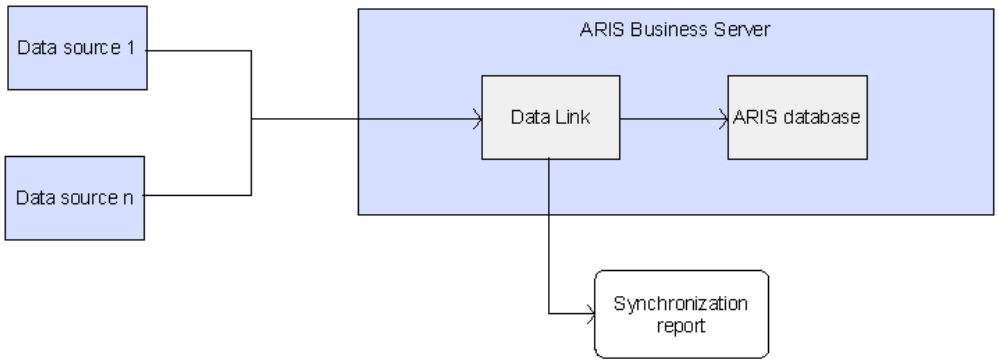
DataLink documentation

Content

1 INTRODUCTION

DataLink is customized solution for one way synchronization of various data to ARIS. This document describes how DataLink works and how can be configured.

DataLink reads data from external data sources and synchronizes ARIS database (objects, connections, models, attributes) with these changes. It also formats ARIS models with selected logic. Synchronization report with change log is created as part of synchronization.



DataLink is licensed by method areas, which are defined by supported ARIS model types along with its objects and connections. For example the HR method area is defined by Organization chart model and all objects and connections allowed for this model type.

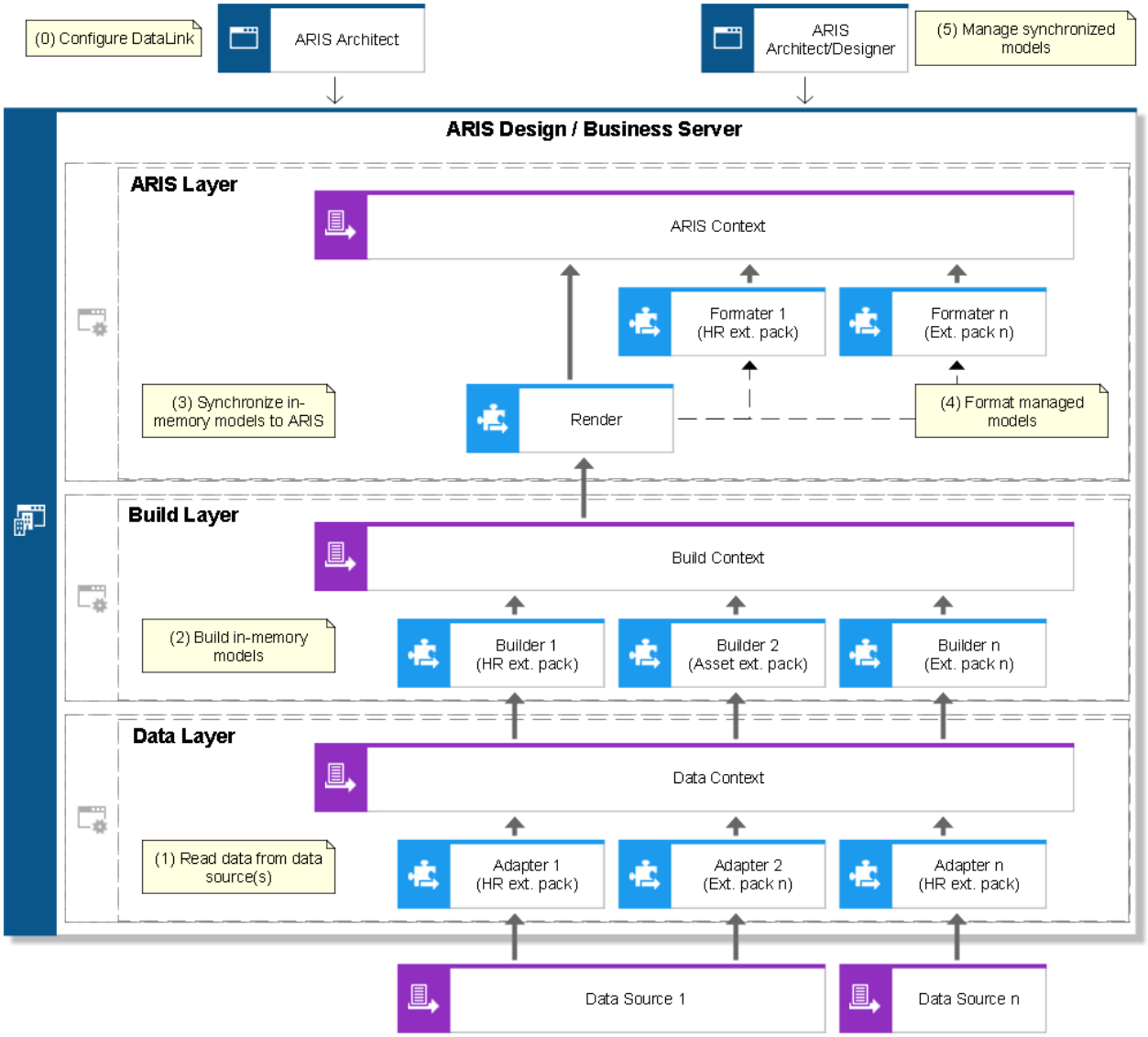
Support for both various data sources and method areas is already available through extension packs and can be developed for others.

