



Unified Framework for Driving Transformations

Customization Workbench

User Guide

Labnaf Customization Tools =>	Labnaf AddIn: Instant Metamodel Manager	Labnaf AddIn: Instant Metadata Manager	Customization Workbench
Categories of Customization			
Metamodel	<i>Customizable metamodel expressed using the Labnaf Language and stored in the production database.</i>		
Select active metamodel (standard, customized standard, or user-defined metamodel)	Y		
Add/Delete connectors to a metamodel	Y		
Upgrade standard metamodel to a new version of Labnaf, while keeping your own customizations	Y		
Generate documentation/diagram about your metamodel customization	Y		
Element Properties			
Create custom property types		Y	Y
Upgrade properties to a new version of Labnaf and keep/restore your own customizations		Y	
Rename/Delete property types		Y	Y
Synchronize property sets in existing elements		All props	MDG props
Add custom properties to the Tags Tab		Y	N
Add Labnaf Properties to the Element Tab		N	MDG-defined
Create named property groups		N	MDG-defined
Connector Properties			
Create custom connector properties		N	MDG-defined
Elements and Connectors			
Add/Delete element and connector types		N	MDG-defined
Toolboxes and Diagram Types/Viewpoints			
Add/Update/Delete Toolboxes		N	MDG-defined
Add/Update/Delete Diagram Types/Viewpoints		N	MDG-defined
Change Shapes		N	MDG-defined
Change Icons		N	MDG-defined

[More info on the Labnaf Guidance Web Site](#)

Labnaf provides two options for language customization

1. Instant Metamodel Manager and the Instant Metadata Manager

The **Instant Metamodel Manager** and the **Instant Metadata Manager** are provided by the Labnaf AddIn i.e. directly in the modeling environment.

Using these tools, you can

- use the standard built-in metamodel, customize it, or create your own user-defined metamodel from scratch,
- visualize, create, rename, delete and synchronize element properties,
- automatically merge new versions of Labnaf with your own metamodel and metadata (properties) customizations

2. Customize the Labnaf MDG Using the Customization Workbench

This is used for **advanced customizations**.

The **Customization Workbench** is used for customizing the Labnaf MDG including, properties/tagged values, element types, connector types, toolboxes, diagram types, and metamodel (still using the Labnaf end user language itself).

To activate this customization mode, start the Customization Workbench, and select the "Load MDG from Files" option. This overrides the above features on (1).

