# REFERENCE MANUAL

# ARIS – ALFABET INTEROPERABILITY INTERFACE

Documentation Version Alfabet 9.9.0

# CONVENTIONS USED IN THE DOCUMENTATION

| Convention | Meaning |
|---|---|
| Bold | Used for all elements displayed in the Alfabet interface including, for example, menu items, tabs, buttons, dialog boxes, page view names, and commands.<br>Example: Click **Finish** when setup is completed. |
| *Italics* | Used for emphasis, titles of chapters and manuals.<br>this<br>Example: see the *Administration* reference manual. |
| Initial Capi-tals | Used for attribute or property values.<br>Example: The object state Active describes... |
| All Capitals | Keyboard keys<br>Example: CTRL+SHIFT |
| File > Open | Used for menu actions that are to be performed by the user.<br>Example: To exit an application, select File > Exit |
| < > | Variable user input<br>Example: Create a new user and enter <User Name>. (Replace < > with variable data.) |
|  | This is a note providing additional information. |
|  | This is a note providing procedural information. |
|  | This is a note providing an example. |
|  | This is a note providing warning information. |

# TABLE OF CONTENTS



# Chapter 1: Introduction

Software AG provides two applications that support IT management and business process management:

- ARIS for business process analysis and management
- Alfabet for Enterprise Architecture Management, IT Planning and IT Portfolio Management

Together, these applications provide Software AG's customers with comprehensive and consistent planning and management support from the highest level of business assessment to the individual activities in IT planning and portfolio management.

Both ARIS and Alfabet maintain data about business process models and objects in the IT landscape in order to support planning and management capabilities. From the perspective of business process modelling, however, each tool provides unique functionalities. ARIS provides a wealth of views and methodologies to plan, develop, and enhance business processes on a detailed level. Entities in the IT landscape such as applications are included as modeling elements in order to map functions in the business process model to the application assets.

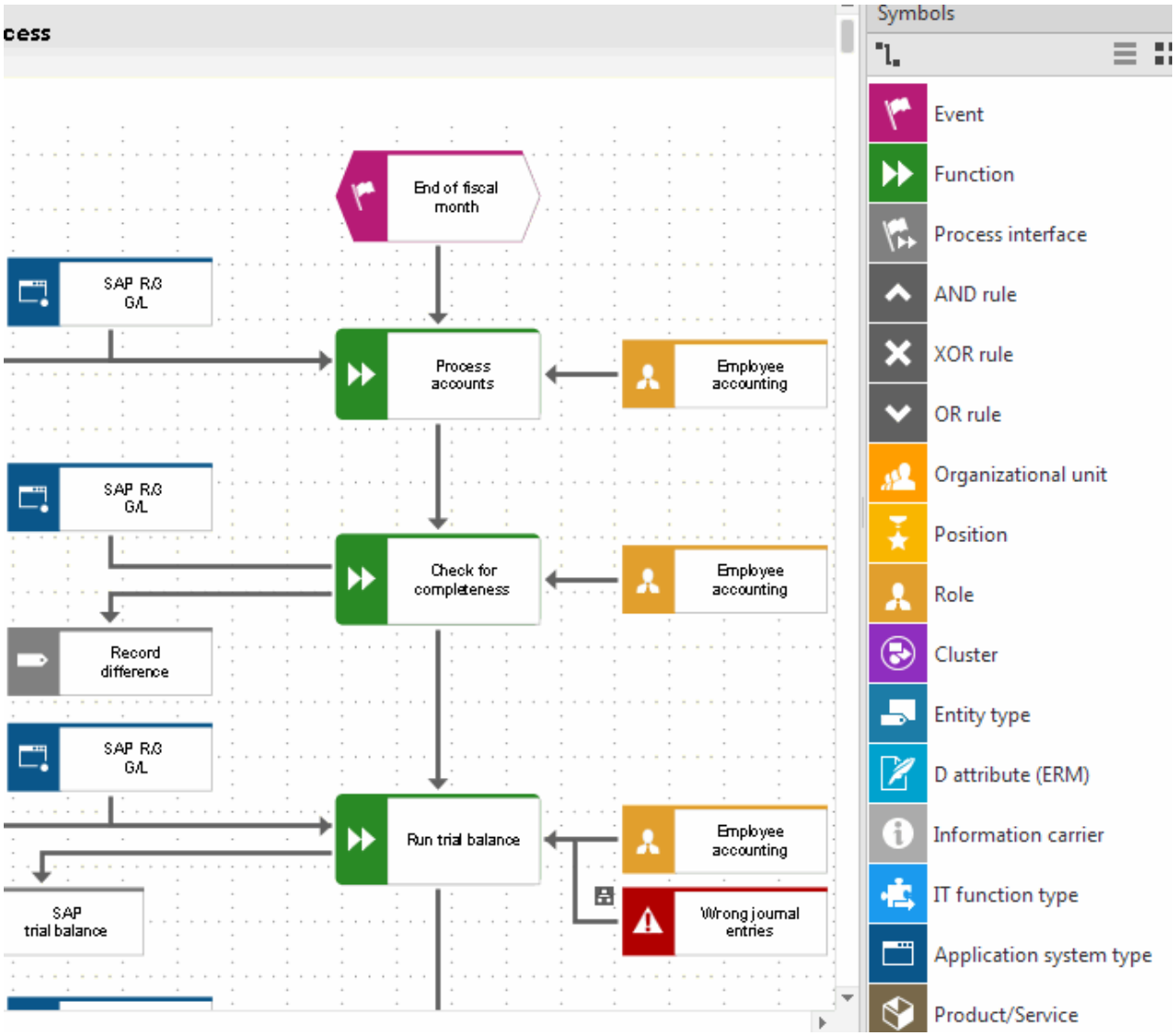



FIGURE 1: *ARIS EPC diagram of business process including applications (blue) used to execute functions (green)*

Alfabet focuses on the role of the IT landscape to support the execution of business processes. This includes, for example, release and deployment planning as well as IT-centric risk and compliance management. The business process model is used to map the applications - existing today or potentially existing some time in the future - to the relevant business processes in order to align IT support with business demands and to plan the development and streamlining of the IT landscape in strategy and master planning.

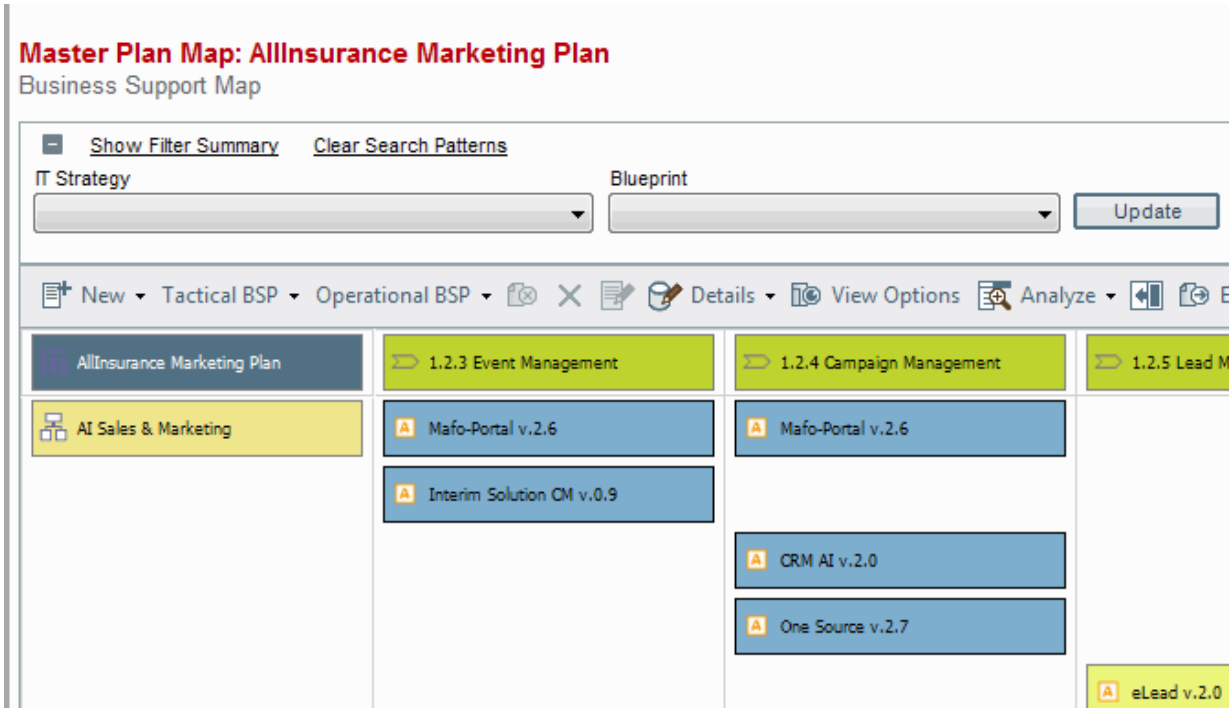



FIGURE 2: *Master Planning in Alfabet with supporting applications (blue) displayed per organization (yellow) and business process (green)*

Interoperability between the two solutions is now available that allows both ARIS and Alfabet to be used in synergy. Data maintained in either ARIS or Alfabet can be re-used in the other application to ease data maintenance and to prevent data inconsistencies between the complementary management tools. The ARIS - Alfabet Interoperability Interface allows for the regular synchronization of data between ARIS and Alfabet. The default configuration of the data synchronization is based on the following:

- Business process models are created and modelled in ARIS and the information about the business process model is transferred to the Alfabet database to be considered in enterprise architecture management.
- The enterprise architecture is managed in Alfabet. Only information about the IT landscape that is relevant for business process planning is transferred to the ARIS databases. In the standard configuration, information about applications as the central objects in the IT infrastructure is transferred to ARIS.

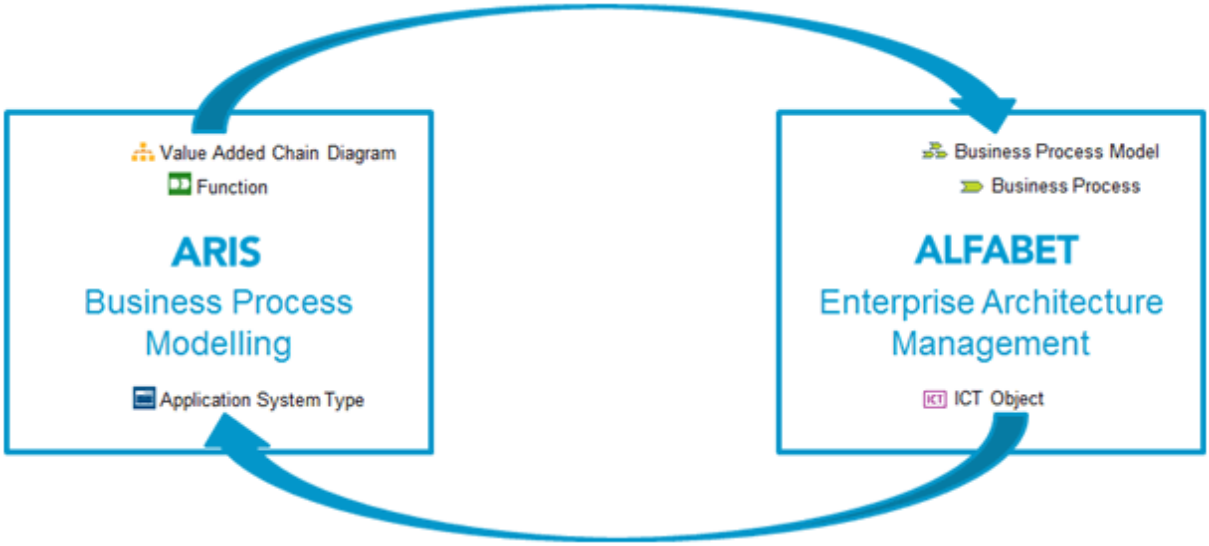



FIGURE 3: *Standard Data exchange configured for the ARIS - Alfabet Interoperability Interface*

The ARIS - Alfabet Interoperability Interface enables the user to easily traverse the user interfaces of ARIS and Alfabet. For example, when a user plans a business process model in ARIS , thereby leveraging application assets; the full 360° of the relevant application portfolios is just one click away.

Two versions of the ARIS - Alfabet Interoperability Interface are currently available. One is based on Web Services and the other, more advanced version, is based on a RESTful service architecture. The following table lists the main differences between the two versions of the ARIS - Alfabet Interoperability Interface as a basis for decision which version to use. In general, it is recommended to use the version based on a RESTful service architecture, because it is easier to implement and offers advanced handling of data exchange via the user interfaces of the applications.

| | RESTful API based ARIS - Alfabet Interoperability Interface | Web Service based ARIS - Alfabet Interoperability Interface |
|---|---|---|
| **Availability** | Alfabet 9.9 | Alfabet 9.6 and higher |
| **Involved components** | RESTful API at both ARIS and Alfabet | ARIS reports<br>ARIS Web Services<br>ARIS - Alfabet Interoperability Interface batch tool<br>Alfabet Web Services<br>Alfabet Data Integration Framework (ADIF) |
| **Data integration mode** | Batch and per object at runtime via the user interfaces | Batch update via a batch tool |
| **Business process model integration** | Existing ARIS business process models are directly integrated | Existing ARIS business process models are integrated via solution business |

| | RESTful API based ARIS - Alfabet Interoperability Interface | Web Service based ARIS - Alfabet Interoperability Interface |
|---|---|---|
| **from ARIS to Alfabet** | to Alfabet as business process models.<br><br>The hierarchy of business processes is derived from relations set in ARIS or alternatively from the hierarchy of diagrams in ARIS.<br><br>Any process model can be used for integration.<br><br>Changes to the ARIS business process model already integrated to Alfabet are displayed to the Alfabet user on the Alfabet user interface and the user can directly take over changes for the Alfabet business process model. | process models into Alfabet. After manual control the solution business process model is then checked in to the Alfabet database as business process model.<br><br>The hierarchy of business processes is derived from the hierarchy of diagrams in ARIS.<br><br>Only business process models of the type Value Chain Diagram can be used for integration.<br><br>Changes to the ARIS business process model already integrated to Alfabet are integrated to Alfabet via solution business process planning in parallel to the methods used for new business process models. |
| **Architecture element integration from Alfabet to ARIS** | Integration is performed either by pushing data of individual objects via the Alfabet user interface to ARIS or by performing update per object or batch update via the ARIS Connect interface. | By default, Alfabet applications are integrated via a batch job. Integration of data for other object classes requires configuration. |
| **Links between the user interface** | Users can navigate between the user interfaces of Alfabet and ARIS via links that are created per object by the user. | Users can navigate between the user interfaces of Alfabet and ARIS via links that are created during integration of object data for all integrated objects, that means for arc in ARIS and business process models |

This document describes the functionality provided by the integration interface between the ARIS and Alfabet applications, the installation of the interface including the required configuration, and the handling of the interface components. The document is focused on the new RESTful service architecture based ARIS - Alfabet Interoperability Interface. Nevertheless, the complete documentation of the Web Service based ARIS - Alfabet Interoperability Interface is included in a separate chapter The Web Service Based Version of the ARIS - Alfabet Interoperability Interface.

