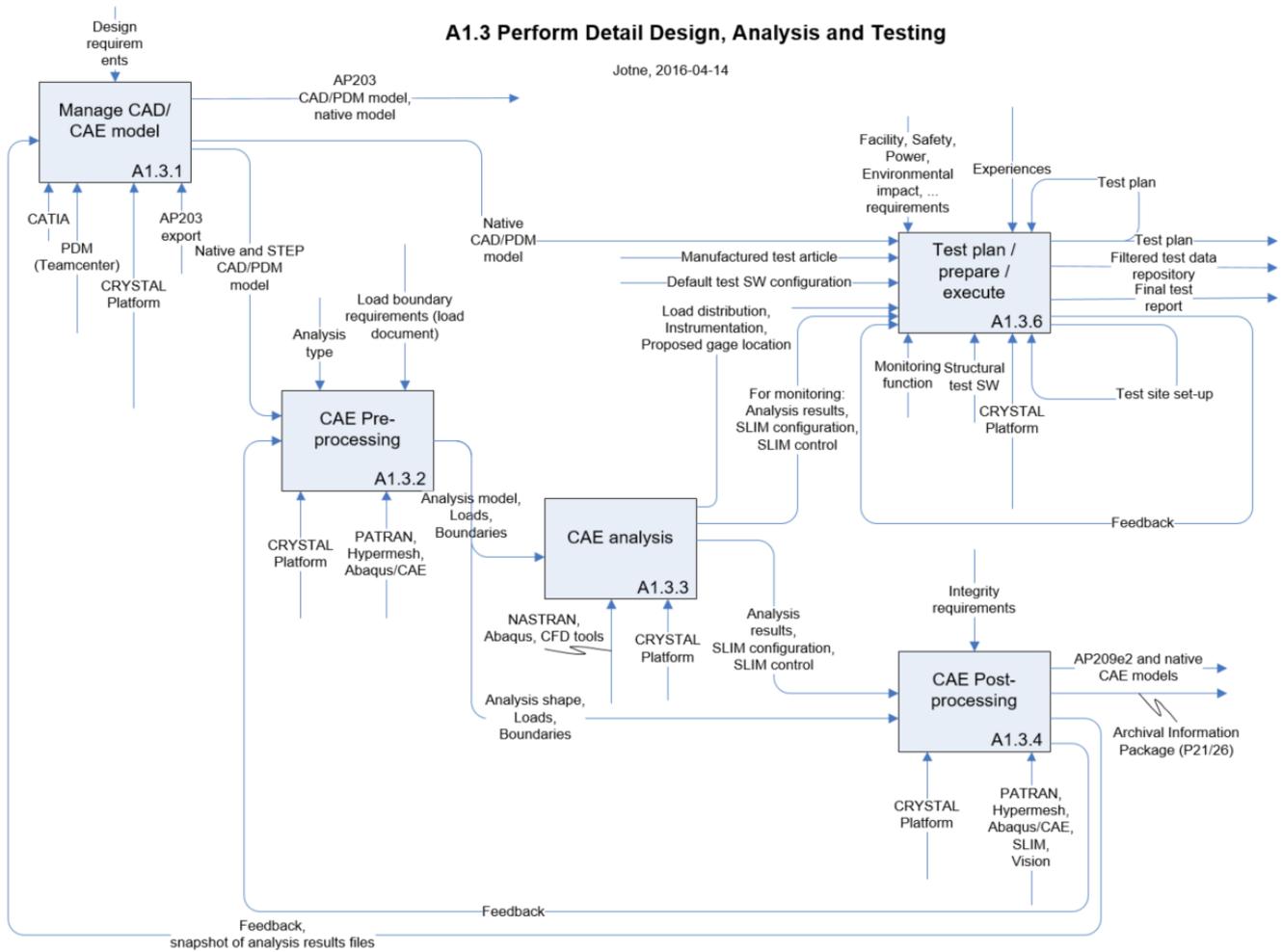
- EDMOpenSimDM interfaces are available for use with off-the-shelf CAD and CAE engineering applications:
  - Provides the mechanisms for single access to product structure, CAD models, CAE simulation data and CAT test data for engineering programs.
- Long term data retention issues are resolved:
  - Enables cost-effective life-cycle processes.