For latest information on licensing, quotes and method areas please contact info@arisdatalink.com

2 ARCHITECTURE

DataLink runs in ARIS Business Server process and is started as an ARIS report.

DataLink architecture is organized into 3 layers, which are responsible for different activities described in the following subsections.



2.1 Data layer

Data layer is responsible for:

- Loading data from possibly multiple data sources.

Data is typically read by multiple Adapters and then stored in Data Context. The Data Context holds data in form of data objects, data connections and attributes.

Both data objects and data connections must have specified unique id and type. The data layer types will be mapped to ARIS types (number, or GUID) and symbols in the build layer.

Attributes can be assigned to the data objects and data connections via referencing their unique id. Attributes in data layer are language dependent.

Object and connection id uniqueness is enforced in the data layer, but consistency of the relations is not checked in the data layer - e.g. attributes can be assigned to non-existing object and a connection can be created between non-existing objects. Consistency is enforced by the build layer.

2.2 Build layer

Build layer has 2 main purposes:

- Map data layer types to ARIS types and enforce consistency
- Create in-memory build model

Typically multiple Builders are used to build in-memory model from the Data Context structure. The in-memory model specifies ARIS like structures on definition level:

- Group
- Model
- Object definition
- Connection definition
- Attribute

The following structures are specified on the occurrence level:

- Object occurrence
- Connection occurrence

Occurrence level of build layer is highly simplified compared to ARIS layer (occurrence positioning, sizing, attribute placements, colors, rounding, graphics objects, etc. are not included). This information can be added later in the ARIS layer.

Build layer is also responsible for mapping item type and attribute type to ARIS type (either number, or GUID).

Consistency of in-memory model is enforced by in the build layer.

DataLink build layer types are based on available ARIS types. Objects and connections require a type number. Symbols, models and attributes support both type numbers and type GUIDs of custom types.

	Type number	GUID
Object type	Yes	No
Connection type	Yes	No
Symbol	Yes	Yes
Model type	Yes	Yes
Attribute type	Yes	Yes

2.3 ARIS layer

ARIS layer has 2 purposes:

- Synchronize build layer in-memory model to ARIS database
- Format ARIS models

The synchronization of in-memory model to ARIS database is done typically by a single Renderer. Synchronization is described in detail in next section.