## Related Documents

The following documents are relevant for the functionality or configuration of the ARIS - Alfabet Interoperability Interface and provide more information about the technologies and models used:

## Documents Relevant for the Alfabet Application

| Consult: | For more information about: | Relevant for RESTful API based ARIS - Alfabet Interoperability Interface | Relevant for Web Service based ARIS - Alfabet Interoperability Interface |
|---|---|---|---|
| **Reference Manual *Alfabet RESTful API*** | Configuration and functionality of the Alfabet Interface for RESTful Web Services. | Yes | - |
| **Reference Manual *Alfabet Data Integration Framework*** | Configuring data import into and export from the Alfabet database with the Alfabet Data Integration Framework (ADIF). | - | Yes |
| **Reference Manual *Web Services for Alfabet*** | Working with the Alfabet Web Services. The document provides information about the structure of the WSDL and authentication for Web services. | - | Yes |
| **Alfabet Online Help** | Working with business process models and solution business process models in Alfabet. | Yes | Yes |

## Documents Relevant for the ARIS Application

| Consult: | For more information about: | Relevant for RESTful API based ARIS - Alfabet Interoperability Interface | Relevant for Web Service based ARIS - Alfabet Interoperability Interface |
|---|---|---|---|
| **Reference Manual *ARIS Server Installation and Administration Guide*** | INcludes the ARIS part of the RESTful API based ARIS - Alfabet Interoperability Interface | Yes | - |
| **Online Help *ARIS* and *ARISScript*** | Configuring reports in ARIS. | - | Yes |
| **Reference Manual *ARISMethods*** | Business Process Planning in ARIS. | Yes | Yes |

## Terminology in ARIS and Alfabet

In order to better understand this document, it is important to know about the differences regarding the terminology used in ARIS and Alfabet because different terms are sometimes used to describe similar concepts or the same term may be used differently in each application.

| ARIS | Definition | Alfabet |
|---|---|---|
| **Object Type** | A classification for the object stored in the database. In Alfabet , each object class has a separate database table. In ARIS , model type information is stored as attribute of the object in the database table for model objects. | Object Class |
| **Attribute** | Information stored about an object in the database (for example, Name and Type). In ARIS , model type attributes are defined centrally and assigned to one or multiple model types. In Alfabet , each object class has an individual set of properties. Object class properties relevant for multiple object classes are configured for each object class individually and changes made to an object class property configured for one object class (for example, changing the Name of a property) are not automatically applied to other object classes. | Object Class Property |
| **Fact Sheet** | Central view for an object to view and edit information stored about the object. The view also provides links to other relevant views.<br><br>In Alfabet , the object view/object profile is the central workspace for an object and highly customizable. Object profiles can either present information about the object and a list of links to the relevant views only (object view) or also directly display the content of graphic reports about the object (object cockpits). | Object View / Object Profile / Object Cockpit |
| **Report** | In ARIS , reports are JavaScript-based and either export data in a customized format to, for example, Excel, Word or XML, or perform actions on the object data. A set of preconfigured reports exists that can be customized. Customers can define their own reports as needed.<br><br>In Alfabet , reports are either tabular or graphic representations of data displayed in the Alfabet user interface. Customers can define their own reports as needed. In terms of the ability to batch export and import data, the Alfabet functionality that corresponds to ARIS Reports is the Alfabet Data Integration Interface (ADIF). ADIF allows import and export of data to be performed from/to Excel, Word or XML files, or from/to external databases. The export and import is based on customer -defined native SQL queries. These are combined with ADIF mechanisms that, for example, ensure database integrity during import. | ADIF |
| **Business Process Model** | In Alfabet , a business process model describes a hierarchy of business processes in the enterprise. In ARIS , a business process model describes in detail the way business processes are executed in the enterprise. A model can have multiple hierarchical levels that start with a general view on the business process and provides navigation to detailed diagrams that allow the execution of functions involved in a busi- | Business Process Model |

| ARIS | Definition | Alfabet |
|---|---|---|
| | ness process to be planned. | |
| **Function** | A set of activities that represent work required to achieve a business objective. In ARIS this is a *function* within a business process model, while in Alfabet this is a business process. | Business Process |
| **Application System Type** | An application is a fully-functional integrated IT product that provides functionality to end users and/or to other applications. In ARIS business process models applications are included as *application system type*. In Alfabet , the object class Application represents applications. | Application |

# Chapter 2: Functionality Provided By the RESTful Service Based ARIS - Alfabet Interoperability Interface

The following functionality is provided by the RESTful service based ARIS - Alfabet Interoperability Interface:

- Integrating ARIS Business Process Model Planning Data Into Alfabet :

Alfabet business process models can be created on basis of business process models already designed in ARIS. If the base business process model in ARIS is later changed, Alfabet users responsible for maintaining the business process model data in Alfabet can view a report about the changes and apply them to the Alfabet business process model. These processes are triggered by a user on the Alfabet user interface.

The way Alfabet business process models are derived from the ARIS business process model data is configurable. Configuration is described in the section Activating and Configuring the RESTful service based ARIS - Alfabet Interoperability Interface .

- *IntegratingAlfabetenterprise architecture management data intoARIS* :

Applications are a central element of enterprise architecture management in Alfabet and correspond to the ARIS*application system type*. The information about the applications maintained in Alfabet can be taken over to ARIS either in a batch process or on a per application basis. This integration is done via functionality available on the ARIS Architect user interface. Integration includes setting of a link to the relevant Alfabet object on the Alfabet user interface. Once set, the links are also available in ARIS Connect.

The implementation of the ARIS - Alfabet Interoperability Interface and the handling of the per object and batch data integration is described in the Reference Manual *ARIS Server Installation and Administration Guide* that is part of the ARIS documentation.

- *Navigation between the user interfaces ofARISandAlfabet* :

The navigation between the user interfaces of Alfabet and ARIS is based on links in one of the applications that open the other application on the linked target view. These links are set differently in Alfabet and ARIS:

  - In Alfabet a solution designer can configure which object classes allow setting of links to the ARIS user interface and which ARIS diagram types can be targeted by the links. On basis of the configuration, users can set links to ARIS diagrams as needed on the page view **ARIS Diagrams** of any object of the object classes configured to allow link setting. Navigation to the ARIS user interface requires a login to the user interface prior to showing the linked view.

For information about the setting of links and navigation to the ARIS user interface, see Navigation From the Alfabet to the ARIS User Interface in this reference manual.

For information about the required configuration of Alfabet to enable the navigation feature, see Activating and Configuring the RESTful service based ARIS - Alfabet Interoperability Interface in this reference manual.

  - In ARIS a link to the Alfabet user interface is set for each object of the application system type that is based on an Alfabet object of the object class application. The link is available in the attributes of the ARIS object. Navigation requires login to the Alfabet user interface. Please note that Read/Write access to the object is only granted if the user has the required access permissions to the object in Alfabet. Otherwise the object data is ReadOnly. Mandates are not supported. That means that a user accessing the Alfabet user interface via a link from ARIS can see the object regardless of the mandate settings.

The implementation of the ARIS - Alfabet Interoperability Interface and the handling of the navigation to the Alfabet is described in the Reference Manual *ARIS Server Installation and Administration Guide* that is part of the ARIS documentation.

The following information is available:



## Integrating ARIS Business Process Model Planning Data Into Alfabet

Alfabet business process models can be created on basis of business process models already designed in ARIS. If the base business process model in ARIS is later changed, Alfabet users responsible for maintaining the business process model data in Alfabet can view a report about the changes and apply them to the Alfabet business process model.

A business process model in Alfabet can be build automatically as copy of an existing business process model in ARIS. The process starts with a selectable ARIS business process model diagram and creates a business process hierarchy according to the configuration of the ARIS - Alfabet Interoperability Interface.

By default mapping is done as follows:

- For each function in the diagram a root business process is created regardless of any hierarchy within the diagram.
- If a function in the diagram is connected to other diagrams of the same type as the start diagram, the functions in the connected diagrams are added as child business processes to the business process representing the function. This is done for the complete hierarchy of functions and diagrams.
- For all imported functions a link to the diagrams that the object is assigned to is added automatically to the **ARIS Diagrams** page view of the business process in Alfabet.

Optionally, the ARIS - Alfabet Interoperability Interface can be configured to perform mapping based on connections between objects in ARIS:

- A hierarchy of business processes is build from the in the diagram according to the connections defined between the functions. Which type of connection is relevant for building the hierarchy is configurable in the ARIS - Alfabet Interoperability Interface. Optionally, your company may have configured the ARIS - Alfabet Interoperability Interface to import other ARIS object types as business processes.
- The process then follows the hierarchy of relevant ARIS connections of the objects in the diagram to other diagrams of the same type or of any other type configured as relevant in the ARIS - Alfabet Interoperability Interface and includes all connected objects in these diagrams to the hierarchy.

- For all imported functions a link to the diagrams that the object is assigned to is added to the **ARIS Diagrams** page view of the business process in Alfabet.

If the Alfabet business process model is build on basis of connections defined between objects in ARIS. Prior to starting import of an ARIS business process model make sure that connections are consistently set in the involved ARIS business process diagrams.

## Creating an Alfabet Business Process Model on Basis of an ARIS Business Process Model

To create an Alfabet business process model on basis of an existing ARIS business process model, you must first create a business process model in the business processes explorer in Alfabet and then link the Alfabet business process model to the ARIS business process model:

1) Click the **Business Processes** node  at the top of the **Business Processes** explorer.
2) In the toolbar, click **New** > **Create New Business Process Model**. The **Business Process Model** editor opens.
3) Enter information into each field, as required.

**Basic Data** tab:

- **ID**: Alfabet assigns a unique identification number to each object in the inventory. This number cannot be edited.
- **Name**: Enter a name for the business process model.
- **Description**: Enter a meaningful description that will clarify the purpose of the business process model.

**Authorized Access** tab:

- **Authorized User**: Click the **Search** icon to assign an authorized user to the selected object. The authorized user will have Read/Write access permissions to the object and is authorized to maintain the object in Alfabet.
- **Authorized User Groups**: Select the checkbox to assign Read/Write access permissions to all users in the selected user group. For more detailed information about the concept of access permissions, see the section *Understanding Access Permissions in Alfabet* in the reference manual *Getting Started with Alfabet*.

4) Click **OK** to save the business process model.
5) Click the new business process model in the explorer. The object profile of the business process model opens.

Depending on the configuration of your Alfabet solution, an object cockpit might open. You can change to the object profile via the link on the top left of the object cockpit.

6) In the object profile, click on **Business Processes**. The **Business Processes** page view opens.
7) In the toolbar, click **New** > **Get Business Process Model From ARIS...**. An Object Selector opens.
8) Enter search criteria into the **Search Pattern** field to find the relevant ARIS business process diagram and click the **Search** button.

9) Select the ARIS diagram from the list of search results and click **OK** to start the integration process. Business processes are created automatically from ARIS functions and structured according to relations defined in ARIS.

## Integrating Changes To an ARIS Business Process Model into the Existing Dependent Alfabet Business Process Model

A business process model diagram that is based on an ARIS business process diagram remains coupled with the ARIS parent diagram and if the ARIS diagram changes, these changes can be taken over to the Alfabet business process model. The Alfabet business process model derived from an ARIS business process model can be updated anytime with changes made to the ARIS business process model since creation of the dependent Alfabet business process model or since the last update of changes.

Update is performed on the **Business Process Model ARIS Source Report** page view of the business process model.

To open the view:

1) Click the new business process model in the **Business Processes** explorer. The object profile of the business process model opens.

   Depending on the configuration of your Alfabet solution, an object cockpit might open. You can change to the object profile via the link on the top left of the object cockpit.

2) In the object profile, click on **Business Process Model ARIS Source Report**. The **Business Process Model ARIS Source Report** page view opens.

The **Business Process Model ARIS Source Report** page view lists all changes that were performed in ARIS on the relevant business process diagrams.

Each column in the table is defined below:

- **Info**: An icon informing about the complexity of the change. The icon can have the following colors:
  - **green**: The change is not critical. For example a new function has been added to the ARIS diagram that would result in adding a new business process to the business process model in Alfabet , or the name or description of an existing business process has changed.
  - **yellow**: The change is not critical, but more complex. For example the structure of the business process model has changed and existing business process within the existing business process model will be moved to a new parent process during change integration.
  - **red**: The change is critical. For example a function has been removed from the ARIS diagram and the respective business process will be deleted from the Alfabet database during change integration.
- **Description**: A short description about which change was made to the ARIS diagram.