## 3 Use cases

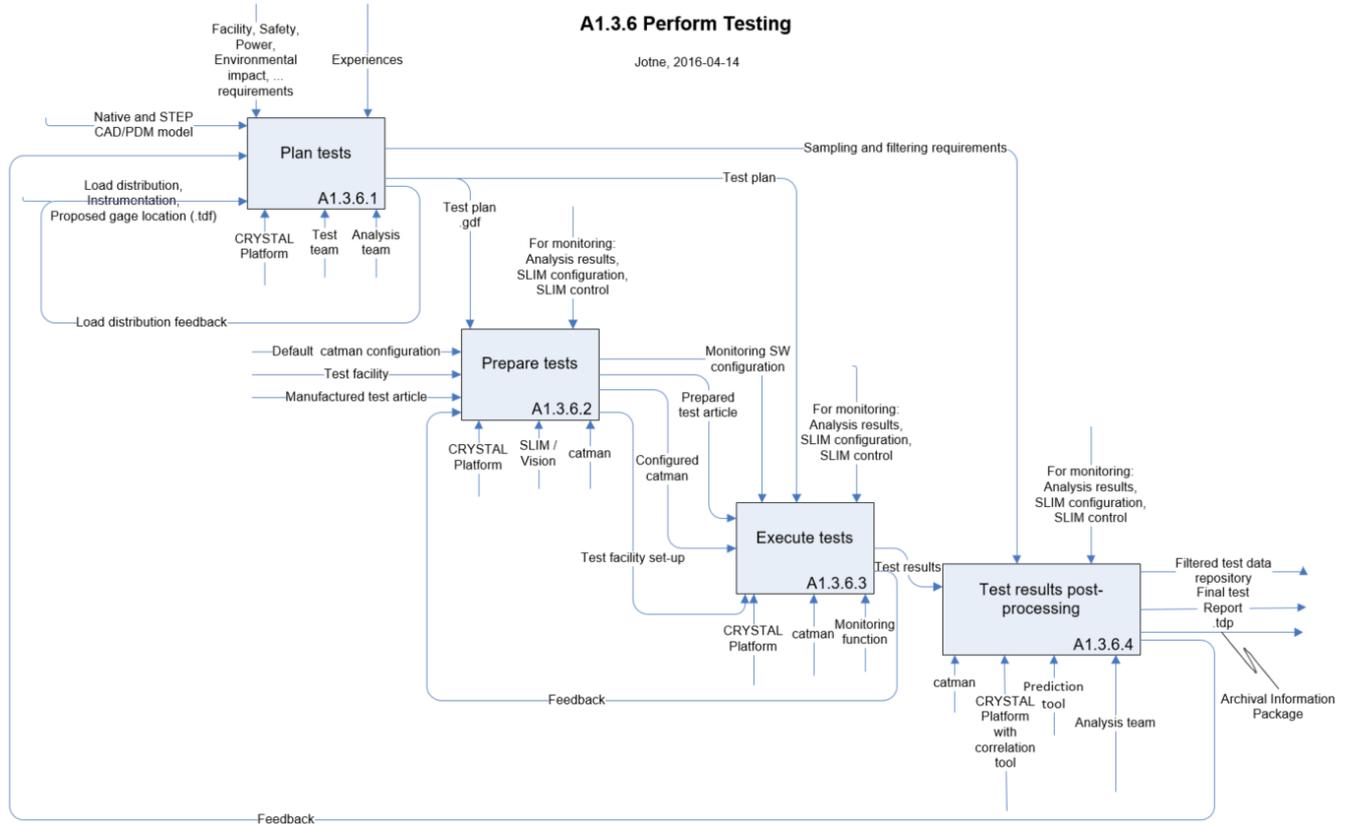
### 3.1 *Management of data on an active program*

