# Design and Reconfiguration User Manual

## 1. Concepts

The Design and Reconfiguration Toolkit provides a modelling environment for MODAPTO modules and production schedules compliant respectively with Asset Administration Shell and Business Process Modelling Notation standards. From the tools, it is possible to interact with the MODAPTO system in order to publish the defined MODAPTO Module in the Digital Twin Manager (responsible for the creation of the AAS compliant digital twin instance of the MODAPTO Module), assign to it smart services defined in the MODAPTO Service Catalogue, and publish the production schema in the MODAPTO Knowledge Base. Additionally, it is supported the modelling of production schema in Petri Net format, needed for specific cases of Co-Simulation.

## 2. Deployment

The Design and Reconfiguration Toolkit is provided as a Windows x86-64 standalone and portable application, that can be downloaded from the following GitHub page:

`https://github.com/Modapto/design-and-reconfiguration-tool/releases`

Once downloaded and extracted the Zip file, the user can lunch the toolkit by running the file "`modelling-toolkit.bat`".

In case the user want to build the Design and Reconfiguration tool from the ADOxx library source code, the instructions are available in https://github.com/Modapto/design-and-reconfiguration-tool/tree/main/aas-adoxx-classic-library .

## 3. Usage

To create a new model, independently if for MODAPTO Module or Production Schema definition, access the menu "Model/New…" and a dialog will show the list of available model types (Asset Administration Shell, Data Binding Model, Business Process Diagram, and Petri Net). After the selection of the appropriate model type and the specification of a name, version, and path, the new model will be created (Figure 1).

The toolkit will show a canvas, where the selected model can be described, by drag and dropping the model elements and their relations, from the Palette on the left side of the canvas.

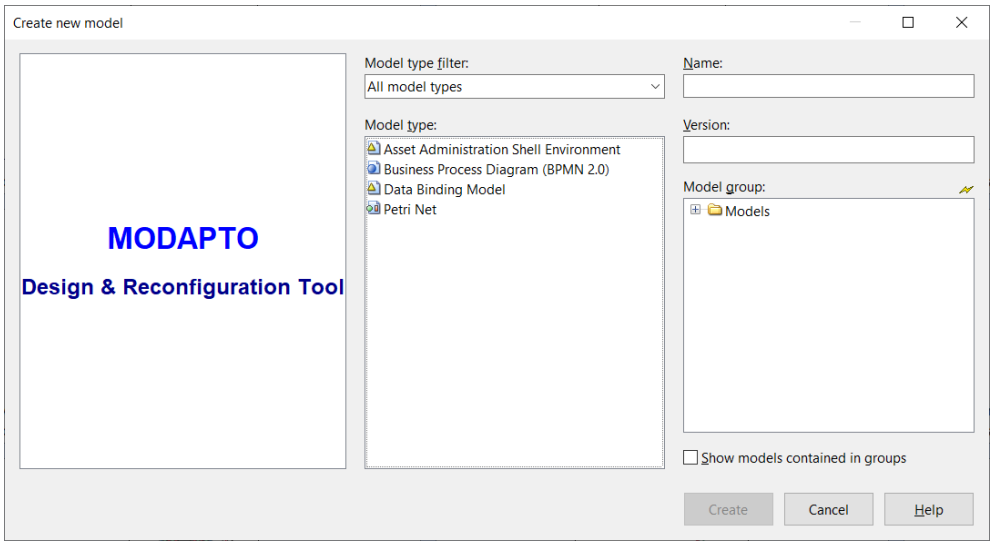


*Figure 1: New Model*

In the sample Asset Administration Shell model of Figure 2 can be seen the modelling canvas filled with different objects from the palette and arrows that relate such objects. Each object and relation in the palette has a type identifier (visible moving the mouse over the palette object) and a set of properties specific for the object type.