On basis of the information in the list you can decide which of the available import mechanisms to use:

- Directly Applying Changes to the Alfabet Business Process Model:

This mechanism directly changes the existing business process model in Alfabet. If a business process already exists, the changes in ARIS overwrite existing data in Alfabet but the business process is not completely overwritten. That means that a change in the description of the business process will overwrite the current description without affecting the other properties of the business process like the name or the relations defined to other objects in the Alfabet database. Business processes that are deleted in the

ARIS business process model are deleted in the Alfabet business process model and depending sub-objects and relations defined in Alfabet are also deleted.

- Applying Changes Via Solution Business Process Planning:

This mechanism creates a solution business process model for the selected business process model including all the changes performed in ARIS. Within the solution business process model, required changes concerning Alfabet specific data can be performed. The current business process model is substituted with the solution business process model after data was adjusted. This is for example relevant if a business process is deleted. The relations to Alfabet object that have been set for the business process can be redirected to other business processes in solution business process planning. Direct integration would delete the existing relations without notice.

Import is always performed on batch. It is not possible to select individual changes for import.

### Directly Applying Changes to the Alfabet Business Process Model

To directly integrate all changes that were made to the ARIS source business process model:

1) In the toolbar, click **New** > **Apply Changes**. All changes are applied automatically to the existing business process model. The report is empty after the action is performed.

### Applying Changes Via Solution Business Process Planning

Complex changes performed in ARIS should be applied via the business process planning functionality in Alfabet.

In business process planning, one or multiple copies of a business process model can be created as solution business process models. All business processes in the original business process model exist as solution business processes in the solution business process model.

At the start, each solution business process has the same property values as the business process it is based on. The information about which business process it is based on is stored in an additional property. The solution business processes in the solution business process model can then be edited, deleted and moved within the business process model while the original business process remains unchanged. New solution business processes (which are not based on an existing business process) can also be created within the business process model. A number of reports and planning tools are available about the potential impact to the IT landscape by the solution business process model. The person planning the business process model changes can then view the impact to the existing landscape without modifying an existing business process model.

Multiple solution business process models can be defined and compared to in order to select the best solution. Once a solution business process model has been approved, it can be checked in to the database. The original business process model will be overwritten with the solution business process model. A solution business process that is based on an existing business process overwrites that business process. Solution business processes that are new in the solution business process model are converted to new business process objects. Business processes that do not have a corresponding solution business process upon check in will be removed from the business process model and marked as no longer in use. The planned changes are now updated to the Alfabet business process model and the actual business process objects.

A new solution business process model is created for the business process model transferred from ARIS to Alfabet. If a solution business process model already exists for the business process model

and the already existing business process model is not yet checked in, it will be overwritten by the newly transferred business process model. An Alfabet user has to check in the solution business process model manually via the Alfabet user interface in order to integrate the changes to the current Alfabet business process models.

Data integration is based on the ARIS GUID, which is transferred with the data about each object. The GUID is stored in the Alfabet database as the property ARIS _GUID for the object classes Business Process, Business Process Model, Solution Business Process and Solution Business Process Model. The import of a business process model includes the following changes to objects:

1) A solution business process model is created for the business process model with the ARIS _GUID identical to the GUID of the transferred business process model.

   If a solution business process model from a prior import has not been checked in yet, this business process model will be overwritten with the currently imported data.

2) For all ARIS*functions* included in the transferred business process model, a solution business process with the ARIS _GUID set to the transferred GUID is created within the solution business process model.
3) For each solution business process within the solution business process model, a check is made to ensure that a business process with the same ALFA_GUID exists in the Alfabet database. If such a business process is found, all properties of the existing business process except for the Name and Description properties are copied to the new solution business process and the solution business process is assigned to overwrite the existing business process during check-in of the solution.
4) The business process model that will be overwritten by the solution business process model is then checked to see if there are any business processes that exist in the business process model but not in the solution business process model. In this case, the business process was removed from the business process model in ARIS. For each of these obsolete business processes, a solution business process will be created with a property **Deleted** set to True. When the business process model is overwritten by the solution business process model during check-in, the corresponding business process will be deleted.

After import, the user responsible for the maintenance of the business process model can consult reports available for the solution business process model to decide whether links from business processes to other objects in the Alfabet database require corrections prior to check in of the solution business process model. When the user has performed all required changes to the Alfabet -specific part of the business process properties, the solution business process model can be checked in manually by the responsible user. The changes performed in ARIS are mirrored in the Alfabet business process model structure only after the solution business process model has been checked-in.

To apply all changes that were made to the ARIS source business process model via a solution business process model:

1) In the toolbar, click **New** > **Create Business Process Solution...**. A solution business process model is created for the current business process model with all changes applied to it. The report is empty after the action is performed.
2) Perform the required corrections to the solution business process model and check it in to substitute the current business process model. For more information about business process planning see *Business Process Planning* .

## Navigation From the Alfabet to the ARIS User Interface

Alfabet users can set links to ARIS diagrams as needed on the page view **ARIS Diagrams** of any object of the object classes configured to allow link setting in the configuration of the ARIS - Alfabet Interoperability

Interface. Navigation to the ARIS user interface requires a login to the user interface prior to showing the linked view.

For each link to an ARIS diagram the following information is displayed on the page view **ARIS Diagrams**:

- **ARIS Diagram Name**: The name defined for the diagram in ARIS.
- **ARIS Diagram Description**: The description defined for the diagram in ARIS.

## Defining a New Link to an ARIS Diagram

To create a new link to an ARIS diagram:

1) In the toolbar, click **Add Aris diagrams...**. An object selector opens.
2) Select a diagram. You can use the **Search Pattern** field to limit the display of search results to a subset of the available diagrams.

   Which ARIS diagram types are displayed in the selector depends on the configuration of the ARIS - Alfabet Interoperability Interface.

3) Click **OK** to create the link. The information about the diagram is added to the table.

## Opening a Link to an ARIS Diagram

To open a link to an ARIS diagram:

1) Select the diagram in the table.
2) In the toolbar, click the **Navigate** button.

The ARIS user interface opens in a new tab of your browser. After login, the selected business process diagram is directly displayed. Which view about the diagram is displayed is configured by your solution designer in the ARIS - Alfabet Interoperability Interface.

## Updating the Information About ARIS Diagrams Displayed in the Table

During business process modelling in ARIS , diagrams for that a link was created in Alfabet may be removed from the business process model. Also, the name and the description attribute of the ARIS business process model diagrams may be changed in ARIS after the link was set. To update the information on the **ARIS Diagrams** page view with the current information about the diagram in ARIS:

1) In the toolbar, click **Refresh Links**.

# Chapter 3: Activating and Configuring the RESTful service based ARIS - Alfabet Interoperability Interface

An API based on RESTful service architecture, in the following called RESTful API, is implemented in both the Alfabet Web Application and ARIS Connect. To perform data integration, both Alfabet and ARIS must be configured to connect to the RESTful API of another and exchange data that is then automatically processed when a user performs an action involving data exchange on the respective user interface.

For example a user working with business processes in Alfabet can open a view **ARIS Diagrams** to configure a link to a relevant ARIS diagram. This functionality is only working if a connection to the ARIS RESTful API is configured for Alfabet.

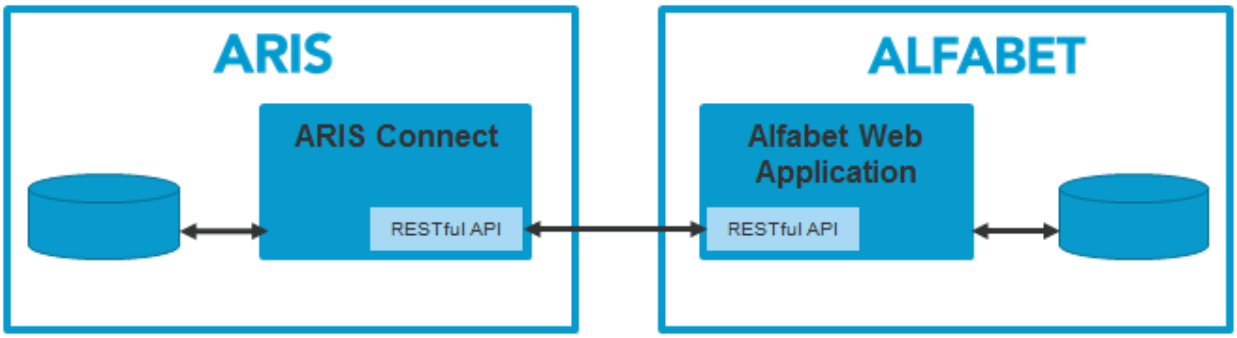



FIGURE 4: *Overview of the components involved in data exchange via the ARIS - Alfabet Interoperability Interface*

The configuration of the ARIS application is described in the reference manual *ARIS Server Installation and Administration Guide* provided with the ARIS documentation. This documentation is limited to the required configuration of the Alfabet application. It includes the following information:

- Configuring the Alfabet Web Application to Read Data from ARIS
  - Configuring the Mapping of Data for Business Process Model Integration
  - Configuring the Navigation to the ARIS User Interface
- Making the Views for the ARIS - Alfabet Interoperability Interface Available on the Alfabet User Interface
- Generating a User Based Access Key for ARIS Access to the Alfabet RESTful API

## Configuring the Alfabet Web Application to Read Data from ARIS

The configuration of the Alfabet application to connect to the RESTful API at ARIS is done in the configuration tool Alfabet Expand. It includes the following:, as well as configuration of mapping of

- Configuration of the connection to ARIS , which requires a connection to an API for authentication and to an API for database access.
- Definition of the ARIS model types that a user can set a link to. This restriction can be individually defined for each individual object class.
- Definition of the ARIS model types that are allowed as start model for business process model integration.

- Optional mapping of custom properties defined for the Alfabet object classes business process model and business process to ARIS attributes to include these properties into integration.

1) In Alfabet Expand , go to the Presentation tab and expand the explorer nodes **XML Objects** > **Administration**.
2) Right-click on **ArisApiConfig** and select **Edit XML...**. The XML object opens.
3) Configure the connection to the ARIS RESTful API by setting the following attributes of the XML element **ArisApiConfig** to the values of your ARIS installation provided by :
   - **active**: Set to true to activate the connection to the defined RESTful API.
   - **service**: Enter the URL to ARIS Connect.
   - **api**: Enter /abs/api. This attribute specifies the path to the ARIS RESTful API starting from the URL specified with the attribute **service**.
   - **auth_path**: Enter the path to the ARIS authorization API. This is usually /umc/api/tokens.
   - **auth_tenant**: Enter the tenant for authorization with the ARIS authorization API (for example " default ").
   - **auth_name**: Enter the user name for authorization with the ARIS authorization API.
   - **auth_password**: Enter the password for authorization with the ARIS authorization API.
   - **auth_key**: Enter the certificate for authorization with the ARIS authorization API.
   - **aris_db**: Enter the name of the ARIS database that includes the data relevant for data integration.
   - **aris_tenant**: Enter the name of the tenant for access to the ARIS database. The tenant must be identical to the tenant specified with the attribute **auth_tenant**.
   - **method_filter**: Enter the name of the filter that shall be applied when accessing the ARIS database (for example "Entire Method").
   - **data_portion**: Enter the maximum number of data sets that shall be exchanged within one transmission during the call. The default value is 50.
   - **search_limit**: Enter the maximum number of relevant objects to be retrieved from the ARIS database for a request. The default value is 200.
   - **link_view**: Enter "item ". This entry provides a mandatory technical term for that currently no other choices are implemented.
   - **link_itemtype**: Enter "default ". This entry provides a mandatory technical term for that currently no other choices are implemented
   - **languages**: Define the languages for that data shall be transmitted as locale ID with language and country code (for example de_DE or en_US ).
4) Make sure that the element ArisApiConfig contains the following child element:

```
<ClassMapping class="ARIS_DiagramLink">
   <AttributeMapping aris_attr="status" alfa_attr="ARIS_STATUS" />
</ClassMapping>
```

5) Activate the navigation to the ARIS user interface for the relevant Alfabet object classes. There is no default navigation behaviour and navigation can only be performed from object classes included in the configuration. The configuration is described below in the section Configuring the Mapping of Data for Business Process Model Integration.
6) Configure the mapping of object data between ARIS and Alfabet for business process models. A default behaviour is implemented and configuration is optional.

The configuration is described below in the section Configuring the Navigation to the ARIS User Interface.

7) In the toolbar of Alfabet Expand , click the **Save** button to save your changes.

The functionality of the ARIS - Alfabet Interoperability Interface is performed on special page views in the Alfabet user interface. The user profile of a user that shall use the functionality must be configured to show the view. For more information about the required views, see Making the Views for the ARIS - Alfabet Interoperability Interface Available on the Alfabet User Interface.

## Configuring the Mapping of Data for Business Process Model Integration

For the integration of ARIS business process models into Alfabet , a default behavior is configured that can be changed in the following ways:

- By default, business process models in Alfabet include all ARIS functions in a selectable source ARIS business process model diagram as root business processes and all functions in diagrams which are linked to the functions in the source diagram and are of the same type as the source business process model diagram as subordinate business functions. The hierarchy of diagrams linked to each other is used to build the hierarchy.

This default behaviour can be changed by adding one or multiple elements **ImportMapping** to the XML object **ArisApiConfig**. The integration processes is then changed in the following way:

  - The hierarchy of business processes is build on connections between objects configured in ARIS. This methods offers the advantage that a hierarchy can also be build for objects in the same diagram if they are interconnected with the connection type used for building the hierarchy. Which connection type is used for building the connection is configurable per diagram type.
  - Optionally other object types than functions can be regarded as business processes during integration. This functionality is only available for the integration process based on connections.