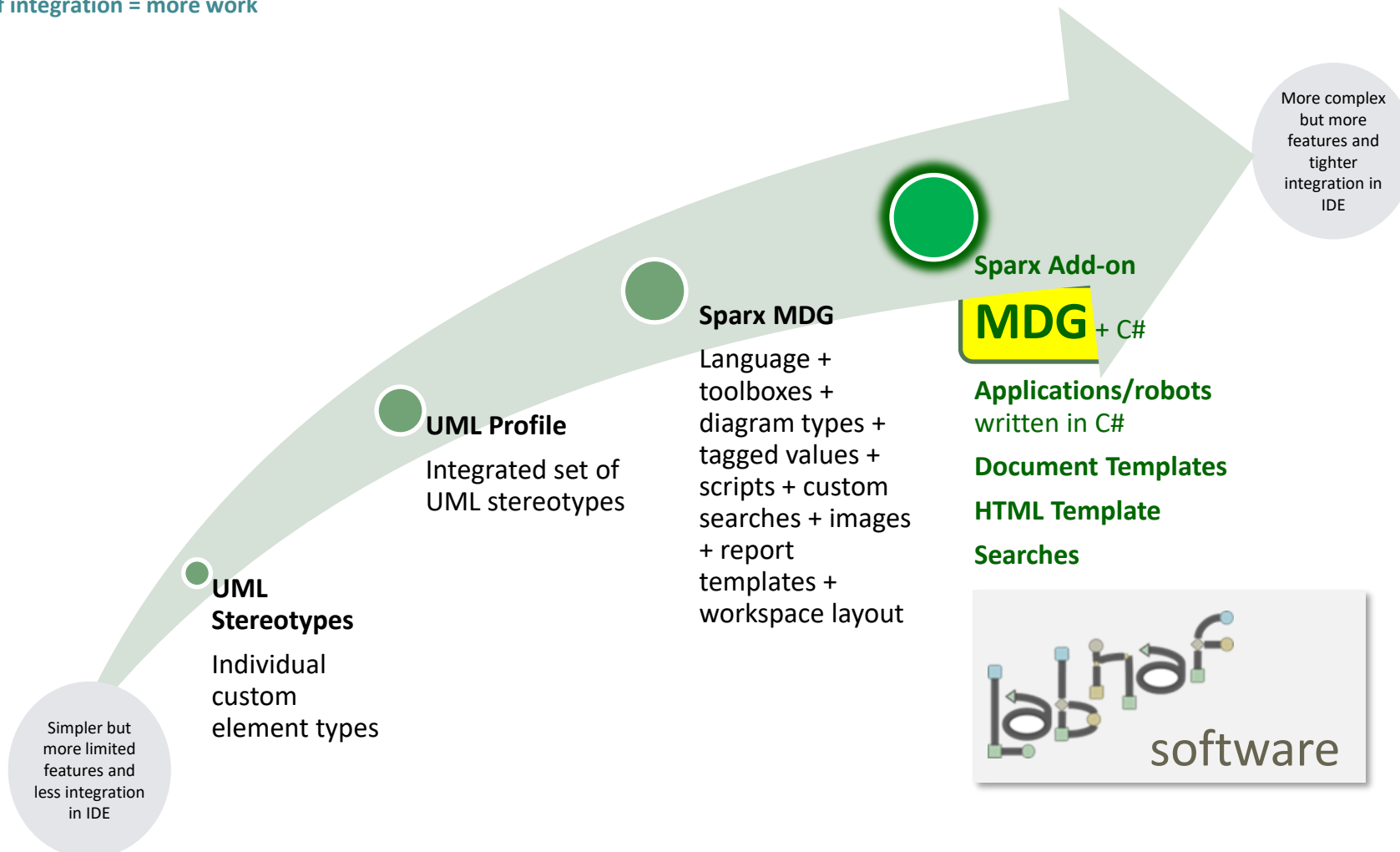
In this case Labnaf still automatically merges the new Labnaf metamodel version with your own customizations, since Labnaf metamodels are entirely controlled by Labnaf .

But for the other parts of the language specifications, which are defined in the MDG, you need to rely on Sparx tools.

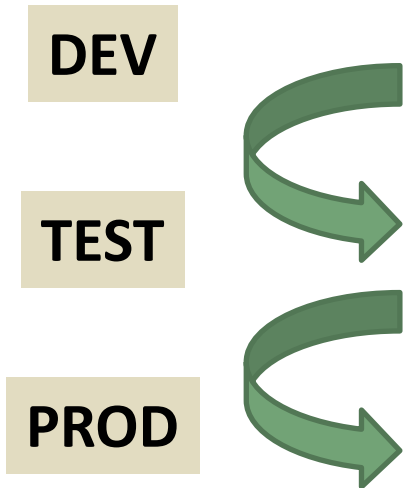
Labnaf was build using the Sparx EA Software Development Kit (SDK)

Enables multiple levels of modeling solution development and integration

Higher level of integration = more work



The Labnaf Customization Workbench provides you with a staging environment for customizing the **Labnaf MDG**



Labnaf Customization Workbench

Save Reload Load MDG from: Files AddIn About

Software Development Lifecycle Environment

Development folder: C:\ALT\SparxDev\Distributed\Labnaf\Environments\1_Dev

Testing configuration folder: C:\ALT\SparxDev\Distributed\Labnaf\Environments\2_Test

Production configuration folder: H:\Tools\LabnafConfig

Active Runtime Configuration on this PC: PROD

Load MDG file from folder: H:\Tools\LabnafConfig

MDG file expected in this folder: H:\Tools\LabnafConfig\Labnaf_Custom_MDG.xml 2019-03-05 11:59:12