Management of data is a process that continues as long as a program is active, as long as new Computer Aided Design (CAD) models or new versions of existing CAD models are created and as long as Computer Aided Engineering (CAE) models are created and analyses are executed and validated against physical tests of the product. The task is to take care of all analysis data in a database and relate them to CAD models and Product Structures (PS). Therefore, the resulting database shall be a database where it is easy to find and display all the different types of design, simulation and test data.

Figure 1 and Figure 2 show diagrams of a product development process with focus on design, analysis and structural testing; Figure 2 is a detailed view of the box named A1.3.6 in Figure 1. The EDMOpenSimDM application supports data management for all these steps.



**Figure 1: Process diagram - Management of data on an active program**



**Figure 2: Process diagram - Details of structural testing**

### 3.1.1 Access control

EDMopenSimDM users are assigned types of access to the system. Users may play two types of roles:

- UserManager<sup>1</sup> – can create/update/delete users.
- SystemManager – can create new product structures and has the rights of a SystemUser.
- SystemUser – can play roles in product structures as Admin, Editor or Reader.

Within a product structure the following roles may additionally be assigned:

- Admin – user has administrator (manager) access rights. User can create new persons, assign access rights for them and approve actions.
- Editor – user is an editor. User can create new folders, upload files and make some updates.
- Reader – user can not update the product structure, but can add comments to objects.

### 3.1.2 Define Product Structures

The person responsible for data management will in A1.3.1 (Figure 1) either import a CAD Product Structure (PS), or create a PS suitable for managing the data files in the simulation process, providing a structure to organize the associated CAD, CAE and test files. These Product Structures may be quite cursory and just enough to organize the simulation models.

<sup>1</sup> Only “superuser” is defined with UserManager role.

The following is a typical sequence of commands to establish and populate a product structure:

- Login as manager
- Create repository
- Create product structure (PS) root folder (new model) in this repository
- Create meta data for the PS (persons, approvals, organization, ...)
- Add properties for the model
- Define methods
- Browse through the meta data (property view)
- Add children to the new product structure
  - Create folders manually, for example, for documents or other project information
  - Import externally defined domain models with product structure, design, analysis and structural testing data from AP209 files to build a product model
- Apply copy, paste and remove functions to folders
- Change default access of administrators, editors and readers to folders
- Add task descriptions to folders
- Assign property values to folders

### 3.1.3 Manage files and models

In process steps A1.3.1 through A1.3.6 (Figure 1) the user will import externally produced data sets for CAD, CAE and also structural testing into the EDMopenSimDM database. The original files and their contents will be managed separately. The files will be treated as independent units, whereas their data sets will be integrated as domain or sub-models into a comprehensive and consistent product model; for details, see the description of the federated model in 4.1 . As part of the import process the CAD, CAE and structural testing sub-models will be linked to items in the Product Structure (PS).

The user may validate imported AP242 or AP209 data according to the AP-schemas. The validation process will output a list of non-conforming data instances used to debug and cleanse the models as necessary. This output will be linked to the original file in the database, not to the sub-model.

The following is a typical sequence of commands to manage files and models:

- Login as editor
- File specific functionality:
  - Create document folders
  - Add document extensions that are missing from your EDMopenSimDM installation
  - Upload files (pdf, bdf, op2, AP203, AP209, AP242, ...) to folders of type Document
  - Establish links to external files (both .stp and non .stp files)
  - Validate STEP-files
  - Run queries against AP209 models

- Remove files
- Browse through the automatically created meta data
- Add meta data to files
- Product model specific functionality:
  - Create a product model parent folder (optional; this may also be done while importing product structure using the import wizard)
  - Import AP209 and other STEP files as sub-models into the product model using the import wizard
- Browse through uploaded files and models, among others, using Vcollab (for NASTRAN and AP209 files)
- Add replies to task descriptions

### 3.1.4 Baselines

In Product Lifecycle Management (PLM) applications product structures have baselines and versions. EDMopenSimDM offers functionality for both; folders may be versioned, and unique versions of the constituents of a product structure may be collected in a baseline.