- By default, standard attributes of ARIS objects like name and description are taken over when an Alfabet business process is created. Both in ARIS and in Alfabet , customers can define custom attributes/properties to store company specific information. These customer defined information can be included in the integration process and taken over from ARIS to Alfabet when the business process model is build or updated.

To build the Alfabet business process model hierarchy on connections between ARIS objects:

1) In Alfabet Expand , go to the **Presentation** tab and expand the explorer nodes **XML Objects** > **Administration**.
2) Right-click on **ArisApiConfig** and select **Edit XML...**. The XML object opens.
3) Add the following child element to the **ArisApiConfig** element:

```
<ClassMapping class="BusinessProcessModel">
</ClassMapping>
```

4) Add one child element **ImportMapping** to the **ClassMapping** element for each ARIS diagram type that you would like to use as source business process model diagram during integration.
5) Set the following attributes for the **ImportMapping** element:
   - **model_type**: Enter the ARIS API name of the ARIS model type that shall be allowed to be used as starting point for integration. This attribute is mandatory.
   - **parent_rel**: Enter the ARIS API name of the ARIS connection type that shall be allowed to be used as starting point for integration. This attribute is mandatory.

- **proc_type**: Define the object classes that shall be regarded as business processes during integration in a comma separated list of ARIS API names of the object types. This definition is optional. If the attribute is empty or not set, only ARIS functions are regarded as business processes. Please note that the list of classes is overwriting the default and the *function* object type (OT_FUNC) must be included into the specification to be regarded as business process if the **proc_type** is defined.

6) In the toolbar of Alfabet Expand , click the **Save** button to save your changes.

The following configuration allows Value-Added Chain Diagrams (MT_VAL_ADD_CHN_DGM) and Function Tree Diagrams (MT_FUNC_TREE) as starting point for integration of business process models into Alfabet. For Value-Added Chain Diagram the business process hierarchy is build on the connection "is process oriented-superior" (CT_IS_PRCS_ORNT_SUPER) and ARIS*functions* (OT_FUNC) and *processes* (OT_PROC) are regarded as business processes in Alfabet. For Function Tree Diagram the business process hierarchy is build on the connection "is component of" (CT_CONS_OF_1) and the default behaviour converting ARIS*functions* to Alfabet business processes applies:

```
<ArisApiConfig ...>
   <ClassMapping class="BusinessProcessModel">
      <ImportMapping model_type="MT_VAL_ADD_CHN_DGM"
      parent_rel="CT_IS_PRCS_ORNT_SUPER"
      proc_type="OT_FUNC,OT_PROC"></ImportMapping>
      <ImportMapping model_type="MT_FUNC_TREE"
      parent_rel="CT_CONS_OF_1"></ImportMapping>
   </ClassMapping>
</ArisApiConfig>
```

To import ARIS attributes as Alfabet properties into the Alfabet business processes during integration:

1) In Alfabet Expand , go to the **Presentation** tab and expand the explorer nodes **XML Objects** > **Administration**.
2) Right-click on **ArisApiConfig** and select **Edit XML...**. The XML object opens.
3) Add the following child element to the **ArisApiConfig** element:

```
<ClassMapping class="BusinessProcess">
</ClassMapping>
```

4) Add one child element **AttributeMapping** to the **ClassMapping** element for each attribute that you want to include into the integration process.
5) Set the following attributes for the **AttributeMapping** element:
   - **aris_attr**: Enter the ARIS API name of the source ARIS attribute. This attribute is mandatory.
   - **alfa_attr**: The value of the Name attribute of the target Alfabet object class property. This attribute is mandatory.
6) In the toolbar of Alfabet Expand , click the **Save** button to save your changes.

The following configuration imports an ARIS status value to a custom property ARIS_STATUS in Alfabet:

```
<ArisApiConfig ...>
   <ClassMapping class="BusinessProcess">
      <AttributeMapping aris_attr="status" alfa_attr="ARIS_STATUS" />
```

```
    </ClassMapping>
</ArisApiConfig>
```

## Configuring the Navigation to the ARIS User Interface

An Alfabet user can only set links to diagrams in the ARIS user interface if the diagram types that are defined in the **ArisApiConfig** XML object in Alfabet Expand.

To activate and configure navigation from the Alfabet to the ARIS user interface:

1) In Alfabet Expand , go to the **Presentation** tab and expand the explorer nodes **XML Objects** > **Administration**.
2) Right-click on **ArisApiConfig** and select **Edit XML...**. The XML object opens.
3) For each object class for that navigation to the ARIS shall be configured, add a child element ClassMapping to the **ArisApiConfig** element with the attribute class set to the value of the Name attribute of the Alfabet object class.
4) Add one child element **ModelLink** to the **ClassMapping** element for each ARIS business process model diagram that shall be allowed as link target.
5) Set the following attributes for the **ModelLink** element:
   - **type**: Enter the ARIS API name of the ARIS model type that shall be allowed as link target. This attribute is mandatory.
   - **status**: Enter the ARIS status of the ARIS model type that shall be allowed as link target. If all status values shall be included, enter "all". This attribute is mandatory.
6) In the toolbar of Alfabet Expand , click the **Save**  button to save your changes.

The following configuration allows users to navigate from the Alfabet object class business process to four different diagram types in ARIS:

```
<ArisApiConfig ...>
    <ClassMapping class="BusinessProcess">
        <ModelLink type="MT_VAL_ADD_CHN_DGM" status="all"></ModelLink>
        <ModelLink type="MT_EEPC" status="all"></ModelLink>
        <ModelLink type="MT_BPMN_COLLABORATION_DIAGRAM"
        status="all"></ModelLink>
        <ModelLink type="MT_BPMN_PROCESS_DIAGRAM" status="all"></ModelLink>
    </ClassMapping>
</ArisApiConfig>
```

## Making the Views for the ARIS - Alfabet Interoperability Interface Available on the Alfabet User Interface

The user can execute the functionalities that require a connection to ARIS on the Alfabet user interface in the views listed in the following table. The Alfabet solution must be configured to provide access to the views for the relevant users. The required configuration is listed in the table.

| View Name | View Caption | Available for Alfabet Object Class | Functionality for ARIS - Alfabet Interoperability Interface | |
|---|---|---|---|---|
| **PRMD_RootProcesses** | Business Processes | Business Process Model | Button interaction to start integration of an ARIS business process model into the current, empty business process model. | The page view is a standard view that is also used for building business process models directly in Alfabet. The button is automatically available if<br>• The attribute **active** of the **ArisApiConfig** element is set to " true " in the XML object **ArisApiConfig** and<br>• the business process model does not have business processes assigned. |
| **ARIS_BMPSource_Report** | Business Process Model ARIS Source Report | Business Process Model | Check and update of changes for Alfabet business process models based on an ARIS business process model. | The user profile of a user that shall use the functionality must be configured to show the view. This requires that the view is added to the object profile of the business process model used for the user profile and the view is configured to be visible.<br>For information about the configuration of user profiles consult the reference manual *Configuring Alfabet with Alfabet Expand*. |

| View Name | View Caption | Available for Alfabet Object Class | Functionality for ARIS - Alfabet Interoperability Interface | |
|---|---|---|---|---|
| **ARIS_DiagramLinks** | ARIS Diagrams | any object class for that it is configured | Definition of links to the ARIS user interface and navigation via the links. | The user profile of a user that shall use the functionality must be configured to show the view. This requires that the view is added to the object profile of the business process model used for the user profile and the view is configured to be visible.<br><br>For information about the configuration of user profiles consult the reference manual *Configuring Alfabet with Alfabet Expand*. |

## Generating a User Based Access Key for ARIS Access to the Alfabet RESTful API

The connection to the to the Alfabet RESTful API requires a user related token for authentication that has to be entered into the configuration at ARIS. The token must be generated with the tool Alfabet Expand and provided to the person responsible for configuring the ARIS - Alfabet Interoperability Interface at ARIS side. The required configuration of the access to the Alfabet RESTful API is described in the reference manual provided by ARIS.

To generate the token for access to the Alfabet RESTful API:

1) In the ADIF tab of Alfabet Expand , right-click the root node and select **Generate User REST API Token**. A new window opens.
2) Select the user name from the drop down list in the field **Select User**.
3) Click **Generate Token**. A code is generated, stored in the Alfabet database and displayed in the field **Generated Token**.
4) Copy the code and store the information about user name and code for use on ARIS side. You can use the Copy to Clipboard button to copy the token to the clipboard.

The generated code must be used for authorization in combination with the user name.

# Chapter 4: The Web Service Based Version of the ARIS - Alfabet Interoperability Interface

The Web Service based ARIS - Alfabet Interoperability Interface was developed prior to the ARIS - Alfabet Interoperability Interface based on a RESTful service architecture. It is available for Alfabet versions 9.6 and higher and requires a more complex configuration and handling as the successor version based on a RESTful service architecture. Data integration is done by a batch job independent from user interaction on the user interfaces.

The following information provides an overview of the functionality and technical architecture of the ARIS - Alfabet Interoperability Interface:



## Functionality Provided by the ARIS - Alfabet Interoperability Interface

The way data exchange and switching between the user interfaces of ARIS and Alfabet at runtime are described in the following:



## Data Exchange between the ARIS and Alfabet Databases

The Web Service based ARIS - Alfabet Interoperability Interface integrates changes to business process models performed in ARIS with the Alfabet database as well as changes made to the application architecture in Alfabet with the ARIS database. Data integration is performed via a batch job that is run by a system administrator and therefore decoupled from user interaction on the ARIS or Alfabet user interface.

The data integration does not include any mechanisms that prevent changes to application data in ARIS or the business process model in Alfabet. Any changes performed manually in ARIS to objects of the application system type or in Alfabet to objects of the object class Business Process Model or Business Process will be lost during data import via the Web Service based ARIS - Alfabet Interoperability Interface.

For example, a business process added to a business process model in Alfabet will be deleted at the next import if the business process does not correspond to a *function* in a Value Added Chain diagram in ARIS. The same applies to an object of the *application system type* defined in ARIS used in a business process model with no corresponding ICT object in Alfabet.

To address the problems that may result by objects being created or edited in the wrong place, the import mechanisms on both ARIS and Alfabet sides includes mechanisms to check and correct the use of these objects prior to deletion.

### Application Data Integration from Alfabet to ARIS

The Web Service based ARIS - Alfabet Interoperability Interface reads data about all ICT objects from the Alfabet database.

For each ICT object, the following object class properties are transferred to ARIS:

- Name
- Description
- GUID
- The Alfabet view that opens if a user directly accesses the Alfabet interface via the object.

For more information about navigation between the ARIS and Alfabet user interfaces, see the section Navigation between the ARIS and Alfabet User Interfaces.

On the ARIS side, the data is merged with the existing data in the database. The merge is based on two interface-specific ARIS attributes:

- **Alfabet- GUID**: Stores the GUID of the ICT object in the Alfabet database as an attribute of the *application system type* object.
- **Alfabet- Deletion Candidate**: This Boolean attribute is set to False per default. It is set to True for all objects that are deleted in Alfabet and therefore should also be deleted in ARIS.

During import, the existence of an object of the *application system type* with an attribute ALFA_GUID identical to the GUID of the imported application is checked. The action that will be performed during the import depends on the result of the check:

| ALFA_GUID in Import Data | ALFA_GUID in Database | Action in the ARIS database |
|---|---|---|
| **yes** | yes | The attributes of the object are updated with the import data. The object is assigned to the sub-group **active** of the **Main group**. |
| **yes** | no | A new object is created with the attributes from the import data and the object is assigned to the sub-group **active** of the **Main group**. |
| **no** | yes | The attribute **Alfabet- Deletion Candidate** is set to True and the object is assigned to the sub-group **delete** of the **Main group**. It is no longer assigned to the sub-group **active**. |

Two groups were added to the **Main group** for the management of objects managed via import from Alfabet:

- **active**: This group contains all objects of the *application system type* that are available both in the Alfabet and ARIS database.
- **delete**: This group contains all objects of the *application system type* that are no longer available in the Alfabet database but are still part of the ARIS database.

Both groups contain an Application System Type diagram that include all *application system type* objects in the group. In ARIS , each object needs to be included in at least one model or diagram to be editable. The Application System Type diagram ensures editability of all objects imported via Alfabet.

After data import, an ARIS user can check the **delete** group to check whether the applications in the group are used in any models and substitute them with another *application system type* object in each of the impacted models. After that, the object should be manually deleted from the ARIS database via the ARIS user interface.

### Business Process Model Integration from ARIS to Alfabet

The Web Service based ARIS - Alfabet Interoperability Interface reads data about business process models from a selected ARIS database that match the following preconditions:

- The model is a Value Added Chain diagram
- The attribute **AlfabetRoot Type** is set to Alfabet- Reference Model.

Each Value Added Chain Diagram matching the preconditions is imported as a separate business process model. All *functions* included in the Value Added Chain diagram are imported to the Alfabet business process model as first level business processes.

All *functions* included in Value Added Chain diagrams that are subordinate to the Value Added Chain diagram are imported to the Alfabet business process model in the hierarchical level corresponding to the level in the hierarchy of diagrams in ARIS. For example, a function in a Value Added Chain diagram directly subordinate to the root Value Added Chain diagram is imported as second level business process. The number of levels that may be imported to Alfabet is configurable. The default is set to 4 levels.

All Value Added Chain diagrams and *functions* with the Alfabet attribute **- Not to be exported** set to True are excluded from export.

For each exported object, the following attributes are imported to the Alfabet business process model:

- Name
- Description
- GUID
- The ARIS view that opens if a user directly accesses the ARIS interface via the object.

For more information about navigation between the ARIS and Alfabet user interfaces, see the section Navigation between the ARIS and Alfabet User Interfaces.

In Alfabet , the update of business process models based on the data imported from ARIS is performed via the business process planning functionality in Alfabet.