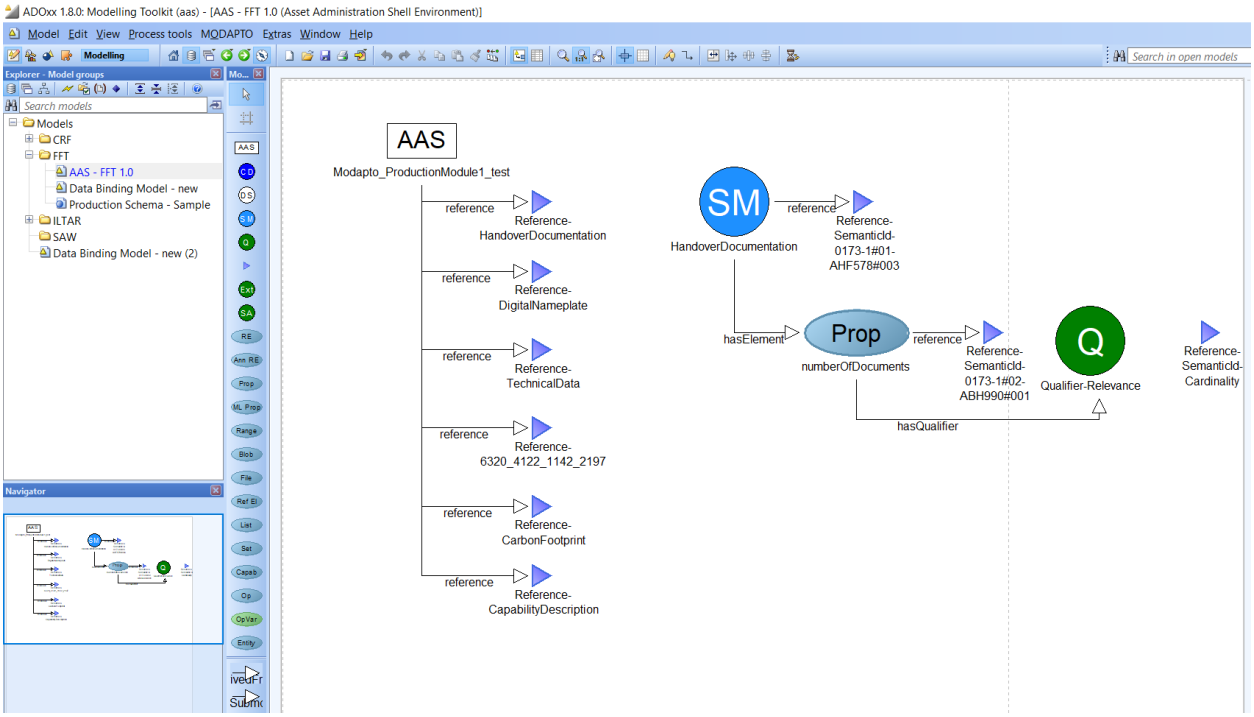


*Figure 2: Asset Administration Shell Model Sample*

To visualize all the properties of an object in the canvas, just double click on the object and a notebook like dialog will appear, with all the object properties grouped in different tabbed sections. The properties can be of different types like short or long strings, number, references to other objects and lists of values.