The following is a typical sequence of commands to establish a baseline:

- Login and select a model that you are administrator for
- Apply final approval to some of the folders
- Create baseline 1 with folders that all are finally approved
- Create baseline 2 and include some folders that are not approved
- Review the resulting baselines.

### 3.1.5 Folder versions

Further editing of a frozen Product Structure is the same as building the next revision of the Product Structure. The following is a typical sequence of functions to deal with folder versions:

- Login and select a model that you are administrator for
- Create folder versions
- Create folder versions of folders belonging to a baseline
- Copy a folder version to another parent
- Review new folder versions
- Assign editors and readers to folder versions
- Link versions of different folders to one another
- Remove a folder version
- Login and select a model that you are editor for

- Change folders of an underlying branch that effect corresponding folders of the previous version of the high-level folder.

### 3.1.6 Links among files, folders and models

Links establish relationships among folders and among models for the sake of traceability. Links may also be created for files to maintain the single source principle. Links among Product Structure folders and for files are established manually; links among product model constituents are created automatically as result of the STEP import wizard. For example, the following types of folders may be linked to a PS folder:

- CAD-model
- Simplified CAD shape for analysis (idealized shape)
- Mesh model
- Loads
- Analysis results
- Product under test
- Test case.

In addition, files may be linked to the different folders from their source locations.

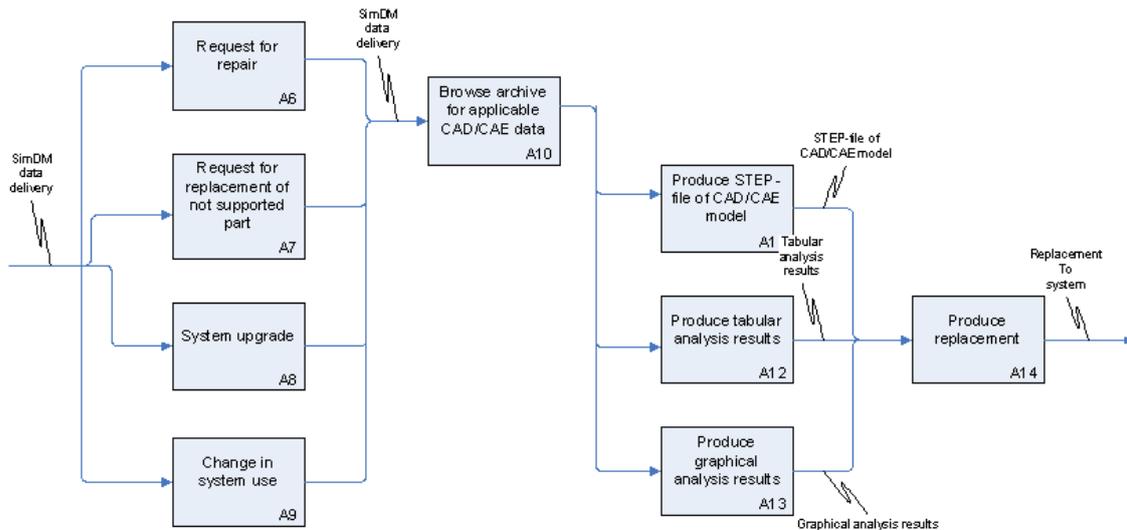
Try the following functions:

- Login and select model that you are editor for
- Link an analysis folder to a design folder
- Click on both the created link objects
- Delete a link
- Upload a file to the design folder
- Drag and drop a link to that file into the analysis folder

## 3.2 *Use in customer support organizations*

With receiving digital engineering analysis models, intended changes to the system-in-support may be analyzed locally, that is, without or at least with less involvement of the manufacturer.