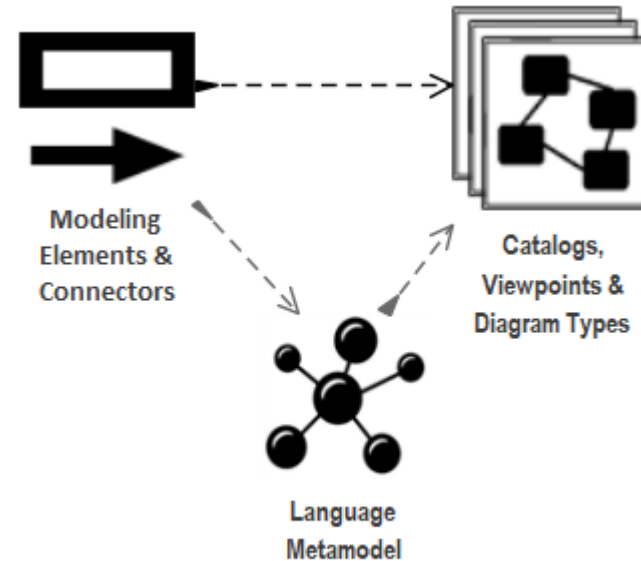
Connectors Definition used (always from PROD): H:\Tools\LabnafConfig\Labnaf_Custom_QuickLinks.xml 2019-03-04 17:36:25

Software Development Lifecycle

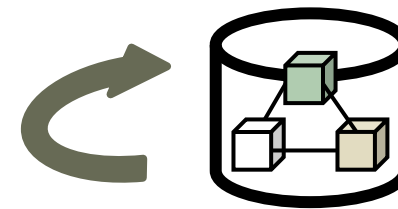
	Time Last Changed
MDG Development	
<input type="button" value="Edit MDG Source Model (EAP)"/>	C:\ALT\SparxDev\Distributed\Labnaf\Environments\1_Dev\Labnaf_Custom 2019-03-05 12:50:36
<input type="button" value="Edit MDG Deployment File (MTS)"/>	C:\ALT\SparxDev\Distributed\Labnaf\Environments\1_Dev\Labnaf_Custom 2019-03-05 11:24:02
<input type="button" value="Generate MDG => Testing"/>	
MDG Testing	
<input type="button" value="Activate Testing Configuration"/>	C:\ALT\SparxDev\Distributed\Labnaf\Environments\2_Test\Labnaf_Custom 2019-03-05 11:59:12
<input type="button" value="Open Testing Repository"/>	C:\ALT\SparxDev\Distributed\Labnaf\Environments\2_Test\Labnaf_Test_Re 2019-03-05 11:41:16
<input type="button" value="Copy MDG from Testing => Production"/>	
Configuring Production	
<input type="button" value="Activate Production Configuration"/>	H:\Tools\LabnafConfig\Labnaf_Custom_MDG.xml 2019-03-05 11:59:12
<input type="button" value="Open Production Repository & Edit Metamodel"/>	H:\Tools\LabnafConfig\Labnaf_Prod_Repository.eap 2019-02-28 18:01:07
<input type="button" value="Generate Connectors Definition => Production"/>	H:\Tools\LabnafConfig\Labnaf_Custom_QuickLinks.xml 2019-03-04 17:36:25

Labnaf Customization Steps

1. Customize the language following your organization requirements

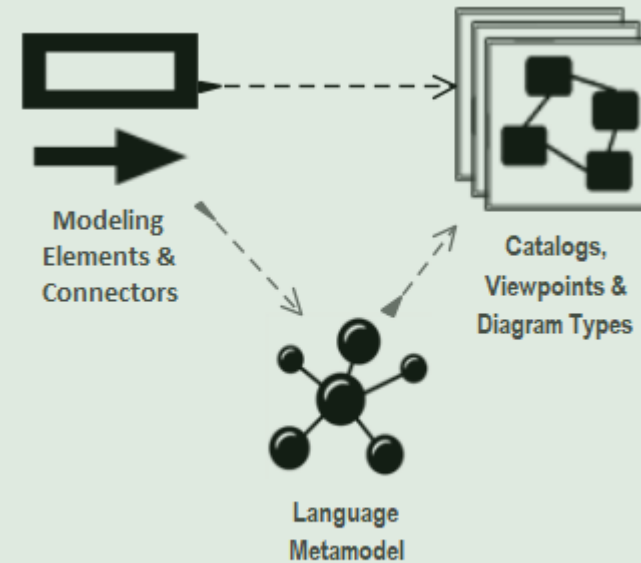


2. Adapt existing repository content

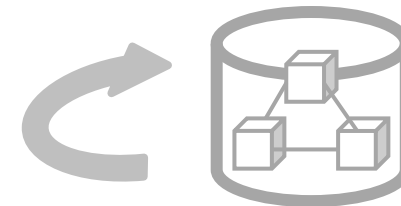


Labnaf Customization Workbench

1. Customize the language following your organization requirements



2. Adapt existing repository content



Sparx EA Modeling Languages

What's in the box

Many model repository options

- **Local** Microsoft **Access** database
- **Shared** database: **SQL Server**, MySQL, Oracle...
- Possible **version control**: Subversion, CVS, TFS...