Formatting of individual models is done by possibly multiple Formatters. Formatters are assigned to models in the build layer.

The communication between DataLink and ARIS Business Server is realized via ARIS Context reporting object.

3 DEFAULT SYNCHRONIZATION

DataLink synchronization transfers build layer in-memory model to ARIS and performs clean up of obsolete ARIS items. During this process items and occurrences can be created, updated and/or made obsolete.

DataLink synchronization works with complete data from the data sources and synchronizes it with content of ARIS database. The synchronization can be configured for different results. Following sections describe the options.

3.1 Item synchronization

The following are Items:

- Group
- Object definition
- Connection definition
- Model

DataLink synchronization depends on the following attributes, which must be maintained by items for the synchronization to work:

- DataLink ID - Holds unique identifier across all DataLink items.
- DataLink Status - Holds status of the item synchronization. Following status values are available:
  - New - Item was created in the last synchronization.
  - Existing - Item already exists in the database and was not updated in the last synchronization.
  - Updated - Item was updated in the last synchronization.
  - Obsolete - Item does not exist in the build layer, typically because it is obsolete in the data source.

Items with maintained DataLink ID attribute and managed by DataLink. Items without maintained DataLink ID attribute are not affected by DataLink synchronization.

DataLink also synchronizes following occurrences:

- Object occurrence
- Connection occurrence

Occurrences are synchronized as part of model synchronization.

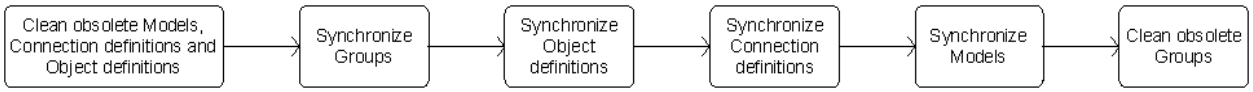
3.2 Attribute synchronization

Item attributes can be synchronized from build layer to ARIS. Following scenarios are possible:

- All attributes from build layer are synchronized. In this scenario, attributes not present in the build layer, but maintained in ARIS will be deleted.
- Selected attributes from build layer are synchronized. In this scenario it can be configured at item type, which attributes will be synchronized.
- No attributes from build layer are synchronized. Attributes from the build layer are not transferred to ARIS (except the DataLink attributes - id, status and optionally history).
- No attributes from build layer are updated. Attributes from the build layer will be added to new objects, but existing objects will stay untouched (except the DataLink attributes - id, status and optionally history).

3.3 Synchronization steps

The synchronization works in the following steps:



3.3.1 Clean obsolete Models, Connection definitions and Object definitions

Check if the database contains DataLink items, which are not present in the build layer. Such items are obsolete and can either be marked as obsolete (DataLink Status attribute), or removed from the database.

This step also optionally removes items marked as obsolete by the previous synchronization.

3.3.2 Synchronize Groups

Create new, or update existing Groups in ARIS.

This step also involves Group attributes synchronization.

3.3.3 Synchronize Object definitions

Create new, or update existing Object definitions in ARIS. Object definition group is updated if needed.

This step also involves Object definition attributes synchronization.

3.3.4 Synchronize Connection definitions

Create new, or update existing Connection definitions in ARIS.

Source and target objects of a connection are not updated. If the source or target objects change, new Connection definition with new DataLink ID should be created.

This step also involves Connection definition attributes synchronization.

3.3.5 Synchronize Models

Create new, or update existing Models in ARIS. Model group is updated if needed.

For models existing in ARIS, DataLink object and connection occurrences are synchronized with the build layer. Non-DataLink occurrences present in the Model are optionally removed.

This step also involves Model attributes synchronization.

As part of this step Object Definition and Connection definitions assignments to the Model are recreated and Formatters are executed on the Model.

3.3.6 Clean empty Groups

Removes empty DataLink groups from ARIS database.