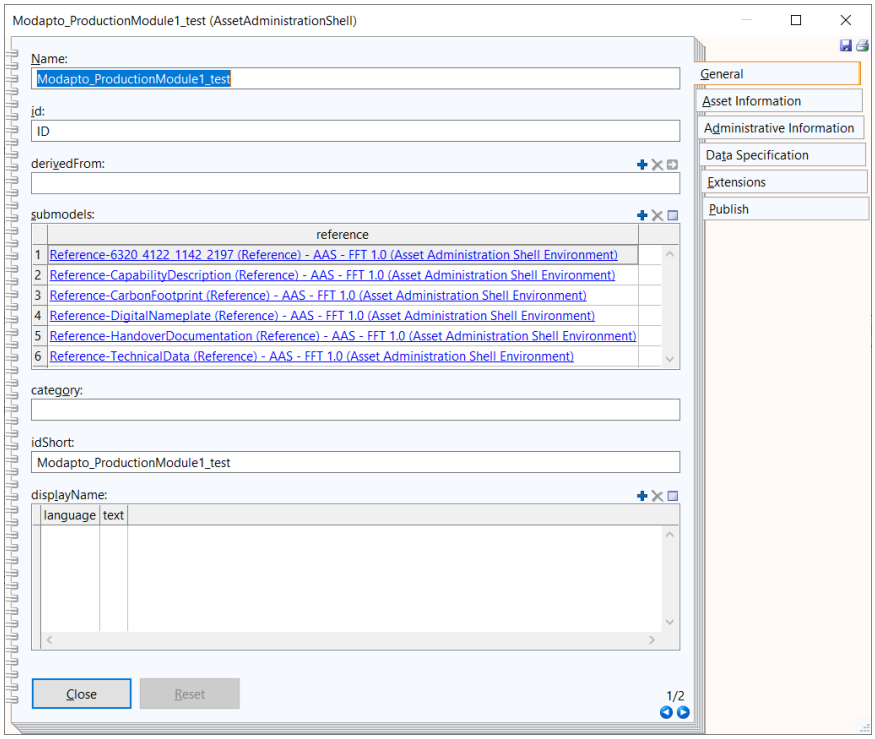


*Figure 3: Object Properties Sample*

In the following the details of the different model types will be provided:

- **Asset Administration Shell (Environment)**: This model type reflects the ASS standard metamodel specification v3.0.7[1] available at https://industrialdigitaltwin.org/en/content-hub/downloads with name "Specification of the Asset Administration Shell Part 1: Metamodel IDTA Number: 01001-3-0". It enables the definition of the main AAS Environment, composed of AssetAdministrationShell objects, ConceptDescription objects, and the different Submodels objects, with their Qualifiers and all the possible SubmodelElements. For each of them all the attributes defined in the metamodel specification are available in the object notebook. The user is suggested to read the metamodel specification to understand all the concepts introduced in this model type.
- **Data Binding Model**: This model type enables the definition of relations between existing Asset Administration Shell objects from models of type Asset Administration Shell (Environment) with services from the MODAPTO Service Catalogue. The "AAS Reference" object contain only one attribute that is the reference, while the objects "External Service", "Internal Service", and "FMU" contains the "catalogueId" attribute as unique identifier of the service in the Service Catalog, a description of the service, and a "publishId" attribute that will contains the id returned from the Digital Twin Manager after the publishing of the assignment (explained later). Selecting the

[1] https://industrialdigitaltwin.org/en/wp-content/uploads/sites/2/2023/06/IDTA-01001-3-0_SpecificationAssetAdministrationShell_Part1_Metamodel.pdf

"Binding" relation arrow from the palette and connecting an "AAS Reference" with different services objects will conceptually represent the assignment of the services to the referenced AAS.

- **Business Process Diagram (BPMN 2.0):** This model type enables the definition of a MODAPTO Production Schema in standard BPMN 2.0 format. The production schema in this way is represented as a process flows, with tasks that can be optional or be performed in parallel, depending on the depicted flow. For an introduction to the BPMN concepts the user is suggested to read the standard modamodel specification at https://www.omg.org/spec/BPMN/2.0/PDF. Additionally, every BPMN Task object includes a new AAS specific section that enable to reference an existing AAS Submodel element, related to the execution of the specific task (Figure 4).