In business process planning, one or multiple copies of a business process model can be created as solution business process models. All business processes in the original business process model exist as solution business processes in the solution business process model.

At the start, each solution business process has the same property values as the business process it is based on. The information about which business process it is based on is stored in an additional property. The solution business processes in the solution business process model can then be edited, deleted and moved within the business process model while the original business process remains unchanged. New solution business processes (which are not based on an existing business process) can also be created within the business process model. A number of reports and planning tools are available about the potential impact to the IT landscape by the solution business process model. The person planning the business process model changes can then view the impact to the existing landscape without modifying an existing business process model.

Multiple solution business process models can be defined and compared to in order to select the best solution. Once a solution business process model has been approved, it can be checked in to the database. The original business process model will be overwritten with the solution business process model. A solution business process that is based on an existing business process overwrites that business process. Solution business processes that are new in the solution business process model are converted to new business process objects. Business processes that do not have a corresponding solution business process upon check in will be removed from the business process model and marked as no longer in use. The planned changes are now updated to the Alfabet business process model and the actual business process objects.

The Web Service based ARIS - Alfabet Interoperability Interface creates a solution business process model for each business process model transferred from ARIS to Alfabet if no solution business process model exists that has not yet been checked in. If a solution business process model which is not yet checked in exists, it will be overwritten by the newly transferred business process model. An Alfabet user has to check in the solution business process model manually via the Alfabet user interface in order to integrate the changes to the current Alfabet business process models.

Data integration is based on the ARIS GUID, which is transferred with the data about each object. The GUID is stored in the Alfabet database as the property ARIS _GUID for the object classes Business Process, Business Process Model, Solution Business Process and Solution Business Process Model. The import of a business process model includes the following changes to objects:

1) A check is performed whether a business process model with an ARIS _GUID exists that is identical to the GUID of the transferred business process model. If this is not the case, a new business process model with properties transmitted via the interface will be created.
2) A solution business process model is created for the business process model with the ARIS _GUID identical to the GUID of the transferred business process model.

If a solution business process model from a prior import has not been checked in yet, this business process model will be overwritten with the currently imported data.

3) For all ARIS*functions* included in the transferred business process model, a solution business process with the ARIS _GUID set to the transferred GUID is created within the solution business process model.

4) For each solution business process within the solution business process model, a check is made to ensure that a business process with the same ALFA_GUID exists in the Alfabet database. If such a business process is found, all properties of the existing business process except for the Name and Description properties are copied to the new solution business process and the solution business process is assigned to overwrite the existing business process during check-in of the solution.

5) The business process model that will be overwritten by the solution business process model is then checked to see if their are any business processes that exist in the business process model but not in the solution business process model. In this case, the business process was removed from the business process model in ARIS. For each of these obsolete business processes, a solution business process will be created with a property **Deleted** set to True. When the business process model is overwritten by the solution business process model during check-in, the corresponding business process will be deleted.

After import, the user responsible for the maintenance of the business process model can consult reports available for the solution business process model to decide whether links from business processes to other objects in the Alfabet database require corrections prior to check in of the solution business process model. When the user has performed all required changes to the Alfabet -specific part of the business process properties, the solution business process model can be checked in. The changes performed in ARIS are mirrored in the Alfabet business process model structure only after the solution business process model has been checked-in.

## Navigation between the ARIS and Alfabet User Interfaces

The data transfer between ARIS and Alfabet also activates the direct navigation between the ARIS and Alfabet user interfaces. Multiple links can be defined per object to enable navigation to different views that are relevant for different contexts. For example, a user can navigate from a business process in Alfabet to the fact sheet of the corresponding *function* in ARIS or to a diagram that the *function* is part of.

If a user opens a fact sheet of an *application system type* in ARIS , the attribute section includes the links to the Alfabet views of the ICT object representing the application:

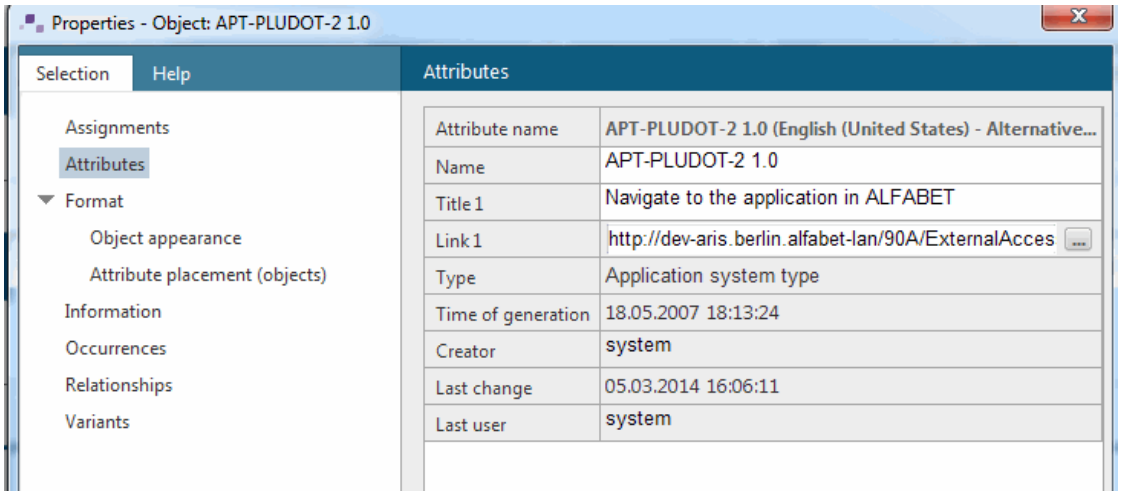



FIGURE 5: *Link to the Alfabet user interface from an ARIS application system type fact sheet*

Clicking on the link opens the Alfabet user interface in a new tab of the browser, displaying the data about the ICT object.

If the user is not currently logged in to Alfabet , the user may have to enter his/her user name and password in a login window prior to viewing the data. Whether login with user name and password is required depends on the configuration of the access to the Alfabet user interface. For example, if Windows Sign On is performed, no login screen will be displayed and login credentials will be checked automatically.

If a user works with a business process or solution business process in Alfabet , a link to the ARIS user interface is included as a web link to the **Attachments** view of the business process.

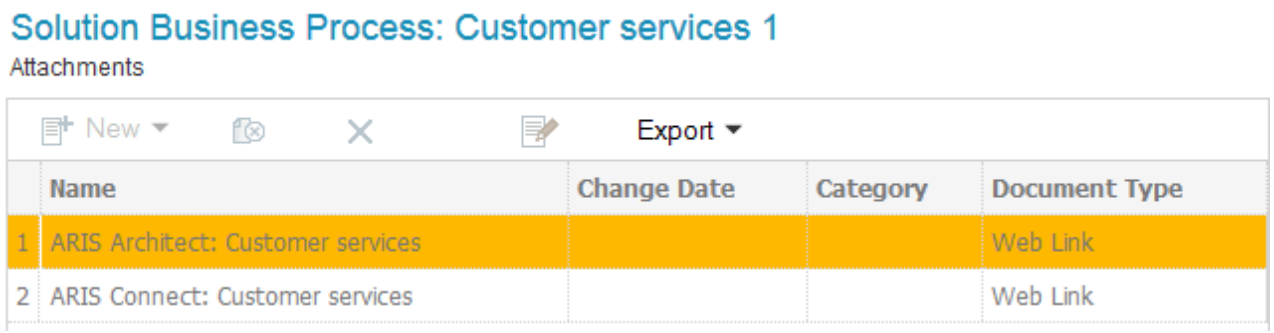



FIGURE 6: *Link to the ARIS user interface from a business process in Alfabet*

Clicking on the link opens a small pane that provides information about the link. The button **Show Details** in the information pane opens the ARIS user interface in a new tab of the browser, displaying the relevant data about the business process.

If the user is not currently logged in to ARIS , the user may have to enter his/her user name and password in a login window prior to viewing the data.

## Interface Architecture

The interface is based on regular data exchange between the components. Data exchange is triggered by a separate software component, the A2A Integrator, that can either be called via a command line on demand or executed in regular intervals via a job scheduling engine. The A2A Integrator calls web services to connect to both the Alfabet Server and the ARIS server.

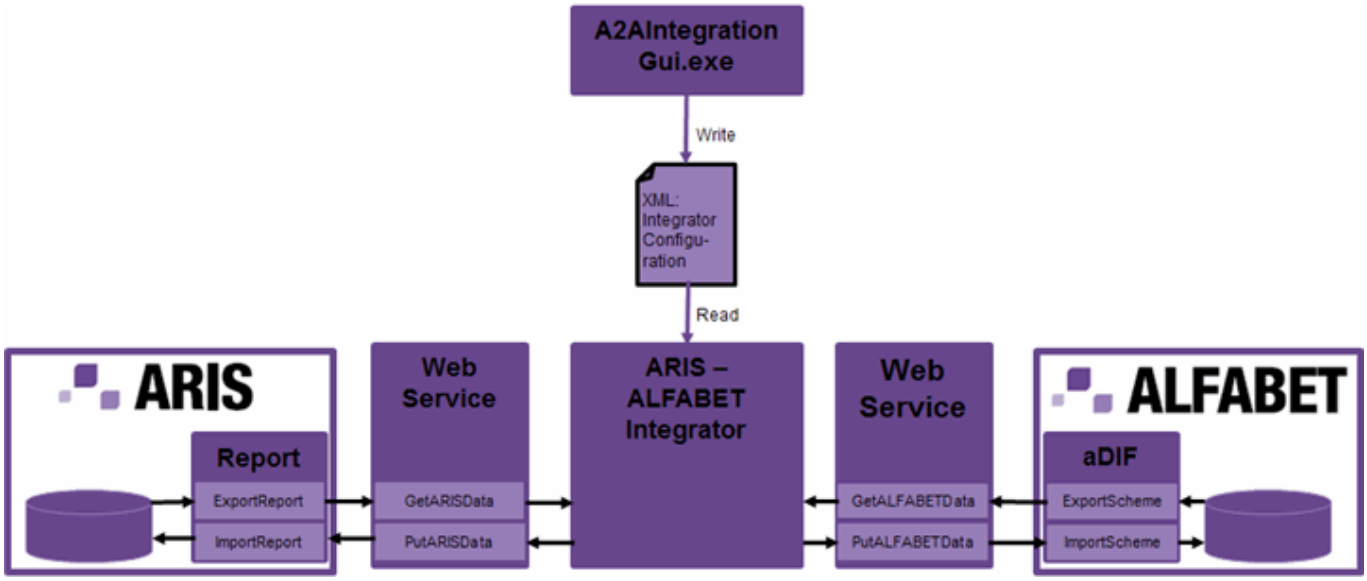



FIGURE 7: *Overview of the technical components of the Web Service based ARIS - Alfabet Interoperability Interface*

Data processing also involves the following mechanisms to ensure data integrity in the target databases and to provide a means for customer configuration of the interface:

- On Alfabet side:
  - The Alfabet Data Integration Framework (ADIF).

ADIF enables data import to or export from the Alfabet database based on ADIF internal mechanisms in combination with native SQL commands. The import or export definition is stored as an ADIF scheme in the Alfabet database. For the Web Service based ARIS - Alfabet Interoperability Interface , the web services trigger the execution of preconfigured, interface-specific ADIF schemes.

- ON ARIS side:
  - JavaScript-based ARIS evaluation reports:

The reports enable data import to or export from the ARIS databases. For the Web Service based ARIS - Alfabet Interoperability Interface , the web services trigger the execution of preconfigured, interface-specific reports.

The Web Service based ARIS - Alfabet Interoperability Interface not only transmits data directly between the ARIS and Alfabet databases. Data can be exported to an XML file and imported from an XML file. This implementation is useful for testing and for indirect data integration in a scenario where data export and import shall be executed at different times or in disconnected environments.

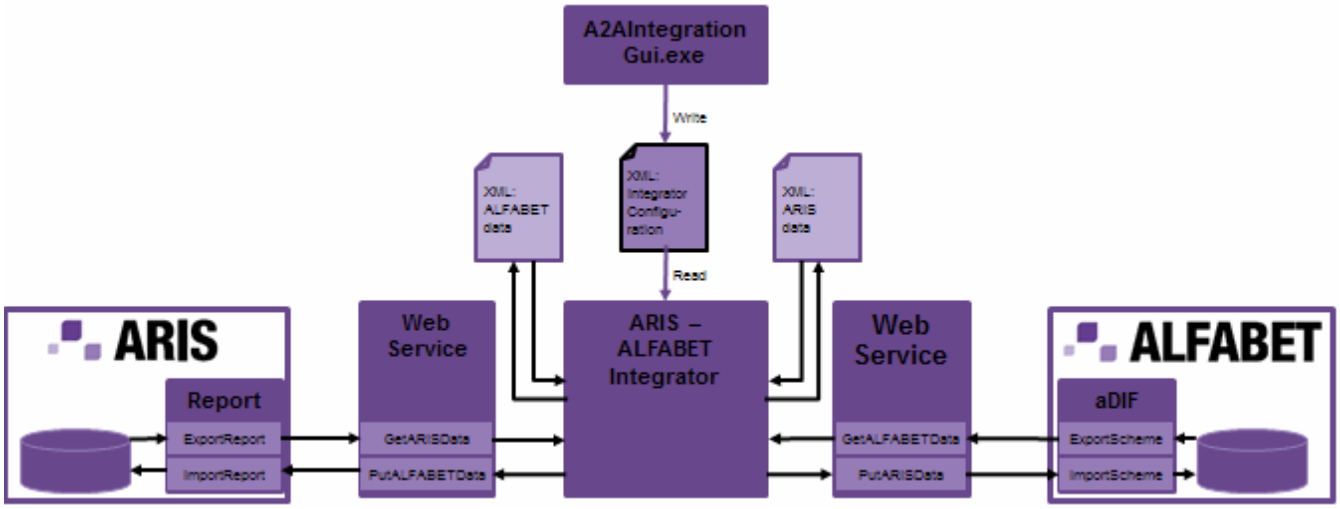



FIGURE 8: *Overview of the technical components including all import sources and export targets*

## Basic Initial Set-Up of the Web Service based ARIS - Alfabet Interoperability Interface

### Pre-Installation Requirements

The following components of the ARIS and Alfabet application are required for execution of the functionalities of the ARIS - Alfabet integration interface:

| Components required for | ARIS | Interface | Alfabet |
|---|---|---|---|
| **Basic set-up, data transfer** | ARIS database on database server (third party component)<br>ARIS Connect Server<br>ARIS Design Server | ARIS Web Service on Web Server (third party component)<br>ARIS - Alfabet Integrator<br>Alfabet Web Service on Web Server (third party component) | Alfabet Server<br>Alfabet database on database server (third party component) |
| **Configuration** | ARIS Architect (optional, only for changes to the standard)s | A2AIntegrationGui.exe | Alfabet Expand (optional, only for changes to the standard) |
| **Navigation between the user interfaces** | ARIS Architect<br>or ARIS Connect Viewer<br>or ARIS Design Client | | Alfabet Web Application<br>Alfabet Web Client |

The figure provides an overview of all components involved in the interface functionality.

