DRAFT

SimLink User Guide

Prepared for

Internal

April, 2015



Office address  
Office address  
Office address

Contents

[Contents iii](#_Toc416031449)

[Acronyms and Abbreviations v](#_Toc416031450)

[SimLink Overview 1-1](#_Toc416031451)

[1.1 SimLink Concept 1-1](#_Toc416031452)

[1.2 App Directory 1-1](#_Toc416031453)

[1.3 Datasource 1-1](#_Toc416031454)

[1.3.1 Choosing an Alternate Data Source 1-1](#_Toc416031455)

[Scenario Management 2-1](#_Toc416031456)

[2.1 Defining a Project 2-1](#_Toc416031457)

[2.2 Defining an Evaluation Group 2-1](#_Toc416031458)

[2.2.1 Role of Evaluation Group 2-1](#_Toc416031459)

[2.2.2 SQL 2-1](#_Toc416031460)

[2.2.3 Reference Evaluation Group 2-1](#_Toc416031461)

[2.2.4 Defining the Reference EG 2-2](#_Toc416031462)

[2.2.5 Splints 2-2](#_Toc416031463)

[2.3 Scenarios 2-2](#_Toc416031464)

[2.4 File Management 2-2](#_Toc416031465)

[2.5 Project Level Supporting Data 2-2](#_Toc416031466)

[2.5.1 Option List 2-2](#_Toc416031467)

[2.5.2 Element List 2-3](#_Toc416031468)

[2.5.3 User-Defined Functions 2-4](#_Toc416031469)

[2.5.4 Constants 2-4](#_Toc416031470)

[Simulation Platform Wrappers 3-5](#_Toc416031471)

[3.1 Network Topology 3-5](#_Toc416031472)

[3.1.1 VarType\_FK 3-5](#_Toc416031473)

[Formulation 4-6](#_Toc416031474)

[4.1 Decision Variables 4-6](#_Toc416031475)

[4.1.1 Purpose 4-6](#_Toc416031476)

[4.1.2 SQL 4-6](#_Toc416031477)

[4.1.3 Key Concepts 4-6](#_Toc416031478)

[4.1.4 Grid Usability 4-6](#_Toc416031479)

[4.1.5 Importing Decision Variables 4-7](#_Toc416031480)

[4.2 Results (Summary) 4-7](#_Toc416031481)

[4.2.1 Purpose 4-7](#_Toc416031482)

[4.2.2 Import 4-7](#_Toc416031483)

[4.3 Results (Timeseries) 4-7](#_Toc416031484)

[4.4 Results (Events) 4-7](#_Toc416031485)

[4.4.1 Purpose 4-7](#_Toc416031486)

[4.4.2 SQL 4-7](#_Toc416031487)

[4.4.3 Fields to Include 4-7](#_Toc416031488)

[4.4.4 Additional Notes 4-8](#_Toc416031489)

[4.5 Performance Variables 4-8](#_Toc416031490)

[4.5.1 Purpose 4-8](#_Toc416031491)

[4.5.2 SQL 4-8](#_Toc416031492)

[4.5.3 General Guidance 4-8](#_Toc416031493)

[4.5.4 Specific Guidance 4-10](#_Toc416031494)

[Running SimLink 5-11](#_Toc416031496)

[5.1 User Defined Simulations 5-11](#_Toc416031497)

[5.1.1 Staging SimLink Development for Scenario Life Cycle Tracking 5-11](#_Toc416031498)

[5.1.2 User-Defined Scenarios Grid 5-11](#_Toc416031499)

[5.1.3 Importing Scenarios 5-11](#_Toc416031500)

[5.1.4 Running User –Defined Runs 5-12](#_Toc416031501)

[5.2 Optimization 5-12](#_Toc416031502)

[5.2.1 Purpose 5-12](#_Toc416031503)

[Special Cases 5-15](#_Toc416031504)

[5.3 Cross-Model (XModel) Linkages 5-15](#_Toc416031505)

[5.4 Splints 5-15](#_Toc416031506)

Acronyms and Abbreviations

Section 1

# SimLink Overview

## SimLink Concept

TODO: Met to fill in the SimLink concept

Reference figure

### Philosophy

Met to fill in

### Status and Roadmap

Met to fill in

## Supported Platforms

## Defining a Data Source

SimLink can use either SQL Server or Microsoft Access as a database back-end. Once installed and configured, you will not generally need to change the data source. However, more advanced users, or users supporting multiple different projects, may wish or need to use different data sources for different purposes. To link to a data source, follow these steps:

# Creating a Project

## Defining a Project

A project contains all relevant information for a specific SimLink evaluation. A user may define multiple projects within a specific SimLink database. Data is shared across a project, and generally accessible to more specific evaluation groups (see Section XXX) within that project. Key aspects of a project:

* References a specific simulation platform
* May contain numerous Evaluation Groups

To create a project:

<<erin: please briefly add the steps for creating the project

### Importing Network Topology

The user may wish to import a file associated with the project that defines the network topology. This can be very useful, because then all network data can be referenced explicitly in functions, cost tables, performance criteria etc. However, SimLink is also intended to be able to be used against a baseline model file without the database network representation of the model; this uses less explicit references (e.g. element names, rather than unique IDs), which is imperfect but almost always sufficient.

Things to know about network topology:

* It is optional for the user to import the network topology. It’s generally a good idea to use this feature, particularly if you need to reference network data (like feature length, size, material etc.) in cost or performance functions
* Network definition may vary by model version; if you are using a more recent version of a network, you may need to coordinate with SimLink development team to support the latest version (whether or not your choose to import the network)
* Importing can be error prone. Please communicate errors and issues to the SimLink team. This is the only way to improve this functionality.

Generally, you should import your network at the creation of a project. Here’s how:

## Defining an Evaluation Group

### Role of Evaluation Group

An evaluation group (EG) is a container for a particular investigation, or evaluation of a “what-if” question. A project may have just one EG, or several. Distinct EG are often used when a different baseline model is needed for the same formulation, or when the goal of the analysis is distinctly different than previous analyses. If supporting data, like element lists or option lists, cost functions is used by multiple different evaluations, it is better to use a single project with multiple EG, rather than multiple projects. An evaluation group (along with a baseline scenario) is automatically created when a project is created. Key concepts for the evaluation group include:

* Baseline model: Different EG will often use a different baseline model. This is defined
* Reference Evaluation Group (see below): An EG can use the formulation information from a different, reference EG, thus avoiding the need to duplicate formulation setup time.

To create an evaluation group:

### Reference Evaluation Group

An EG may be linked to a reference EE; this means that the EG uses the formulation data (e.g. decision variables, results and performance definition) of a different EG. This is a very useful feature for saving time, and avoiding to have to make changes in multiple EG. When an EG is developed, the user has the ability to define a reference EG.

## Scenarios

If an optimization is performed, scenarios will be generated automatically. In many cases, a user may wish to define specific scenarios.

## Project Level Supporting Data

These data are stored at project level, and always retrieved or inserted for the active project.

### Option List

Option lists are used to provide pre-defined lists of potential values to SimLink.

Example: Providing pre-defined diameter options to a decision variable

Ribbon Location: Home\\ Supporting Data \\ Option List

### Element List

Element lists are lists of model elements of the same type (eg pipe, weir, pump). They enable a decision variable or results to reference numerous model elements with a single decision variable.

The ‘type’ of an element list is NOT defined for the element list- although there is a type field, this is really just for the user. The TYPE is defined by the VarType\_FK on a DV.

### User-Defined Functions

User-defined functions can be added for the calculation of DV or performance values (as opposed to a lookup list).

OOPS

I forgot to add Category. Please add a small edit box for Category beneath label.

#### Function Editor (Future functionality)

Creating a viable function requires the user to follow a special syntax; this will be hard for most users to create. In the future, a function editor would allow users to create functions that reference particular SimLink data types in a more intuitive manner.

NOT FOR CURRENT IMPLEMENTATION

### Constants

The user may add constants for referencing in functions. These are tied to a PROJECT, except for pre-loaded constants which are tied to ProjID\_F=-1 (This is not relevant for this task- but is relevant when loading to a drop down list).

# Formulation

## Overview

The bulk of setup time needed to use SimLink is spent on the formulation- telling SimLink to understand what the user wants it to do- e.g. what variables to change in the model, what result to pull, how to cost alternatives, and how to formulation simulation information into performance factors.

The following sections define how the user can setup a formulation within the SimLink interface. Because many setup tasks may require repetitive operations, in many cases a CSV import feature is more efficient for setup.

Finally, SimLink includes numerous functionalities which are not yet supported within the interface. The must be defined “behind the scenes”. Advanced users may gain an understanding of the backend database, and be able to formulate their problem more efficiently “behind the scenes”, that is, by directly populating the database. As development continues, more advanced functionality will be integrated into the user interface.