EDMopenSimDM makes this easier and cheaper by delivering system data in a standard format. This enables the end user to select among several engineering analysis tools for local treatment of the data. In addition, the EDMopenSimDM end user receives an application to browse the engineering analysis system data. Those who do not need to manipulate the data will, thus, not need to purchase any additional application to display the data.



**Figure 3: Process diagram - Use in customer support organizations**

### 3.2.1 Browse in EDMopenSimDM client

In this use case, the software is only used for reading data, not for manipulating them, as in the use case for management of active program data. The user can browse, search and extract data.

The following is a typical sequence of functions to deal with browsing of data:

- Login and select model that you have read-access to
- Select in repositories and models panel the function "Import from archive"
- Browse through the imported Product Structure
- Browse through meta data of the product structure, the product model and the files
- Browse through the product model, textually and graphically (VCollab)
- Open and browse through various types of files
- Download a file
- Select a folder, create and then download a data package
- Query the contents of the product model or of AP209 files
- Search in product structure meta data

Organizations who want to restrict access to the EDMopenSimDM server data or need clients without installation may use the web-client. It offers the following functionality, which is not yet identical with the typical sequence of functions in the list above:

- Login: Login to Web Client
- Browse model: Browse a model
- Download file: Download a file
- Export branch: Export a structure
- Meta-data: Explore the meta-data of a folder
- Add folder: Create a folder in the product structure
- Add folder version: Create a version of an existing folder
- Set folder version as preferred: To establish a default view of the folder

- Add file: Upload a file
- Add remark: Add a remark to a folder
- Add approval: Add an approval to a folder
- Import folder: Upload a zipped folder as folder or branch to an existing folder
- Delete folder: Remove a folder permanently.

### 3.2.2 Browse in CAE tool

The following is a typical sequence of functions to deal with browsing in a viewing tool:

- Login and select model that you are reader for
- Select a suitable CAE-file and open it in Vcollab Presenter
- Browse and manipulate the opened model
- Add another CAD/CAE model to the same display
- Try out the callback functionality by selecting (double-click) the two models after each other and observe the product structure panel.

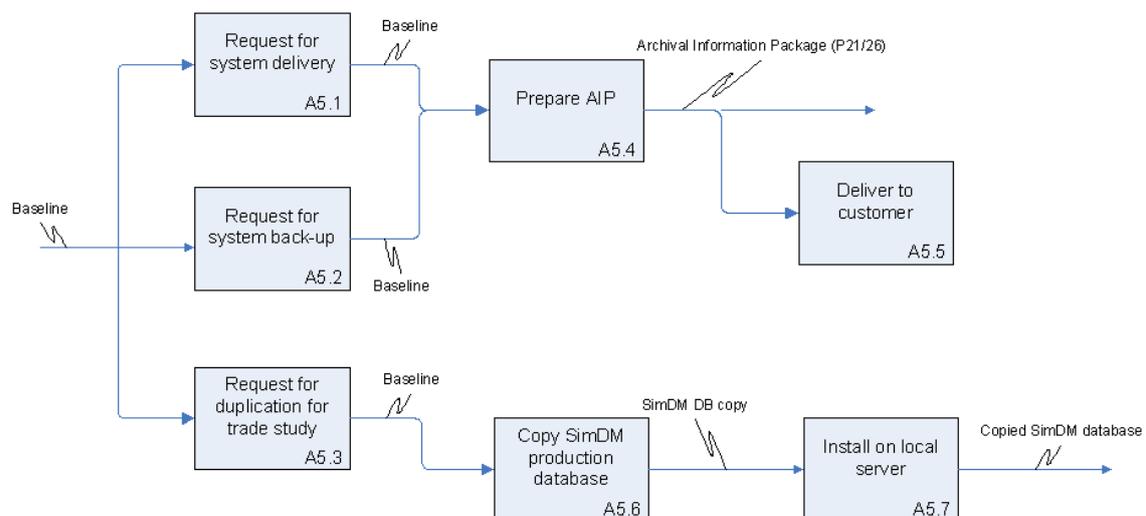
### 3.3 Long term archival of program data