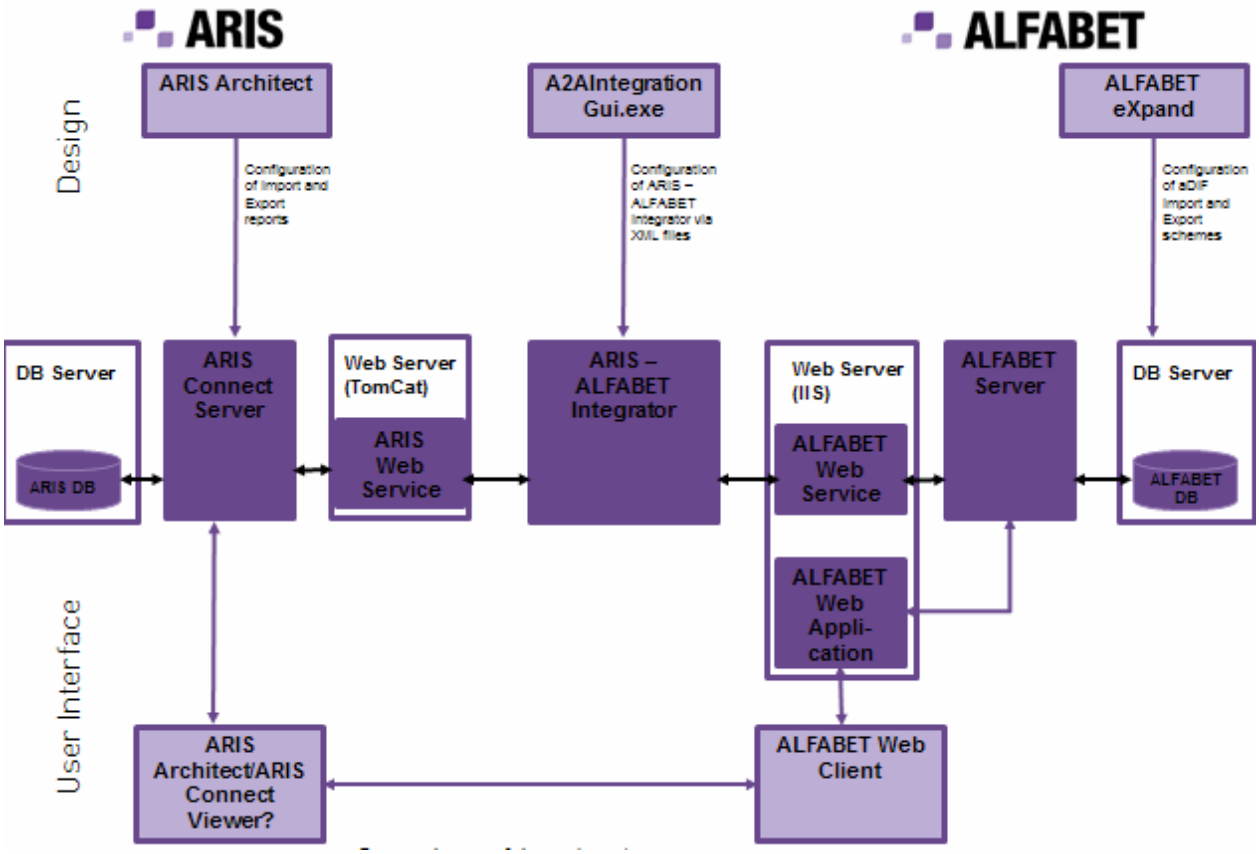



FIGURE 9: *Components involved in the Web Service based ARIS - Alfabet Interoperability Interface*

The technical requirements for the relevant ARIS and Alfabet components apply. For an overview of the technical requirements of ARIS and Alfabet components, consult the documentation of the respective application. The interface does not have any additional requirements.

## Installation

The following parts of the interface are standard components of ARIS 9.6 and Alfabet 9.6 and do not require any further installation:

- The Alfabet ADIF schemes.
- The ARIS reports
- The Alfabet -related attributes in ARIS.

The following overview of the installation procedures required to install the Web Service based ARIS - Alfabet Interoperability Interface presuppose that both ARIS and Alfabet are already installed and is limited to the procedure required to install the interface only:

Installation required on the ARIS side:

- Implement the ARIS web services as a Web Application.

The setup procedure depends on the Web server used. Consult the documentation of your Web server for information about how to set up a Web service. For more information about web service implementation for ARIS , consult the ARIS documentation.

Installation required on the Alfabet side:

- Set up the Alfabet web services on a Web server as a Web Application.

The setup procedure depends on the Web server used. Consult the documentation of your Web server for information about how to set up a Web service. For more information about Alfabet web services, consult the Alfabet documentation.

Installation of the ARIS - Alfabet Integration Interface application:

- For customers with a license for the Web Service based ARIS - Alfabet Interoperability Interface , the required executables are located in the **programs** directory of the Alfabet components. If the interface is licensed for a running version of Alfabet 9.9 , the executables are delivered separately and must be copied to the **programs** directory of the Alfabet components.
- Configure the integration tool. The required configuration is described in the section Configuration.

## Configuration

The following information is available about the configuration required for the Web Service based ARIS - Alfabet Interoperability Interface:

- Configuring the ARIS Data for Import to Alfabet
- Configuring the Web Service based ARIS - Alfabet Interoperability Interface

### Configuring the ARIS Data for Import to Alfabet

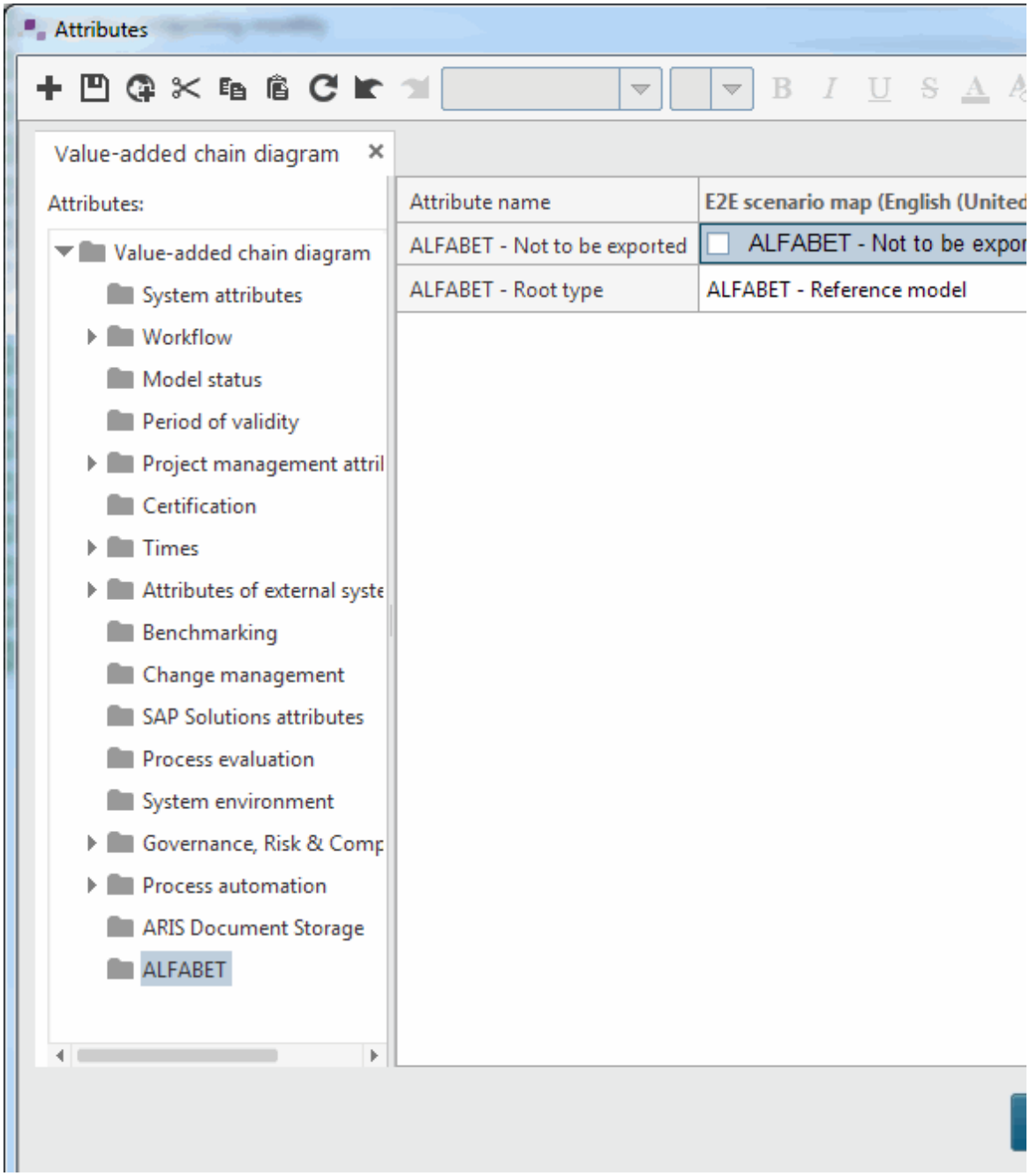
Data export from the ARIS database requires the following configuration of data before implementing the Web Service based ARIS - Alfabet Interoperability Interface for data transmission:

Ensure that you use a method filter that shows the required attributes for the Web Service based ARIS - Alfabet Interoperability Interface. The method filter "Entire Method" enables setting of the required attributes. The attributes are members of the attribute group Alfabet.

- The business process models that are to be re-used in Alfabet must be modelled in Value Added Chain diagrams.
- For each Value Added Chain diagram that is to be imported as a business process model in Alfabet , set the attribute **AlfabetRoot Type** to **Alfabet- Reference Model**.

To set the attributes of a Value Added Chain diagram or any other object in ARIS , right-click the object in the Explorer and select, for example, **Attributes**. In the **Attributes** window that opens, click Alfabet in the **Attributes** explorer. The attributes related to the Web Service based ARIS - Alfabet Interoperability Interface are displayed.




- Functions that are part of the root Value Added Chain diagram and all subordinate Value Added Chain diagrams are exported and imported to Alfabet as business processes. To exclude a function or all functions of a subordinate Value Added Chain diagram from export, select the check box of the attribute **Alfabet- Not to be exported** in the Attribute window of the function or Value Added Chain diagram.

### Configuring the Web Service based ARIS - Alfabet Interoperability Interface

The connection to the Alfabet and ARIS components must be configured in the configuration file for the Web Service based ARIS - Alfabet Interoperability Interface. Multiple configuration files can be created and stored in parallel.

To configure the Web Service based ARIS - Alfabet Interoperability Interface:

1) In the in the **programs** directory of the Alfabet components , double click A2AIntegrationGui.exe. An interface opens.

2) In the interface, open the **ARISAccess Configuration** tab and configure the call to the ARIS web services:

   - **ARISWEB Service**: Enter the URL of the ARIS web services for the integration of Alfabet data.
   - **ARISaccess configuration**:
     - **User Name**: Enter the user name for access to the ARIS connect server.
     - **Password**: Enter the password for access to the ARIS server.
   - **ARISdata access configuration**:
     - **ARISDatabase**: Enter the name of the ARIS database that is target for the data import.
     - **ARISTenant**: Enter the name of the tenant for accessing the database.
   - **Export**:
     - **ARISExport Report**: Optionally enter the name of the ARIS report that triggers data export for the ARIS - Alfabet Interoperability Interface. If this field is left blank, the standard report **ARIS2Alfabet- Export processes** is executed.
     - **ZIP export**: Select the check box if the XML export data shall be compressed in a ZIP file.
   - **Import**:
     - **ARISImport Report**: Optionally enter the name of the ARIS report that triggers data import for the ARIS - Alfabet Interoperability Interface. If this field is left blank, the standard report **Alfabet2ARIS- Import application systems** is executed.
     - **ARISImport Expects ZIP**: Select the check box if XML data compressed in a ZIP file shall be imported.

3) Open the **Alfabet Access Configuration** tab and configure the call to the Alfabet web services:

   - **Alfabet WEB Service**: Enter the URL of the Alfabet web services for the integration of ARIS data. To test the connection to the web services, click **Check Connection**. A log message regarding the success of the connection is displayed in the field in the bottom of the pane.
   - **Alfabet Application Server Configuration**:
     - **Alfabet Application Server URI (tcp protocol)**: The Alfabet Web Service connects to the Alfabet Server via TCP. Enter the path to the Alfabet Server as:

       ```
       tcp://<host>:<port>/<alfabetServerAlias>
       ```

       For example:

       ```
       tcp://127.0.0.1:1880/Alfabet
       ```

     - **Ensure Security**: Select the check box if the .NET EnsureSecurity shall be used to secure the communication channel to the Alfabet Server.