Many languages & Toolboxes

- (Too) Many languages
- No integration of standards
- Scope / domain-specific
- Switch between many complex toolboxes

Sparx SDK

- Tailoring the tool for an organization
 - Language
 - IDE
 - document generation
 - web publication
 - ...

Built-in MDGs and related Toolboxes

Use Case	XML Schema	GoF Patterns
Class	Documentation	ICONIX
Object	Test Domain	LieberLieber AUTOSAR Engineer
Composite	Dashboard	MindMapping
Communication	XMLTransform	NIEM
Interaction	ArcGIS	ODM
Timing	ArchMate	Project Management
State	ArchMate2	RiskTaxonomy
Activity	BPMN 1.1	SCMF 2.1
Component	BPMN 2.0	SPEM
Deployment	BPMN 1.0	User Interface - Simple
Profile	UML Standard Profile	SoaML
Metamodel	Business Rule Model	Strategic Modeling
Analysis	CodeEngineering	UMM 2.0 Profile
Business Modeling	Data Flow Diagrams	UPCC 2.0
Custom	Data Modeling	UPCC 3.0
Requirements	Entity Relationship Diagram	UBL Model Management
Maintenance	Eriksson-Penker Extensions	WebModeling
User Interface	GML	Whiteboard
WSDL	GRA-UML	User Interface - Win32
		Wireframing

