

LABNAF CHANGE HISTORY

B0204

REPOSITORIES

- Sample repository
 - End to end navigable example (BOPCO: The Bus Operating Company) including
 - [dashboards](#) and especially [solution architecture completion dashboards](#).
 - generated documents
 - ...
- Mandatory [viewpoints scenarios](#)

A project needs comply with a specific solution architecture scenario.
The projects dashboard shows the completion status for each project
If you click on one of the projects, you can see, for that project

 - what viewpoints are expected
 - which viewpoints are missing
 - how many are missing (with auto-coloring depending on the number)
- All repositories: Word document assembly for Principles document and Standards document generation

LANGUAGE TRANSFORMER

- Simplified and extended language transformation capabilities
- Documented language transformation development lifecycle
- Full set of language transformation examples

METAMODEL

- [Language metamodels documentation](#), including
 - mapping of the metamodel to the transformation process
 - sample metamodel evolution
 - your metamodel step by step
 - related viewpoints needed
- Additional default connections

B0203

ADDIN

- Process: Added "Automation" tagged value
- Added missing connectors for "... Service" stereotypes in default metamodel

B0202

ADDIN

- Generate Tabular Report on demand (right-click on any Tabular Report template)
- Floating licenses managed by Sparx Systems Keystore Service

REPOSITORIES

- Sample Repository
 - Added different kinds of sample charts
 - Added [template content for project scope](#)
- Repositories documentation
 - Added links to EA documentation and sample files to illustrate how to import new elements from CSVs

CHANGES TO THE DEFAULT LANGUAGE METAMODEL

Added

- Application, Application Component, Data Store, Equipment and Material --is part of --> Representation
- Entity, Role and Process --is owned by--> Functional Domain, Area, Block, Category and Service

POWERSHELL

- ImportCsv creates tagged values even if they have not been defined in the modeling language

B0201

LABNAF GUIDANCE MODEL REPOSITORY

This is a new item added to the Labnaf product list. It is the source model repository from which the entire online guidance is generated. It is provided for free with a license of the Labnaf Server.

EA'S USER EXPERIENCE OPTIMIZATIONS FOR LABNAF

On startup, Labnaf Addin now always optimizes EA's user experience for Labnaf as follows:

- Show the Properties Info Section
- Hide Stereotypes in the Project Browser
- Disable fully scoped object names
- Disable Auto Instance
- Disable Automatic Sub-Activities
- Auto Group Elements OFF
- Prompt for connector delete
- No Strict Connector Syntax

CHANGES TO THE DEFAULT LANGUAGE METAMODEL

Added

- Logical Node --Realizes--> Technology Function
- Logical Node --Network Interlink-- Communication Network (*needed when Logical Node is a firewall*)

TAGGED VALUE CHANGES

Default value for tag "Form Factor" = "Virtual":

- Logical Node and Communication Network are now virtual by default.

B0200

Added [Labnaf PowerShell](#) featuring:

- Job scheduler
- Diagram generation following Labnaf diagram templates defined in the Labnaf model repository.
- Excel generation following Labnaf Excel templates
- Word generation following Sparx System's document template approach.
- Publication to an HTML site where pages can be shared using standard e-mails.
- Automated tagged value calculation following configuration defined in the Labnaf model repository.

- Invalid connector detection and email error reports distribution
- Backup to local model repository.

B0195

Added [Language Transformer](#) featuring automatic transformation, in a repository, of existing modeling language elements and connectors in terms of types, stereotypes and tagged values. Enables translation of any language e.g. ArchiMate, BPMN or UML, into any other language e.g. Labnaf (and vice versa).

B0191

Added Startup repository and Sample repository.