       The settings for host name or IP address, port number, and ensure security must be identical to the specifications in the server alias configuration of the Alfabet Server. For more information about the server alias configuration, see the reference manual *System Administration* in the Alfabet documentation.

   - **Alfabet DB access Configuration**:

     The Alfabet Web services connect to the Alfabet database via the Alfabet Server. Nevertheless, authentication occurs directly on the database level by providing a user name and password for login to the database on the database server.

- **User Name**: Enter the name of the database user for log in to the Alfabet database on the database server.
- **Password**: Enter the password for log in to the Alfabet database on the database server.

- **Export**:
  - **ADIF Export Job**: Enter the name of the ADIF export scheme that shall be executed for data export from Alfabet. To use the standard functionality of the ARIS - Alfabet integration interface, enter ARIS_Export.
  - **Parameters**: For use of the standard Web Service based ARIS - Alfabet Interoperability Interface , enter the URL of the ARIS user interface as:

    ```
    Html5UI=<URLForAccessToalfabetHTML5WebClient>,ObjectView=<ObjectViewToUse>
    ,UserType=<Named|Anonymous>
    ```

The URL is used to generate the links for access to the Alfabet user interface that are stored in ARIS as attributes of the imported object of the *application system type*. The link is built by connecting the URL defined here with the information about the individual object's view that shall be opened. This is generated during export by the ADIF export scheme. Both a link to the new HTML5 based user interface and the classic user interface can be included in parallel in a comma-separated string.

Additional parameters can be entered into the field if the export is based on a customer-defined ADIF scheme that is configured to perform data export based on parameters given in the command line call for ADIF scheme execution. Define the parameters and their values in a comma-separated string. The format of each parameter definition must be <parameter name>=<value>. The listed parameters can be used in all SQL commands of the ADIF import scheme using the syntax @<parameter name>.

The fields Assembly, Class and Method are not required for the standard export of data via ADIF. They are provided in case an export scenario that cannot be handled via ADIF requires a custom code delivered by Alfabet for the specific export scenario.

- **Import**:
  - **ADIF Import Job**: Enter the name of the ADIF import scheme that shall be executed for data import to Alfabet. To use the standard functionality of the Web Service based ARIS - Alfabet Interoperability Interface , enter ARIS_Import.
  - **Parameters**: This field is not required for the standard import functionality of the Web Service based ARIS - Alfabet Interoperability Interface. It is required if the import is based on a customer-defined ADIF scheme that is configured to perform data import based on parameters given in the command line call for ADIF scheme execution. Define the parameters and their values in a comma-separated string. The format of each parameter definition must be <parameter name>=<value>. The listed parameters can be used in all SQL commands of the ADIF import scheme using the syntax @<parameter name>.

The fields Assembly, Class and Method are not required for the standard export of data via ADIF. They are provided in case an export scenario that cannot be handled via ADIF requires custom code delivered by Alfabet for the specific export scenario.

4) In the toolbar, click the **Save**  button. A browser window opens.
5) Select the name and location for the configuration file and click **Save** to store the file at the defined location.

To edit an existing configuration file, click the **Open**  button in the toolbar and select the configuration file in the browser window that opens.

To clear all fields and start a new configuration, click the **New** button in the toolbar.

Configurations can be tested via the tool A2AIntegrationGui.exe. The tool allows you to configure settings for the export and import and to execute the import or export directly via the interface.

The settings for source and target of the import or export may be stored in the configuration file, for example if the configuration is saved after testing without removing the settings. Nevertheless they are ignored when running the Web Service based ARIS - Alfabet Interoperability Interface. The action performed by the Web Service based ARIS - Alfabet Interoperability Interface is exclusively defined by the command line parameters.

Export can be directed to an XML file to test the export only. Import can be performed from an XML file containing the import data to test import only.

The following table lists the settings required in the A2AIntegrationGui.exe interface in order to perform different tests for import and/or export and the action required to perform the test.

The window below displays the configuration tabs. Error messages are displayed during test execution.

| Test Action to Perform | Required Settings | Starting the Test |
|---|---|---|
| **Export from ARIS to XML file** | In the **ARISAccess Configuration** tab, go to the **Export** section and perform the following settings:<br>• Select the check box **Send Export To File**.<br>• Enter the path to and name of the target XML file in the **Export File** field. You can click the **Browse** button to select a location in the browser window. If the specified file does not exist, it is created during the export process. | Click the **Export** button in the **Export** section of the **ARISAccess Configuration** tab. |
| **Import to Alfabet from XML file** | In the **Alfabet Access Configuration** tab, go to the **Import** section and perform the following settings:<br>• Select the **Import From File** check box.<br>• Enter the path to and name of the source XML file in the **Import File** field. You can click the **Browse** button to select the file in the browser window. | Click the **Import** button in the **Import** section of the **Alfabet Access Configuration** tab. |
| **Export from Alfabet to XML file** | In the **Alfabet Access Configuration** tab, go to the **Export** section and perform the following settings:<br>• Select the **Send Export To File** check box.<br>• Enter the path to and name of the target XML file in the **Export File** field. You | Click the **Export** button in the **Export** section of the **Alfabet Access Configuration** tab. |

| Test Action to Perform | Required Settings | Starting the Test |
|---|---|---|
| | can click the **Browse** button to select a location in the browser window. If the specified file does not exist, it is created during the export process. | |
| **Import to ARIS from XML file** | In the **ARISAccess Configuration** tab, go to the **Import** section and perform the following settings:<br>• Select the **Import from File** check box.<br>• Enter the path to and name of the source XML file in the **Import File** field. You can click the **Browse** button to select the file in the browser window. | Click the **Export** button in the **Export** section of the **ARISAccess Configuration** tab. |
| **Data transfer from ARIS to Alfabet** | In the **ARISAccess Configuration** tab, go to the **Export** section and perform the following settings:<br>• Select the **Send Export to Alfabet** check box.<br>In the **Alfabet Access Configuration** tab, go to the **Import** section and perform the following settings:<br>• Select the **Import fromARIS** check box. | Click the **Export** button in the **Export** section of the **ARISAccess Configuration** tab or the **Import** button in the **Import** section of the **Alfabet Access Configuration** tab. |
| **Data transfer from Alfabet to ARIS** | In the **ARISAccess Configuration** tab, go to the **Import** section and perform the following settings:<br>• Select the **Import from Alfabet** check box.<br>In the **Alfabet Access Configuration** tab, go to the **Export** section and perform the following settings:<br>• Select the **Send Export toARIS** check box. | Click the **Export** button in the **Export** section of the **Alfabet Access Configuration** tab or the **Import** button in the **Import** section of the **ARIS Access Configuration** tab. |

## Running the Web Service Based ARIS - Alfabet Interoperability Interface

The Web Service based ARIS - Alfabet Interoperability Interface can be run via a command line or as part of a job scheduling engine.

| | |
|---|---|
| Executable: | A2AIntegrationBatch.exe located in the **programs** directory of the Alfabet components. |

| | |
|---|---|
| Access of applications via | Web Services on both ARIS and Alfabet side. |
| Preconditions: | • All components of the Web Service based ARIS - Alfabet Interoperability Interface are successfully installed and running and a configuration file is available. For information about installing and configuring the interface, see Basic Initial Set-Up of the Web Service based ARIS - Alfabet Interoperability Interface .<br>• The configuration file is located in the installation directory of the interface.<br>• Only for import of data from an XML file: The XML file containing the data in the XML structure required for import must be located in the installation directory of the interface. |
| Logging | Standard Alfabet logging. For information about standard logging and the command line options, see Command Line Parameters for Logging . |
| Command line help | Start executable with -h or -help. |

The executable must be started with the following parameters:

```
A2AIntegrationBatch.exe -ConfigFile <ConfigurationFile.xml> -ACTION <Action>
```

The table below displays the command line options:

| Command Line Option | Mandatory/ Optional | Explanation |
|---|---|---|
| **-ConfigFile <XMLConfiguration-File.xml>** | Mandatory | Enter the name of the configuration file that stores the connection data required for the action that you want to define. |
| **-ACTION <Action>** | Mandatory | Enter the type of data transfer that you want to perform. Permissible values are:<br>• ARISFILE for data export from ARIS to an XML file.<br>• FILEARIS for data import from am XML file to ARIS.<br>• ARISALFABET for data transfer from ARIS to Alfabet<br>• ALFABETFILE for data export from Alfabet to an XML file.<br>• FILEALFABET for data import from an XML file to Alfabet.<br>• ALFABETARIS for data transfer from Alfabet to ARIS. |

For example to transfer business process model data from ARIS to Alfabet , a configuration file Transfer_ArisAlfabet.xml was created defining the required connection to the ARIS and Alfabet application. To start the data transfer from a command line, open a command line window, for example by starting cmd.exe, navigate to the directory of the ARIS - Alfabet integration interface and start the executable with:

```
A2AIntegrationBatch.exe -ConfigFile Transfer_ArisAlfabet.xml -ACTION ARIS
ALFABET
```

For the standard data transfer of the Web Service based ARIS - Alfabet Interoperability Interface , the following transfer mode is recommended:

- **Transfer of application information fromAlfabettoARIS** :

Application data is subject to constant changes. At the same time, transfer of application data to Alfabet does not require an immediate user interaction via the ARIS user interface. If no changes were performed in Alfabet and data is transferred to ARIS , no changes will be performed in ARIS.

It is recommended to transfer changes to the application landscape to ARIS in regular intervals. Data transfer can be automated for example by scheduling the execution of the batch utility by means of a Windows® batch job.

- **Transfer of business process models fromARIStoAlfabet** :

Business processes are usually more stable than application utilization. At the same time, integration of changes to business process models from ARIS to Alfabet require the manual check in of the resulting solution business process model via the Alfabet user interface.

It is therefore recommended that you transfer changes to the business process model on request (only if changes have actually been made to the business process model in ARIS ).

After data transfer, the following action is required in the interface of the target application:

- Transfer of Application information from Alfabet to ARIS:

Open the group **delete** in the Main Group of your target database and check whether the applications that are in the group are used in any models other than the Application System Type model in the group **delete**. *Application system types* in the group **delete** do not exist any longer in the Alfabet database and shall be deleted in ARIS as well. If an *application system type* in the group **delete** is used in ARIS , substitute it with an *application system type* from the group **active** in the respective models. After correction of modeling, delete the *application system types* in the group **delete**.

- Transfer of business process models from ARIS to Alfabet:

Check the solution business process models for correct assignment of subordinate objects and check in the solution business process models.

## Command Line Parameters for Logging

A standard logging method exists for the Web Service based ARIS - Alfabet Interoperability Interface. Logging information is written to a log file at runtime in the following format:

```
<date and time> <message type> <message text>
```

For example:

```
2009-02-24T12:30:15.4397058Z INFO start data transfer
```

The timestamp is the UTC time (coordinated Universal Time) and may therefore differ from the time in your local time zone. The timestamp is written in ISO 8601 combined date and time format as year-month-dayThour:minutes:secondsZ.

The message type can be one of the following:

- **ERROR**: An error occurred. The message describes the type of error.
- **WARNING**: Problems were encountered during the execution of the utility that are not as severe as an error. The process was executed but the result should be checked. The message describes the problem.
- **INFO**: Information about the normal execution of the utility is provided.

Command line options that can be used when calling the Web Service based ARIS - Alfabet Interoperability Interface utility allow you to specify the number and location of log files as well as the amount of detail of the logging information:

| Command Line Option | Default | Description |
|---|---|---|
| **-logpath <log file path>** | Log file is stored in the working directory of the executable. | Specification of a path to a directory for storage of the log file. |
| **-logfile <log file name>** | <Executable><timestamp>.log | Specification of the log file name. Allowed file extensions are LOG and TXT.<br><br>**NOTE**: Setting of -logfile changes the way log information is stored. If -logfile is not defined, a new log file is created each time the utility is used and the log file name is extended with a timestamp specifying the current UTC time. The timestamp is the UTC time written as "_yyyymmdd_hhmmss". For example: logfilename_20090328_150827.log.<br><br>If -logfile is defined, logging information is appended to the already existing log file each time the utility is used. To restrict the file size, you can set the -logclear option to delete old log messages. |
| **-nologappend** | | This option is only evaluated if -logfile is defined.<br><br>If -nologappend is set, the log file defined with -logfile and -logpath is overwritten with a new file containing the current log information each time the utility is used.<br><br>If -nologappend is not set, logging information is appended to the already existing log file each time the utility is used.<br><br>**NOTE**: To restrict the file size, you can set the -logclear option to delete old log messages. |
| **-logverbose** | | If -logverbose is set, additional information about the running process is logged. The content and amount of additional information messages depend on the utility used. If the setting of -logverbose does not change the |

| Command Line Option | Default | Description |
|---|---|---|
| | | log output, there are no additional information messages available for the utility.<br><br>**NOTE**: Verbose logging is in most cases not required and can lead to a decrease in performance. |
| **-logclear <number of days>** | Infinite | This option can only be used if -logfile is defined and -nologappend is not set.<br><br>During logging, the log file is scanned for log messages with a timestamp older than the number of days specified with -logclear and these messages are deleted.<br><br>**NOTE**: The scanning process can lead to drawbacks in performance. |

## Advanced Set-Up: Customizing Data Transfer from Alfabet to ARIS

The standard data exchanged triggered by the Web Service based ARIS - Alfabet Integration Interface can be adapted by the user to meet company specific requests. For example:

- For both applications and business processes the range of attribute/property values that is transmitted from ARIS to Alfabet and vice versa can be changed to include customer defined or standard data stored about the application or business process.
- Other object types can be added to the transfer of data between the applications.