4 REPORTS

DataLink optionally produces report from the synchronization process.

4.1 Synchronization report

Synchronization report contains DataLink synchronization messages.

The following messages are reported:

- New items created
- Existing items updated including attribute synchronization
- Obsolete items removed and items made obsolete
- Synchronization issues

Synchronization report output can be produced in custom format (Word, Excel, XML, plain text). Standard format is Word and plain text.

4.2 DataLink logging

DataLink uses Java logging, which can be configured in ARIS standard config/logging.cfg configuration file.

The log file contains information about DataLink processing and issues. After DataLink implementation is finished it is recommended to set DataLink logging on warning level.

5 CONFIGURATION

DataLink can be configured to different scenarios through pre-configured extension packs based on chosen use case (e.g. Organizational Analysis, IT Asset Management,..). The configuration is stored in xml files on the server.

Following section describes configuration options. Configuration used for this project is attached as appendix including the XML schema definition.

5.1 Main configuration

Main configuration section contains configuration relevant for all DataLink parts:

- Configuration name - description of the configuration
- Configuration default flag - indicates configuration that is loaded by default
- DataLink license key
- ARIS language codes DataLink will synchronize into
- ARIS attribute types, that will be used as DataLink ID and DataLink Status
- Logging settings for DataLink layers

5.2 Mapping

Mapping section specifies mapping from data source types to ARIS types and symbols (models, object definitions, connection definitions, attributes).

5.3 Adapters

Adapters section specifies configuration of Adapters that are used to read source data.

5.4 Builders

Builders section specifies configuration of Builders that are used to create in-memory object model from the source data. Builders create Groups, Models, Object definitions and occurrences, Connection definitions and occurrences and Attributes. Builders can also assign Formatters to Models.

5.5 Renderers

Renderers synchronize the in-memory object model to ARIS database.

Default Renderer corresponds to synchronization steps described in previous section and has the following synchronization options:

5.5.1 Attribute synchronization

- syncGroupAttrs - Synchronize group attributes
- syncObjDefAttrs - Synchronize object definition attributes
- syncCxnDefAttrs - Synchronize connection definition attributes
- syncModelAttrs - Synchronize model attributes

5.5.2 Item synchronization

Removing items in the first run of DataLink:

- removeObjDefs - Remove previously non-obsolete object definitions when found obsolete (true), or set status to obsolete (false)

- removeCxnDefs - Remove previously non-obsolete connection definitions when found obsolete (true), or set status to obsolete (false)
- removeModels - Remove previously non-obsolete models when found obsolete (true), or set status to obsolete (false)
- removeEmptyGroups - Remove empty DataLink groups under the selected root group

Removing obsolete items in the second run of DataLink:

- removeObsoleteObjDefs - Remove obsolete object definitions from database
- removeObsoleteCxnDefs - Remove obsolete connection definitions from database
- removeObsoleteModels - Remove obsolete models from database

5.5.3 Occurrence synchronization

Removing occurrences of DataLink items:

- removeObsoleteObjOccs - Remove obsolete object occurrences from DataLink models
- removeObsoleteCxnOccs - Remove obsolete connection occurrences from DataLink models

Removing occurrences of non-DataLink items:

- removeNonDataLinkObjOccs - Remove object occurrences without DataLink Id from DataLink models>
- removeNonDataLinkCxnOccs - Remove connection occurrences without DataLink Id from DataLink models

5.6 Formatters

Formatters format ARIS models with custom look and feel.