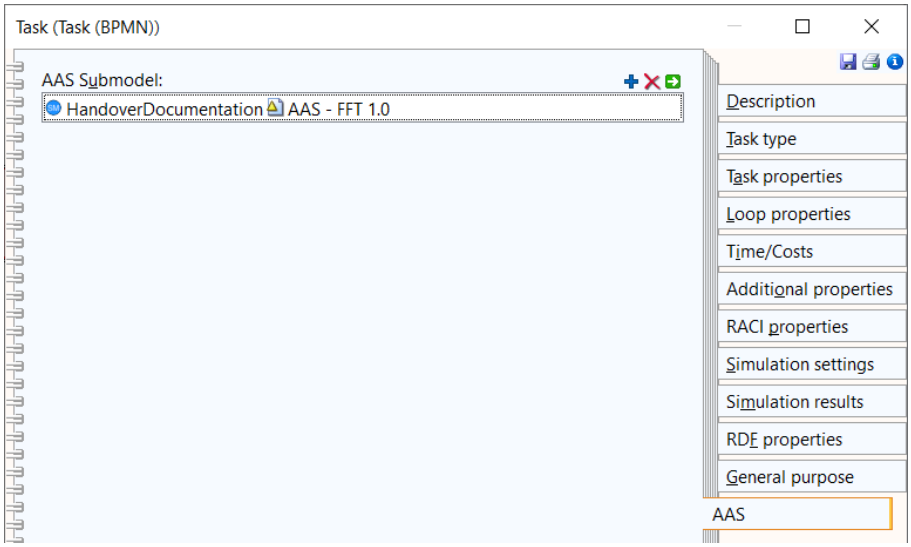


*Figure 4: AAS reference in BPMN Tasks*

- **Petri Net:** This model type enables the definition of a MODAPTO Production Schema with Petri Net logic. Petri Nets is one of the longest standing diagrammatic modelling methods, with minimal but powerful semantics based on strong mathematical foundations. The core concepts are Places (states), Transitions (changes, actions) and Arcs (indicating the flow) as well as Tokens (marks) which capture the behavioural dynamics of a system. The firing (execution) of Transitions signify an action taken by passing the Tokens between their adjacent Places according to the defined flow and may take place only if the connected Place contains at least the number of Tokens specified in the Arc. The user is suggested to read one of the many tutorial on Petri Net available on the internet like the one available at http://files.untiredwithloving.org/petrinetTUTORIAL.pdf.

MODAPTO specific functionalities, both generic and specific for the different model types are available under the MODAPTO menu (Figure 5). From here is possible to configure the connection to the MODAPTO system, login and logout to the MODAPTO authentication, publish the Production Schema, publish the Asset Administration Shell defining the MODAPTO Module, publish the assignment of services to the published MODAPTO Module and import the services from the MODAPTO Service Catalogue.

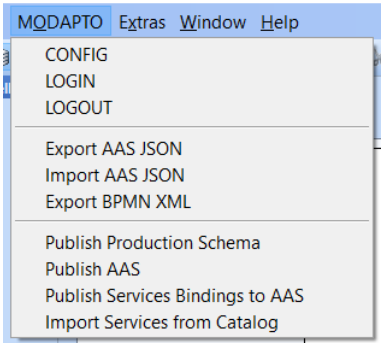


*Figure 5: MODAPTO functionalities menu*

The configuration section allows to specify all the endpoints of the different MODAPTO components (Figure 6) in particular of:

- servicesBaseUrl: Base URL of the Service Catalog
- keycloakBaseUrl: Base URL of the MODAPTO authentication system Keycloak
- keycloakRealm: REALM configured in Keycloak for MODAPTO
- keycloakClientId: Client Id configured in Keycloak for MODAPTO
- keycloakClientSecret: Client Secret configured in Keycloak for MODAPTO
- kafkaEndpoint: Endpoint of the MODAPTO Message Bus running on Kafka
- dtmanagerBaseUrl: Base URL of MODAPTO Digital Twin Manager