  For example organizations are both part of ARIS project planning and Alfabet enterprise architecture management. The data for organizations may therefore be included into the data exchange between ARIS and Alfabet.

- The target of the links between the applications can be changed to access the object in the other user interface via a different than the standard view.

  For example a customized object profile has been designed for ICT objects in Alfabet with the name APP_ObjectView_Customized. This object profile includes links to customer configured reports allow to edit and manage some customer defined properties of the ICT object that are not part of the standard Alfabet meta-model. When a user navigates from an *application system type* in ARIS to the corresponding ICT object in Alfabet , the object profile APP_ObjectView_Customized shall be displayed instead of the standard object profile APP_ObjectView.

The data exchange between ARIS and Alfabet is triggered via an XML that is written by one of the applications providing the data and processed by the application to that the data is imported. To change the content of data transmission, the content of the XML must be adapted to the include the new content. This requires changes in both the interface writing the data and the interface processing the data:

- On ARIS side the reports that generate or process the XML data must be changed. The standard reports for data exchange are

- **Alfabet2ARIS- Import application systems**
- **ARIS2Alfabet- Export processes**

The reports are Java based and can be customized in a user interface in the ARIS Architect. To change the scope of data import from or export to ARIS , the standard reports can be altered or completely new reports can be defined. If a new report is defined, the Web Service based ARIS - Alfabet Interoperability Interface must be configured to trigger the customer configured report.

- For information about how to configure ARIS reports, see the ARIS documentation.
- For information about how to configure the ARIS - Alfabet Integration Interface to use any other than the standard report, see Configuration.

- On Alfabet side the ADIF Import and Export schemes that generate and process the XML data must be changed. The standard ADIF schemes for data exchange are:
  - **ARIS _Import**
  - **ARIS _Export**

The ADIF schemes are XML based and can be customized in a user interface in the tool Alfabet Expand . To change the scope of data import from or export to Alfabet , the standard ADIF schemes can be altered or completely new ADIF schemes can be defined. If a new ADIF scheme is defined, the Web Service based ARIS - Alfabet Interoperability Interface must be configured to trigger the customer configured report.

- For information about how to configure ADIF schemes, see the Alfabet documentation (Reference manual *Alfabet Data Integration Framework* ).
- For information about how to configure the ARIS - Alfabet Integration Interface to use any other than the standard report, see Configuration.

This chapter provides you with the special information about the XML structure of the XML for data exchange between Alfabet and ARIS required to perform the correct configuration of both Alfabet ADIF schemes and ARIS reports.



## The XML for Data Transfer From Alfabet to ARIS

The XML has the following structure:

| XML level | Elements | Attributes | Allowed child elements | Stored information |
|---|---|---|---|---|

| XML level | Elements | Attributes | Allowed child elements | Stored information |
|---|---|---|---|---|
| **Root** | Objects | None | Object, Relation | |
| **First Level** | Object | alfa_type<br>alfa_guid<br>aris_type | Attribute, Object | Information about an object of a defined ARIS object type/ Alfabet object class. |
| **Second Level** | Attribute | Name<br>type<br>language<br>aris_type | <text> | Stores the value of a defined ARIS object type attribute / Alfabet object class property for the object. |
| **Second Level** | Object | alfa_type<br>aris_type | Attribute | Stores a link for navigation between the user interfaces of ARIS and Alfabet. |
| **Third Level** | Attribute | Name<br>type<br>language<br>aris_type | <text> | Stores data relevant for the link between the user interfaces of ARIS and Alfabet. |

Each element **Object** represents an object that is exported/imported via the interface. The element **Object** must have the following attributes to store the information about which object is exported and to map the object with existing objects in the target database:

| Attribute | Value | Mandatory/Optional |
|---|---|---|
| **alfa_type** | The identifier for the object class in Alfabet. It is recommended to use the name of the object class in Alfabet to identify the object class. The ADIF export scheme must be configured to write the identifier for the object class into the attribute alfa_type for each object of the class. | Mandatory |
| **alfa_guid** | The GUID of the object in the Alfabet database. In the ARIS database, the alfa_guid values are stored by the attribute type Alfabet - GUID. | Mandatory |
| **aris_type** | The identifier for the object type in ARIS. It is recommended to use the API name of the object type to identify the object type. The mapping of the object with a defined identifier to the correct object type is performed | Mandatory |

| Attribute | Value | Mandatory/Optional |
|---|---|---|
| | via the ARIS import report. | |

Attributes of the object class properties in Alfabet are mapped to attribute types in ARIS. Mapping is done by adding one element **Attribute** as child element to the **Object** element for each ARIS attribute / Alfabet property mapping that shall be performed. The **Attribute** element has the following attributes:

| Attribute | Value | Mandatory/Optional |
|---|---|---|
| **Name** | The identifier for the object class property in Alfabet. It is recommended to use the name of the object class property in Alfabet to identify the object class property. The ADIF export scheme must be configured to write the identifier for the object class property into the attribute Name for each Attribute element exporting data for the object class property. | Mandatory |
| **aris_type** | The identifier for the attribute type in ARIS. It is recommended to use the API name of the attribute type to identify the attribute. The mapping of the attribute with a defined identifier to the correct attribute type is performed via the ARIS import report. | Mandatory |
| **type** | The data type of the value. Permissible data types are: Boolean, Date, DateTime, Integer, Real, String, Text, Time, URL. | Mandatory |
| **language** | The language culture name of the language the value is defined for. The language culture name must be written in lower case letters with a hyphen between language and country information (for example en-us, de-de) | Mandatory |

The information about the value of the attribute is stored as text within the **Attribute** element.

### Links from ARIS to Alfabet

Multiple links to the Alfabet user interface can be defined per object. For each link, an element **Object** must be added as child element to the **Object** element defining an imported object. The **Object** element must be defined with the following attribute settings that shall not be altered:

| Attribute | Value | Mandatory/Optional |
|---|---|---|
| **alfa_type** | A2ALink | Mandatory |
| **aris_type** | AT_LINK_INFO | Mandatory |

The Object element must have two child elements **Attribute** to define a single link. The **Attribute** elements define the URL to the Alfabet user interface and the title of the link displayed on the ARIS user interface. URL and title are defined as text within the **Attribute** elements while the setting of the attributes of the **Attribute** element specify whether this element stores a title or a URL.

The following table lists the required settings for the attributes of the **Attribute** elements defining a link:

| Attribute | Value for Title Definition | Value for Link Definition | Remark |
|---|---|---|---|
| **Name** | `Name` | `Link` | |
| **type** | `string` | `url` | |
| **language** | locale id of culture | locale id (LCID as string) of culture | The ARIS user interface is available for multiple language cultures. The locale ID (LCID) defines for which language the link definition in the current element is used. |
| **aris_type** | AT_EXT_<number> | `AT_EXT_<number>` | Multiple links to the Alfabet user interface can be defined for the same object. Each link defined for an object must have a different number. Please note that the same number must be used for all Attribute elements within an Object element. This includes different language versions of a link. |
| **<text> (content of element)** | The title string to be displayed | URL definition | The required syntax for the URL is described in the section Required Syntax for the URL Linking to the Alfabet User Interface. |

### Required Syntax for the URL Linking to the Alfabet User Interface

The link for access to the Alfabet user interface must have the following syntax:

**Alfabet _URL**`/ExternalAccess.aspx?AccessType=ExternalAccess&UserType=Name&Object=`**REFSTROfObject**`&View=ObjectView:`**ObjectViewName**

with the following variables depending on the Alfabet installation and the object to be accessed:

- **Alfabet_URL**: The URL for access to the Alfabet user interface.
- **REFSTROfObject**: The value of the REFSTR property of the object in the Alfabet database. Within an Alfabet database, the REFSTR is a unique property that allows the object to be identified unambiguously.
- **ObjectViewName**: The name of the object profile that shall open when the user clicks the link. The object profile must be defined for the object class of the object for that the URL is defined.

An object profile in Alfabet corresponds to the fact sheet of an object in ARIS. Object profiles can be created and/or customized. For more information about object profiles in the Alfabet meta-model and the customization of the object profiles, see the reference manual *Configuring Alfabet with Alfabet Expand* of the Alfabet documentation.

#### Defining Links for Display in Multiple Languages

When the Alfabet user interface is displayed in multiple different languages, the two **Attribute** elements defining the link must be repeated for each language. The settings are the same except for the language attribute that must define the locale id the definition is valid for and the text content of the title definition that must include the translation into the defined language.

## The XML for Data Transfer From ARIS to Alfabet

The XML has the following structure:

| XML level | Elements | Attributes | Allowed child elements | Stored information |
|---|---|---|---|---|
| **Root** | Objects | None | Object, Relation | |
| **First Level** | Object | alfa_guid<br>aris_type<br>aris_guid | Attribute, Object | Information about an object of a defined ARIS object type/ Alfabet object class. |
| **First Level** | Relation | from_guid<br>to_guid<br>alfa_name<br>aris_type | <text> | Information about a relationship with another object. |
| **Second Level** | Attribute | Name<br>type<br>language<br>aris_type | <text> | Stores the value of a defined ARIS object type attribute / Alfabet object class property for the object. |
| **Second Level** | Object | alfa_type<br>aris_type | Attribute | Stores a link for navigation between the user interfaces of ARIS and Alfabet. |

| XML level | Elements | Attributes | Allowed child elements | Stored information |
|---|---|---|---|---|
| **Third Level** | Attribute | Name<br>type<br>language<br>aris_type | <text> | Stores data relevant for the link between the user interfaces of ARIS and Alfabet. |

Each element **Object** represents an object that is exported/imported via the interface. The element **Object** must have the following attributes to store the information about which object is exported and to map the object with existing objects in the target database:

| Attribute | Value | Mandatory/Optional |
|---|---|---|
| **alfa_type** | The identifier for the object class in Alfabet. It is recommended to use the name of the object class in Alfabet to identify the object class. The ADIF import scheme must be configured to write the identifier into the attribute alfa_type for each object. | Mandatory |
| **aris_guid** | The GUID of the object in the ARIS database.<br>The Alfabet attribute ARIS _Guid stores the information about the aris_guid. | Mandatory |
| **aris_type** | The identifier for the object type in ARIS. It is recommended to use the API name of the object type to identify the object type. The mapping of the object with a defined identifier to the correct object type is performed via the ARIS import report. | Mandatory |

Attributes of the object type in ARIS are mapped to object class properties in Alfabet and vice versa. Mapping is done by adding one element Attribute as child element to the **Object** element for each ARIS attribute / Alfabet property mapping that shall be performed. The **Attribute** element has the following attributes:

| Attribute | Value | Mandatory/Optional |
|---|---|---|
| **Name** | The identifier for the object class property in Alfabet. It is recommended to use the name of the object class property in Alfabet to identify the object class property. The mapping of the attribute with a defined identifier to the correct object class property is performed via the ADIF import scheme. | Mandatory |
| **aris_type** | The identifier for the attribute type in ARIS. It is recommended to use the API name of the attribute type to identify the attribute. The ARIS export report must be configured to add the correct identifier to the XML for defined Attribute Types. | Mandatory |

| Attribute | Value | Mandatory/Optional |
|---|---|---|
| **type** | The data type of the value. Permissible data types are: Boolean, Date, DateTime, Integer, Real, String, Text, Time, URL. | Mandatory |
| **language** | The language culture name of the language the value is defined for. The language culture name must be written in lower case letters with a hyphen between language and country information (for example en-us, de-de) | Mandatory |

The information about the value of the attribute is stored as text within the **Attribute** element.

### Links from Alfabet to ARIS

Multiple links to the Alfabet user interface can be defined per object. For each link, an element **Object** must be added as child element to the **Object** element defining an imported object. The **Object** element must be defined with the following attribute settings that shall not be altered:

| Attribute | Value | Mandatory/Optional |
|---|---|---|
| **alfa_type** | AT_LINK_INFO | Mandatory |
| **aris_type** | LINK_ARCHITECT or LINK_CONNECT | Mandatory |

The Object element must have two child elements **Attribute** to define a single link. The **Attribute** elements define the URL to the Alfabet user interface and the title of the link displayed on the ARIS user interface. URL and title are defined as text within the **Attribute** elements while the setting of the attributes of the **Attribute** element specify whether this element stores a title or a URL.

The following table lists the required settings for the attributes of the **Attribute** elements defining a link:

| Attribute | Value for Title Definition | Value for Link Definition | Remark |
|---|---|---|---|
| **Name** | `Name` | `Link` | |
| **type** | `string` | `url` | |
| **language** | locale id of culture | locale id (LCID as string) of culture | The ARIS user interface is available for multiple language cultures. The locale ID (LCID) defines for which language the link definition in the current element is used. |

| Attribute | Value for Title Definition | Value for Link Definition | Remark |
|---|---|---|---|
| **aris_type** | AT_EXT_<number> | AT_EXT_<number> | Multiple links to the ARIS user interface can be defined for the same object. Each link defined for an object must have a different number. Please note that the same number must be used for all Attribute elements within an Object element. This includes different language versions of a link. |
| **<text> (content of element)** | The title string to be displayed | URL definition | The required syntax for the URL is described in the section Required Syntax for the URL Linking to the Alfabet User Interface. |

### Required Syntax for the URL Linking to the ARIS User Interface

The link for access to the ARIS user interface must have the following syntax:

**ARIS _URL**/#default/item/**objectidentifyer**

with the following variables depending on the Alfabet installation and the object to be accessed:

- **ARIS _URL**: The URL for access to the ARIS user interface.
- **objectidentifyer**: link target definition as defined in ARIS export report.

### Defining Links for Display in Multiple Languages

When the ARIS user interface is displayed in multiple different languages, the two **Attribute** elements defining the link must be repeated for each language. The settings are the same except for the language attribute that must define the locale id the definition is valid for and the text content of the title definition that must include the translation into the defined language..

# Index