In general Formatters can affect the following:

- Object size, position and color
- Connection points and style
- Placed attributes
- Free text and graphics
- Applied template

5.7 Extension Packs

Extension pack is pre-prepared configuration set suitable for specific data domain. The following Extension Packs are available:

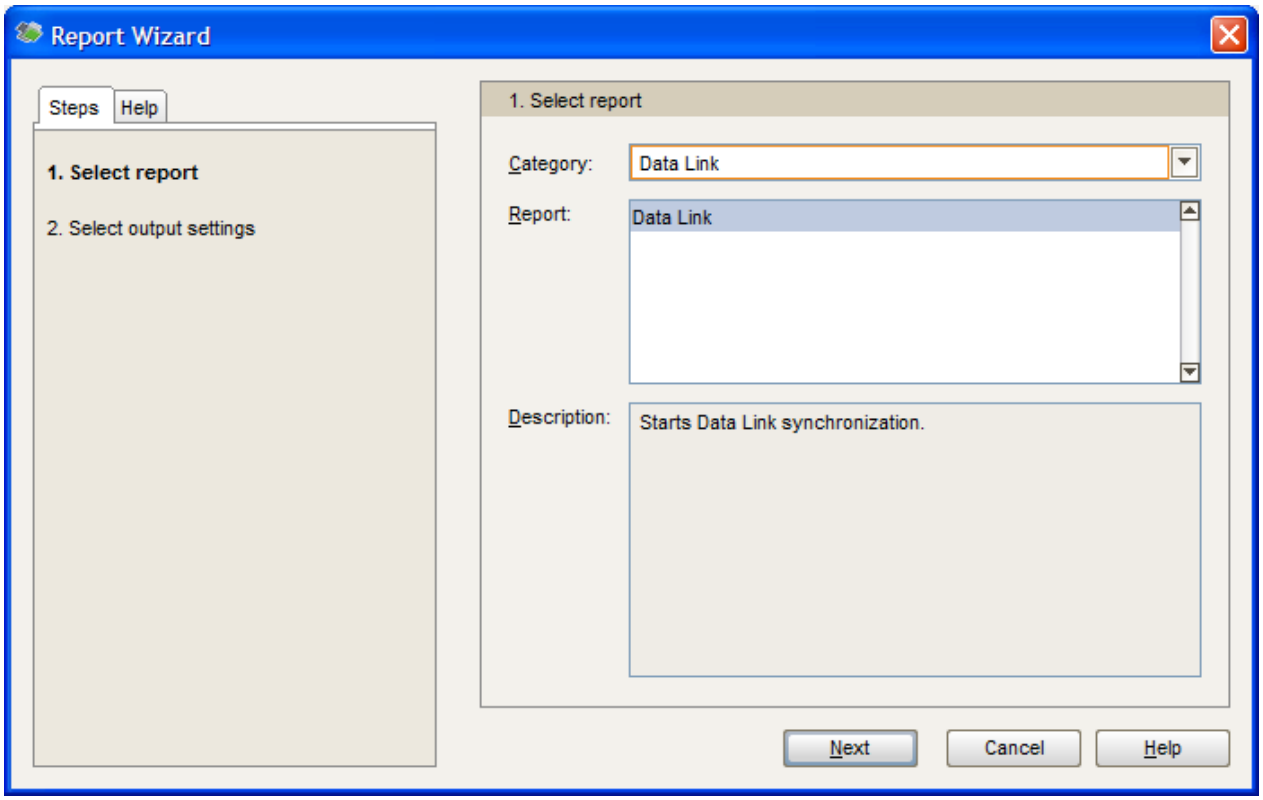
Extension Pack	Description	Supported use cases	Supported source systems
HR	HR data importing	Organizational Analysis	SAP HR
EAM	- IT assets importing - HR data importing	- IT Assets Management - Identity Management - Enterprise Architecture Management	Tivoli Change and Configuration Management Database /*
EA Integration	Integration to Enterprise Architect	Model-To-Execute	Enterprise Architect by Sparx /*
SDK	API, documentation and examples for custom development	Custom development	Eclipse

/* Planned for upcoming release that is going to be launched by the end of September 2014.