The process diagram in Figure 4 shows the main activities involved in the archival use case. The three activities on the left trigger an archival process:

1. A system is ready for delivery to a customer. The engineering analysis data related to this system shall accompany the delivery and shall also be archived at the source organization, that is, usually the production facility.
2. A back-up of the production system is required, as part of a back-up routine or extraordinarily.
3. An on-going study shall be branched, that is, several options shall be investigated. A current branch shall be copied and re-used.

In all cases, the activity can be documented by a copy of the archived data set itself.

The two first of the triggers, A5.1 and A5.2, require a full archival process to be executed; the third one only needs a copy of the active data set.



**Figure 4: Process diagram - Long Term Archival and Retrieval of Simulation Data**

### 3.3.1 Archival

An archival task is started at any time upon command from a user.

Especially for delivery to customers the archived items will need to fulfill certain quality standards and approval procedures. A result report will be included in the archival package.

Any node in EDMOpenSimDM can be the root node for an archival package. CAD/CAE models can be archived individually.

To create the delivery package the user executes a function to select the correct configuration of the product. This function will check that all necessary information about the selected product version is available in the database. For example, archival items will not be included in the package unless they are approved. If that is not the case, the function is interrupted and the user must provide the missing approval.

Each archival operation results in a complete archival package. Its successful creation is logged by an entry in the Detail-view of EDMOpenSimDM. There is no functionality to relate new archival packages to already existing ones. Outdated packages may be removed.

The following is a typical sequence of functions to deal with archiving:

- Login and select model that you are editor for
- Select a folder and create an archive package; include validated and not validated STEP-files
- Select a baseline and create an archive package; include only validated STEP-files
- Select entire PS and create an archive package
- Unpack the created zip-file into a local folder in the operating system.

### 3.3.2 Retrieval

The following is a typical sequence of functions to deal with retrieval:

- Login and select model that you are editor for
- Select a repository and import an archived package into a new model / product structure
- Alternatively, select a folder in the product structure of your choice and add the retrieved package from the new model as new branch.

## 3.4 Overview of supporting EDMOpenSimDM functionality

The following functionality is available from EDMOpenSimDM to support the different use cases:

- i. For managing design, engineering analysis and test data:
  - Work collaboratively in a multi-user environment
  - Organize your program data in repositories and models
  - Create product structures, automatically from STEP-files or manually
  - Use product structures to manage your files
  - Add to items in the product structure externally defined domain models from design, analysis and structural testing into one consistent product model
  - Link engineering domain models for traceability
  - Assign properties to product structure items

- Upload and download files of any type, including AP209
  - Import AP209 files to build a lifecycle product model as a federated AP209 instance model
  - Get meta data, like timestamp and creator, automatically assigned
  - Manage distributed and local files in the database
  - Assign resources and methods to tasks
  - Provide task feedback
  - Manage peer approval and final approval
  - Version items in product structures
  - Create baselines for all or branches of a structure
  - Validate data provided in STEP-format
  - Search in meta data
- ii. For viewing of design, engineering analysis and test data:
- Browse product structures like folders
  - Execute engineering queries into AP209 data
  - Run survey queries to find critical spots in analyses
  - View files with your favorite tools
  - Browse analysis data, native and AP209, graphically with built-in 3D Viewer
  - Create plots of correlations between analysis results and test measurements.
- iii. For archiving:
- Export an entire product structure or branches of it as “Data package”
  - Export baselines
  - Archive for the long term according to LOTAR standards (EN 9300-11)
  - Import product structures and other engineering domain models from STEP files
  - Import archived projects/programs and baselines
  - Apply validation and verification during ingest and retrieval processes
- iv. For access control:
- Login access control
  - Separation of responsibilities into manager, editor and reader
  - Locking of product structures
  - Open Standard API and Web services
  - Customized access control for
    - product structures and their elements
    - low level assignment of roles

## 4 The EDMOpenSimDM database

This chapter explains how the EDMOpenSimDM database creates, grows and maintains an AP209 product model.