What we don't want

What we want



Merged Standards & Best Practices



One Strategy & Architecture Process



One Modeling Language

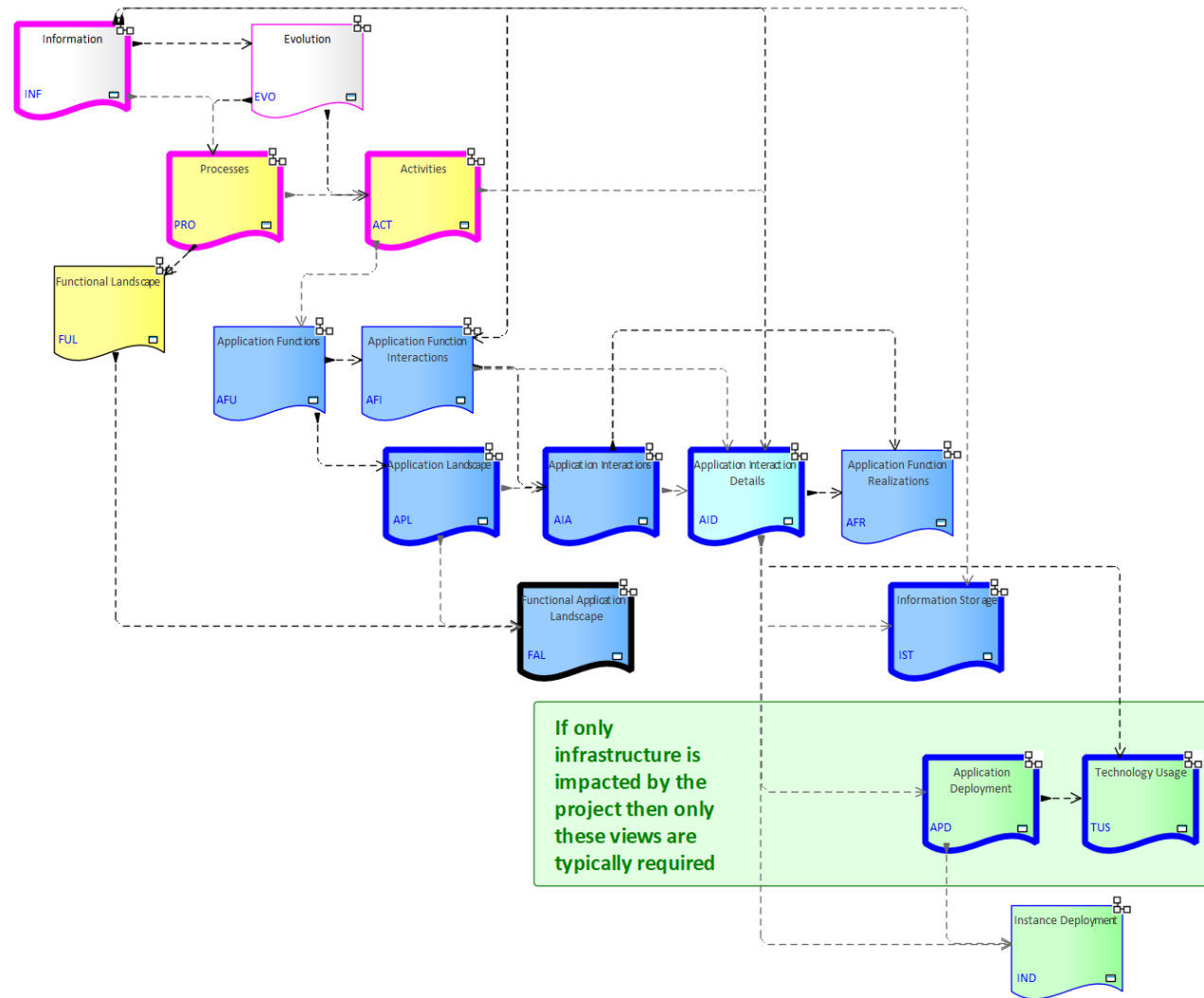


One Tool & One Repository



Extensive On-line Documentation

We need one modular language and tool but address specifically each type of view



Each type of architecture view needs to have its toolbox with element and connector types

Elements & Connectors

- Some come from standards
- Some were adapted
- Some are proprietary

What we want

Examples

Activities

- Activity
- Gateway
- Intermediate Event
- End Event
- Role
- Start Event
- Swimlane
- Data Object
- Representation
- Application Function
- Application Component
- Location

Activities Connectors

- Activity Trigger
- Realizes
- Access

Application Deployment

- Application as a Service
- Technology as a Service
- Application Deployment Set
- Application Component
- Data Store
- Logical Node
- System Software
- Location

Application Deployment Connectors

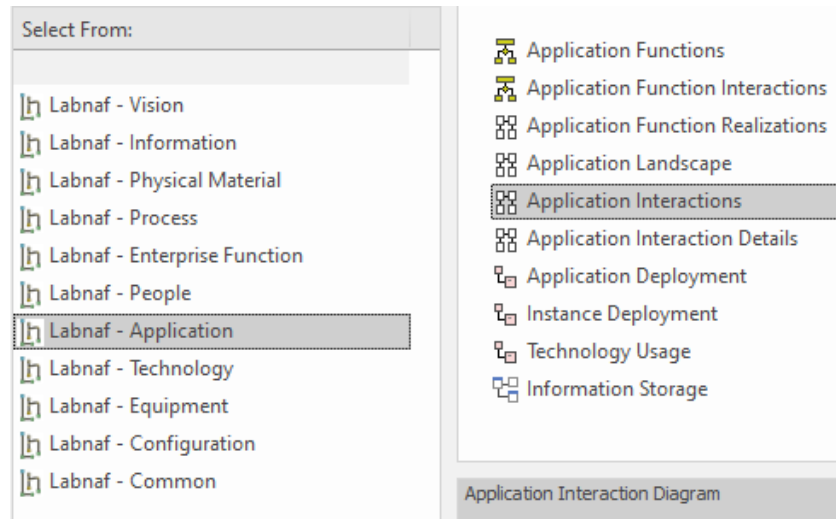
- Is deployed on
- Is part of
- Realizes
- Is dependent on
- Flow allowed by firewall

We use verbs instead of nouns to indicate the meaning of the connector's directions

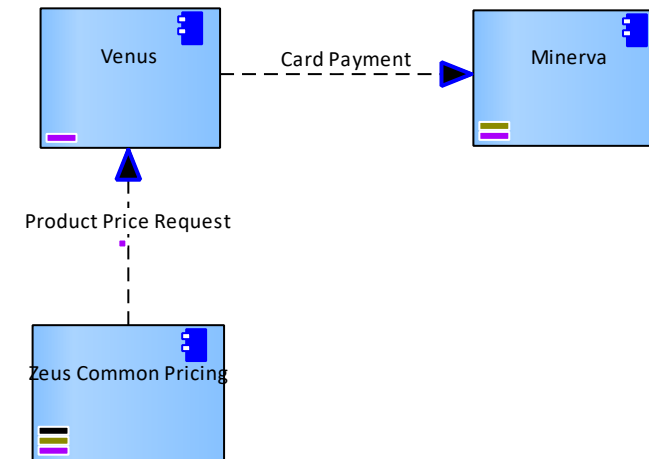
Diagram Types

What we want

When you select New diagram / ... the following list of Labnaf perspectives and diagram types appears



A Labnaf diagram type addresses a specific strategy or architecture viewpoint



Toolboxes

What we want

Each diagram type has its own toolbox.