6 RUNNING DATALINK

6.1 DataLink synchronization

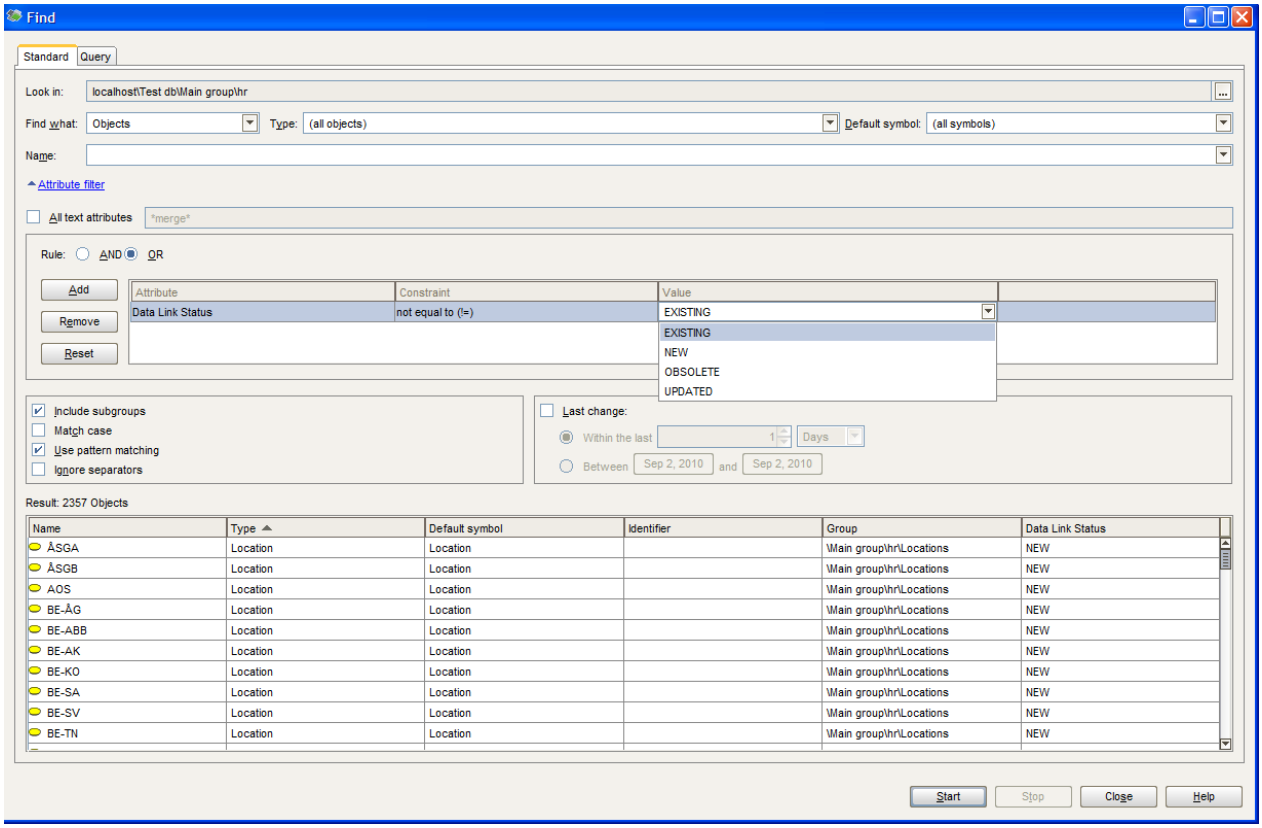
DataLink is started as ARIS report in Group context, which will be the root group for DataLink synchronization.



After synchronization is finished, successful of failed message is displayed.

It is also possible to start the report in silent mode, when no GUI is displayed. The silent mode can also be scheduled using ARIS report scheduling or standard ARIS script runner interface.

Duration and memory requirements of DataLink synchronization depends on the amount of data that needs to be synchronized and non-DataLink content in the database. In typical scenarios the synchronization runs for a few minutes, but in huge scenarios it can run for several hours.



DataLink synchronized objects can be also identified with Find by DataLink ID, or DataLink Status attributes.

TROUBLESHOOTING

DataLink logs its activity to datalink.log file in ARIS server log directory.

If further support is needed, please contact support at support@arisdatalink.com.