### 4.1 Federated AP209 product data model

A federated AP209 product data model consists of several data sets, also called domain models or sub-models that together make up a complete and consistently interconnected model. It may consist of the following types of models:

- Nominal Shape Model (Design Model)
- Idealized Shape Model (Design Model)
- Analysis Model with sub-models, Mesh, Load Cases and Analysis Results
- Structural Testing Model.

*EDMOpenSimDM* has functions to establish links between instances in the different models to ensure the consistency of the federated model.

Between a Design Model and a corresponding Analysis Model links are established when importing the Analysis Model into a federated model that already contains a Design Model. Likewise links are created when importing Structural Test Models. Whenever applicable, the AP209 recommended practices [2] are followed. Particularly, for linking concepts of design and analysis, Figure 19 applies, “Analysis vs. Design Discipline Product Definition”.

In the following the different possible links created upon import are presented.

#### 4.1.1 Importing a Design Model

The user may import a model containing the Design of a product to EDMOpenSimDM. Different cases for this import exist:

- a) The imported model is the first Design of the represented product imported to the federated product model:
  - a. No additional instances are created in the Federated Model beyond those in the imported file.
- b) The imported model is a different representation of an already existing Design in the federated model:
  - a. An instance of `shape_representation_relationship` is created to indicate that the new shape is an alternative to the existing one.
- c) The imported model is a different version of an already existing Design in the federated model:
  - a. An instance of `product_definition_formation` is created to establish a new version of the product structure item that the shape is supposed to represent; thus, not the shape is versioned, but the parent of the shape.

The attributes of these new instances are set to the following values:

**Entity: `shape_representation_relationship`**

name	'alternative representation'
description	'alternative representation'
rep_1	shape representation of the existing Design Product
rep_2	shape representation of the imported Design Product

### Entity: product\_definition\_formation

id	<version id assigned by user>
description	<version description assigned by user>
of_product	The Product of the existing Design Model

The imported Design Model may contain the idealized shape of a design. At the time of import, EDMopenSimDM does not recognize a shape as an idealized shape unless it is related to an Analysis in the imported file already. Upon import of a shape, the role of the shape is not taken into account; the shape is stored as the initial shape of a part or assembly or as an alternative shape representation. Shapes will first be categorized as idealized ones by the user during analysis imported (see section 4.1.2).

#### 4.1.2 Importing an Analysis Model

The user may import a model containing the analysis of a design to the federated model. We assume here that a Design Model is already present in the database, but this is not mandatory.

During import, the user may choose to include an idealized shape with the analysis model. This may be one or multiple models already existing in the database as Design Models or imported as part of the analysis STEP file.

The imported model may contain:

- a) An analysis product representing the analysis of a single product (part).
- b) An analysis product representing the analysis of an assembly of products (parts).

The only impact of this difference is in the assignment to the correct level in the product structure tree. For both cases and independent of whether an idealized model is available or not, a single entity is created:

```
product_definition_formation_relationship
```

between the product\_definition\_formation of the design of the single part or of the assembly (relating) and the product\_definition\_formation of the analysis (related).

If the Design Model includes idealized shape(s) for the analysis, the user associates the idealized shape(s) with their corresponding nominal shape(s), which are also parts of the Design Model of the same single part or assembly. We assume a many-to-many relationship between the two types of shapes. Two types of relations are created using the following entity data types:

```
shape_representation_relationship and
shape_definition_representation.
```

For each combination of idealized and nominal shape an instance of `shape_representation_relationship` is created with `rep_1` being the idealized `shape_representation` and `rep_2` the nominal `shape_representation`.