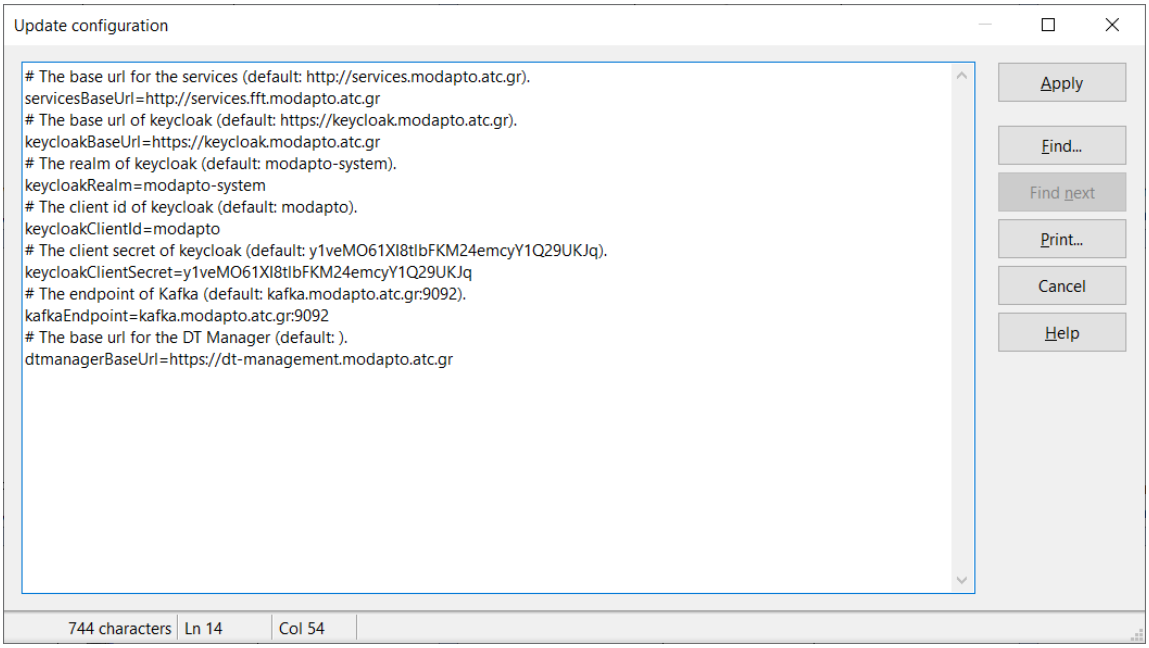


*Figure 6: MODAPTO Configuration*

In order to use all the publishing functions and connecting to other MODAPTO components the LOGIN functionality have to be performed. This will show a dialog to provide your personal credentials for accessing MODAPTO system and will show a confirmation message if they were correct (Figure 7).

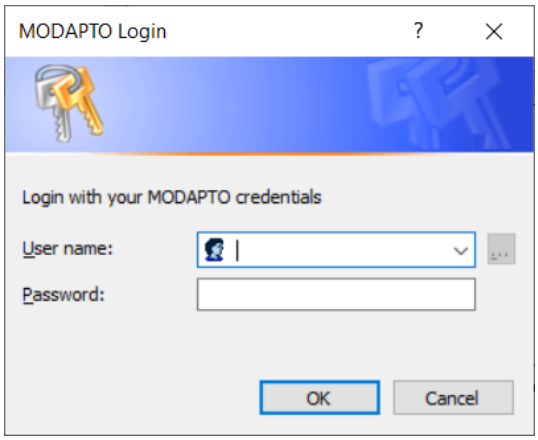


*Figure 7: MODAPTO Login UI*

After the login to the system is possible to:

- **Publish the production Schema**: The functionality will show a dialog to select the BPMN model you want to publish as Production Schema and generate a success of failure response.
- **Publish the AAS**: The functionality will show a dialog to select the AAS model you want to publish as MODAPTO Module and show a success of failure response.
- **Publish Service Binding**: The functionality will show a dialog to select the Data binding model you want to publish (assign the services to the published AAS) and show a success of failure response.
- **Import Services from Catalog**: The functionality is available when a Data binding model is opened and allows to first visualize a list of all the available services in the Catalogue, then after the selection of the desired services, the different services objects will be automatically created in the modelling canvas for the selected services.

Additional functionalities are also available for local testing, in particular:

- Export AAS JSON: Export locally the JSON definition of the selected AAS model.
- Import AAS JSON: Import an AAS model from a local JSON definition.
- Export BPMN XML: Export locally the standard BPMN definition of the selected BPMN model.

A complete documentation of the user interface functionalities of the modelling environment can be found at https://www.adoxx.org/documentation/02_introduction/05_user_interface.html.