## Decision Variables

### Purpose

Decision variables allow the user to define either functions or look up list for creating new model parameters based upon a value (generally from an algorithm). Decision variables tie closely to the particular framework being modeled; they require a fine degree of granularity to enable detailed model changes for potentially large numbers of model elements.

### Key Concepts

* Primary vs Secondary: Secondary decision variables ALWAYS reference a primary variable, and use the same DNA bit to create changes. Secondary decision variables enable consequential/contingent changes
* Option Lists: Referenced list of pre-calculated values that are used (lookup table)
* Functions- Functions may calculate the decision variable based upon the DNA bit- they may ALSO use an option list
* Element List: a list of model elements for which model changes are generated. Decision variables are always linked to an element list (even if only one model element is changed, it must be in a list – though in the future this may be changed)

### Importing Decision Variables

The user may import DV using the import function.

* have the function info in there if irrelevant

## Results (Summary)

### Purpose

### Import

Similar to 4.1.5: see ‘result.csv’

* I changed the field names; you should just require them to be in this order

## Results (Timeseries)

## Results (Events)

### Purpose

Events are processes performed on timeseries data. The user requests a type of sweep to be performed on the timeseries data, according to parameters defined in the event. The SimLink engine will then run through the TS data, and may find zero, one, or many events. They are thus very closely linked to TS data. Presently events do not have their own label, but probably one will be addd.

### Additional Notes

The event is tied to a tblResultTS record. When editing a record the user should be able to select an alternate TS record from that EG (or ref EG, if applicable). Therefore the ResultTS\_or\_Event\_ID\_FK should be populated with the key/label vals of tblResultTS for active EG.

## Performance Variables

### Purpose

Performance variables are metrics of performance of components or the system as a whole. They may be as simple as simply repeating a result (e.g. no transformation or post-processing is required on a performance variable), or quite complicated, involving reference to the numerous predecessor tables. Examples include:

* Summing up result totals from numerous locations
* Calculating the difference between baseline and alternative conditions (any result)
* Cost
* Cost effectiveness
* Objective functions: single objective aggregation of numerous competing objectives

# Running SimLink

## User Defined Simulations

User-defined simulations are a useful way for users to selectively explore the decision space, useful for: testing, running specific solutions of interest, defining seed solutions for the optimization. SimLink is geared towards a flexible, customizable interaction between the user and simulation automation- meaning that the user may wish to be able to control characteristics of the simulation execution. For instance, some simulations may execute over long periods of time after SimLink is closed; the user must be able to finalize results processing and performance definition after a period of execution.

### Staging SimLink Development for Scenario Life Cycle Tracking

* Stage 1: User manages tracking ‘stage’ of life cycle in between discontinuous sessions (SimLink manages during a process)
* Stage 2: SimLink syncs scenario status in between discontinuous sessions.

UI support for Stage 1 will be a big improvement and is the current priority.

#### Scenario Life-Cycle

Simlink offers the user fine control over how a scenario is executed. Stages of a scenario ‘life cycle’ are coded as numbers. For now, the user will need to define these numbers.

*nScenLCExist = 1;*

*nScenLCModElementExist = 2;*

*nScenLCBaselineFileSetup = 3;*

*nScenLCBaselineModified = 4;*

*nScenLCModelExecuted = 5;*

*nScenLCModelResultsRead = 6;*

*nScenLModelResultsTS\_Read = 10;*

*nScenResultTS\_Operations = 12;*

*nScenDefineEvents = 15;*

*nScenEventAggregation = 15;*

*nScenLCSecondaryProcessing = 20;*

### Importing Scenarios

It is too laborious to enter the number of scenarios manually (though user should have option, similar to DV grid)- there may be dozens or hundreds.

Therefore, a simple import is required for the user to import scenarios. Please see accompanying CVS ‘scenario.csv’ for the format to be supported. A record should be inserted for row in the .csv. Similar to the DV import (4.1.5), an option to export a template for the user would be nice.

### Running User –Defined Runs

#### Run all – please add to Home\Scenario\Run scenarios (menu item)- please add an empty handler, which will call a simlink function. Note: this only runs scenarios where HasBeenRun = false.

* + **Run single**: Context menu on grid row, ‘Run Scenario’ which runs that individual scenario (again - please add an empty handler. The menu item should only be enabled if ‘HasBeenRun’ = false

At present, these commands will launch new processes, and the user will need to collect results later.

## Optimization

### Purpose

# Special Cases

## Cross-Model (XModel) Linkages

## Splints