Each diagram-specific toolbox only contains the elements and connectors that are relevant to this diagram type.

Toolbox for the diagram type “Application Interactions”

- Application Interaction
 - Application
 - Role
 - Organization Function
 - Organization
- Application Interaction Connectors
 - Application Flow
 - Depends on

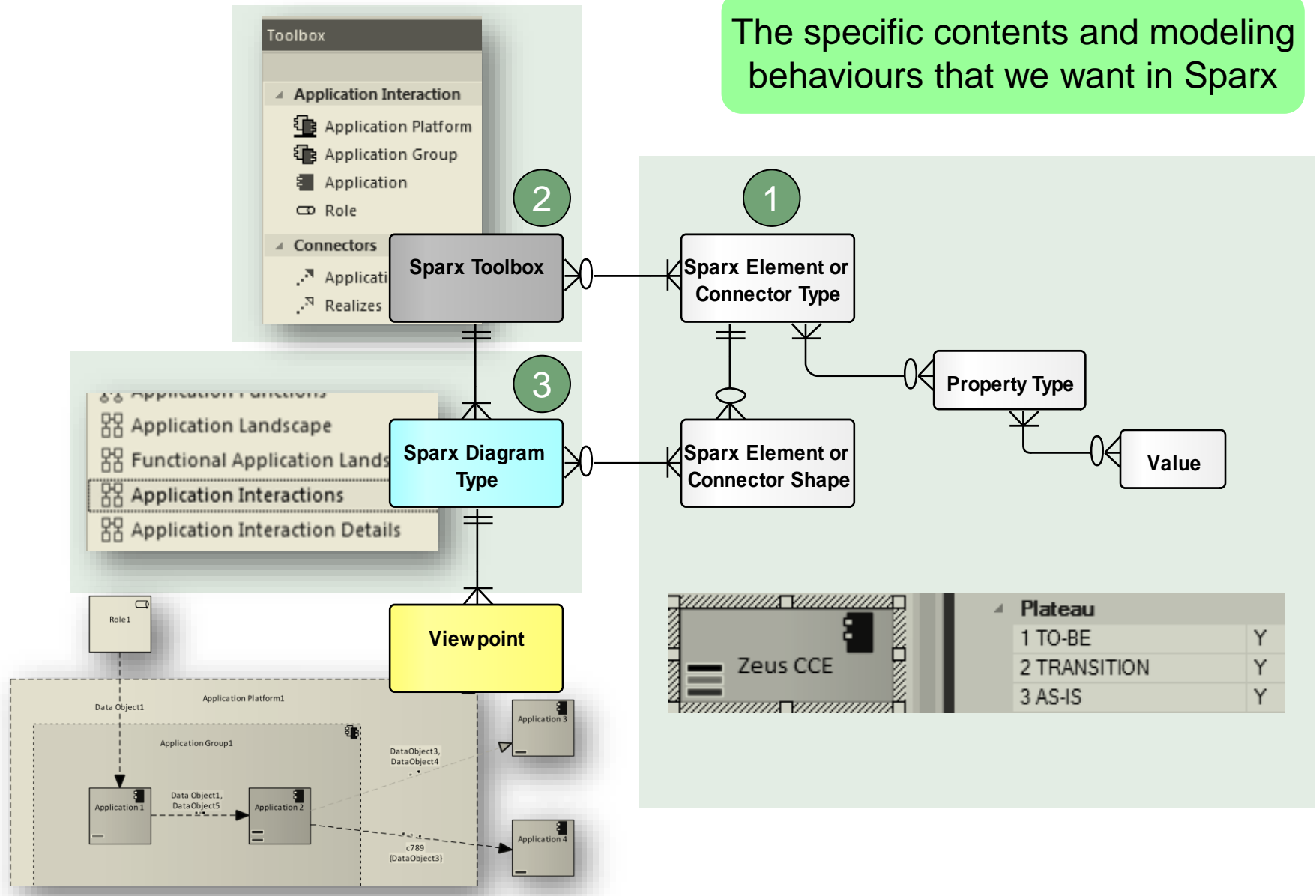
All elements and connectors used

(These are not actual toolboxes)