For each idealized shape an instance of `shape_definition_representation` is created to link the idealized shape to the `product_definition_shape` of the analysis product; this `product_definition_shape` shall point to the `product_definition_formation` instance that is the related one in the above `product_definition_formation_relationship`. Thus, after this operation each idealized shape will be pointed to by two instances of `shape_definition_representation`, one connecting it to the design product and another one connecting it to the analysis product.

The attributes of the involved entity data types are set to the following:

**Entity: `product_definition_formation_relationship`**

id	<GUID set by the system>
name	'analysis design version relationship'
description	<no value>
relating_pdf	product definition formation of Design Product
related_pdf	product definition formation of Analysis Product

**Entity: `shape_definition_representation`**

definition	product definition shape of Analysis Product
used_representation	shape representation of Idealized Shape

**Entity: `shape_representation_relationship`**

name	'basis'
description	<no value>
rep_1	shape representation of Nominal Shape
rep_2	shape representation of Idealized Shape

### 4.1.3 Creating or Importing Structural Test Models

Also Structural Test Models are created from import of STEP-models into *EDMopenSimDM*.

Instructions for how to instantiate AP209 data to represent structural test data are not included in the AP209 recommended practices [2]. Instead the results of the scientific paper “Relating Structural Test and FEA Data with STEP AP209” [3] have been applied.

The first part of the process is to import gage definition data, which will initialize the Structural Test Model by establishing a product under test. Within this new domain model the main `product` represents the physical part that is being tested. In the second part of the process the user specifies the test definition, which establishes a specific test case for the product under test. The last part of the process is to import the test results data.

#### 4.1.3.1 Importing Gage Definitions

Upon import of the gage definitions instances of `product_definition_formation_relationship` are created to relate the representation of the tested part to the Analysis models.

Their attributes are set to the following values:

**Entity: `product_definition_formation_relationship`**

id	<GUID set by the system>
name	'physical analysis version relationship'
description	<no value>
relating_pdf	product_definition_formation of Analysis Product
related_pdf	product_definition_formation of Tested Part Product

**Entity: `product_definition_formation_relationship`**

id	<GUID set by the system>
name	'physical design version relationship'
description	<no value>
relating_pdf	product_definition_formation of Design Product
related_pdf	product_definition_formation of Tested Part Product

#### 4.1.3.2 Importing Structural Test Definitions

Upon integrating the Test Definitions in the previous stage, the Structural Test Model is complemented with the representations of the structural tests. Instances of type `action_view_relationship` are created to relate the test cases to analysis load cases, one instance for each test case.

The attributes are set to the following values:

**Entity: `action_view_relationship`**

id	<structural test case ID> + <load case ID> + 'relation'
name	'structural test analysis load case'
description	<no value>
discretised_model	control analysis step of Analysis Product
idealised_action	action representing the physical test

#### 4.1.3.3 Importing Structural Test Results

Upon importing the Structural Test Results, the Structural Model is further complemented with the results data. The main entity of the results data is a product. The link to the Analysis Model is done with entity `product_definition_formation_relationship` with the following attribute values:

id	<GUID set by the system>
name	'test result analysis relationship'
description	<no value>
relating_pdf	product definition formation of Analysis Product
related_pdf	product definition formation of Test Result Product

## 5 References

- [1] ISO 10303-209:2014, Industrial automation systems and integration — Product data representation and exchange — Part 209: Application protocol: Multidisciplinary analysis and design
- [2] Recommended Practices for AP 209 ed2, 10303-209:2014, March 30, 2016, Version 2.0
- [3] Relating Structural Test and FEA Data with STEP AP209, R.Lanza, J.Haenisch, K.Bengtsson, T.Rølvåg, 2018-03-05, under publication by “Advances in Engineering Software”, ISSN 0965-9978