The startup model repository structure includes

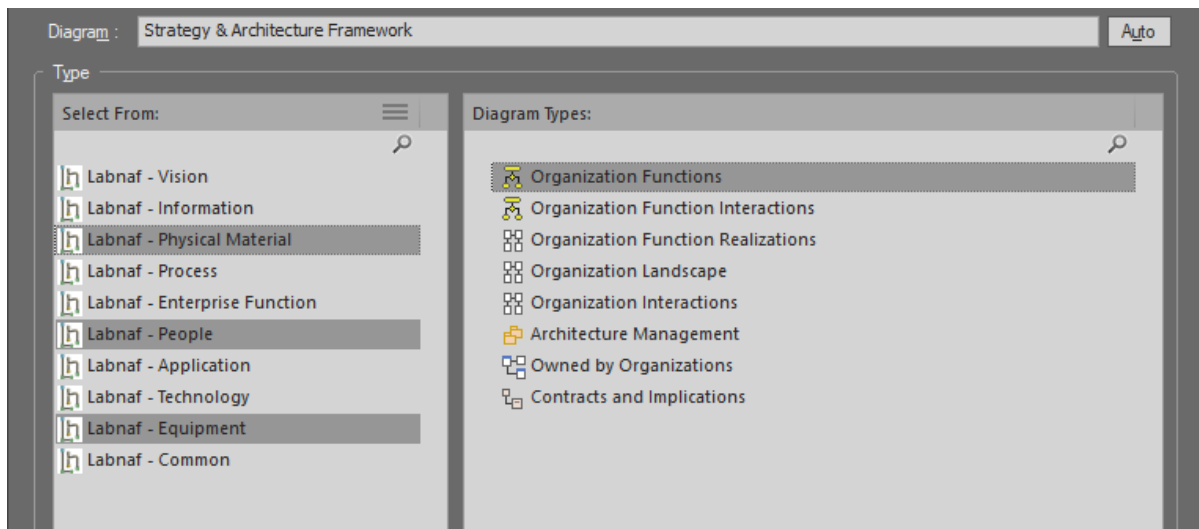
- A configurable language metamodel with dynamic alignment of the model validation rules
- Canonical and configurable repository structure of packages, elements and diagrams that one can copy from
- Configurable templates for diagram generation
- Model templates for configuring value calculations
- Reusable diagrams and elements (auto-legends...)

The sample model repository structure includes samples models and a canonical folder structure. Content of the canonical folder structure, at any level of detail, can be cut and paste into the production repository.

B0190

The [strategy and architecture content](#) organization has been semantically purified

- The language is richer in terms of vocabulary. You can now do precise modeling of IoT and ecosystems.
- [Viewpoints](#) are very consistently organized following a common refinement approach
- Toolboxes have been grouped accordingly



- The [language metamodel](#) has been extended and thoroughly organized into different perspectives.
- Architects configure the metamodel in one click and generate the validation rules in one other click
- Everything is now precisely [aligned to the original systems semantics](#).

The language [customization workbench](#) has been much improved and is now also configured in one click.

The screenshot shows the Labnaf Customization Workbench application window. At the top, there is a menu bar with 'Save', 'Reload', and 'Load MDG from:' options (Files, Addin, None). Below this is the 'Software Development Lifecycle Environment' section with three rows for Development, Testing, and Production folders, each with a 'Select Folder' button and an 'Activate' button. A yellow highlighted section titled 'Active Runtime Configuration on this PC:' shows 'TESTING' as the active configuration, with 'Load MDG file from folder:' and 'MDG file expected in this folder:' fields. Below this is a table of actions categorized by 'Software Development Lifecycle' and 'Time Last Changed'.

Software Development Lifecycle	Time Last Changed
MDG Development	
Edit MDG Source Model (EAP)	C:\Users\User\Desktop\Labnaf\Customization\1_Dev\Labnaf_Custom_Deve 2019-09-01 17:45:05
Edit MDG Deployment File (MTS)	C:\Users\User\Desktop\Labnaf\Customization\1_Dev\Labnaf_Custom.MTS 2019-09-02 11:58:57
Generate MDG => Testing	
MDG Testing	
Activate Testing Configuration	C:\Users\User\Desktop\Labnaf\Customization\2_Test\Labnaf_Custom_MDG 2019-09-01 17:45:34
Open Testing Repository	C:\Users\User\Desktop\Labnaf\Customization\2_Test\Labnaf_Test_Repository 2019-08-19 12:19:16
Copy MDG from Testing => Production	
Configuring Production	
Activate Production Configuration	C:\Users\User\Desktop\Labnaf\Customization\3_Prod\Labnaf_Custom_MD 2019-09-01 17:45:34
Open Production Repository & Edit Metamodel	C:\Users\User\Desktop\Labnaf\Customization\3_Prod\Labnaf_Prod_Reposit 2019-09-02 11:59:30
Generate Connectors Definition => Production	C:\Users\User\Desktop\Labnaf\Customization\3_Prod\Labnaf_Custom_Quir 2019-09-01 21:25:37