- Vision
 - Strategic Theme
 - Strategic Objective
 - Goal
 - Standard
 - Principle
 - Demand
 - Epic
 - Capability
 - Feature
 - Story
- Information
 - Information Domain
 - Entity
 - Representation
 - Data Object
- Process
 - Process
 - Start Event
 - Intermediate Event
 - Gateway
 - Activity
 - End Event
 - Swimlane
- Enterprise Function
 - Functional Domain
 - Functional Area
 - Functional Block
 - Functional Category
 - Functional Service
 - Access Point
- Physical
 - Equipment Function
 - Equipment Service
 - Equipment
 - Equipment Type
 - Facility
 - Distribution Network
 - Location
 - Material
- People
 - Organization Function
 - Organization Service
 - Role
 - Organisation
 - Individual
 - Contract
- Application
 - Application Function
 - Application as a Service
 - Application Service
 - Application Platform
 - Application Group
 - Application
 - Application Component
 - Data Store
- Technology
 - Technology Function
 - Technology as a Service
 - Node Type
 - System Software
 - Communication Network
 - Interface Protocol
 - Application Deployment Set
 - Logical Node
 - Instance Deployment Set
 - Node Instance
- All Connectors
 - Access
 - Association
 - Application Flow
 - Component Flow
 - Depends on
 - Deployment
 - Evolves into
 - Flow is allowed by firewall
 - Functional Flow
 - Functional Performer Flow
 - Impacts
 - Influences
 - Instance is part of
 - Instance is deployed on
 - Instance realizes
 - Is a constituent of
 - Is assigned to
 - Is bound by contract
 - Is owned by
 - Is part of
 - Network Interlink
 - Node to Network Connection
 - Path between Nodes
 - Physical Flow
 - Realizes
 - Specializes
 - Triggers

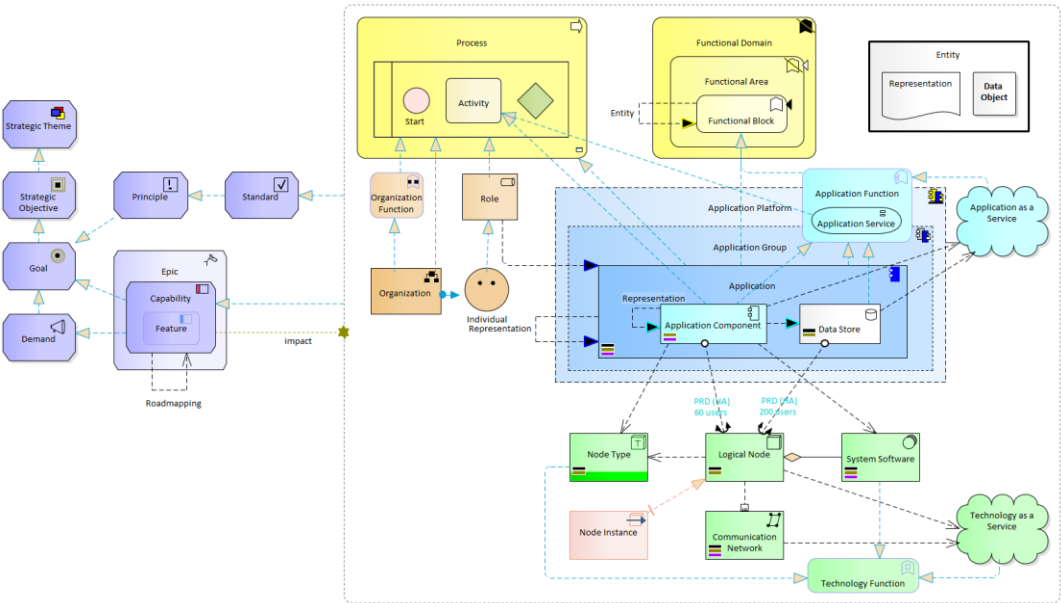
Key items to be designed and configured

The specific contents and modeling behaviours that we want in Sparx



We also want an agile Language Metamodel used both for documentation & automatic model validation

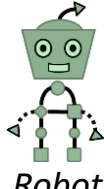
What we want



While Modeling

Existing Invalid Connectors

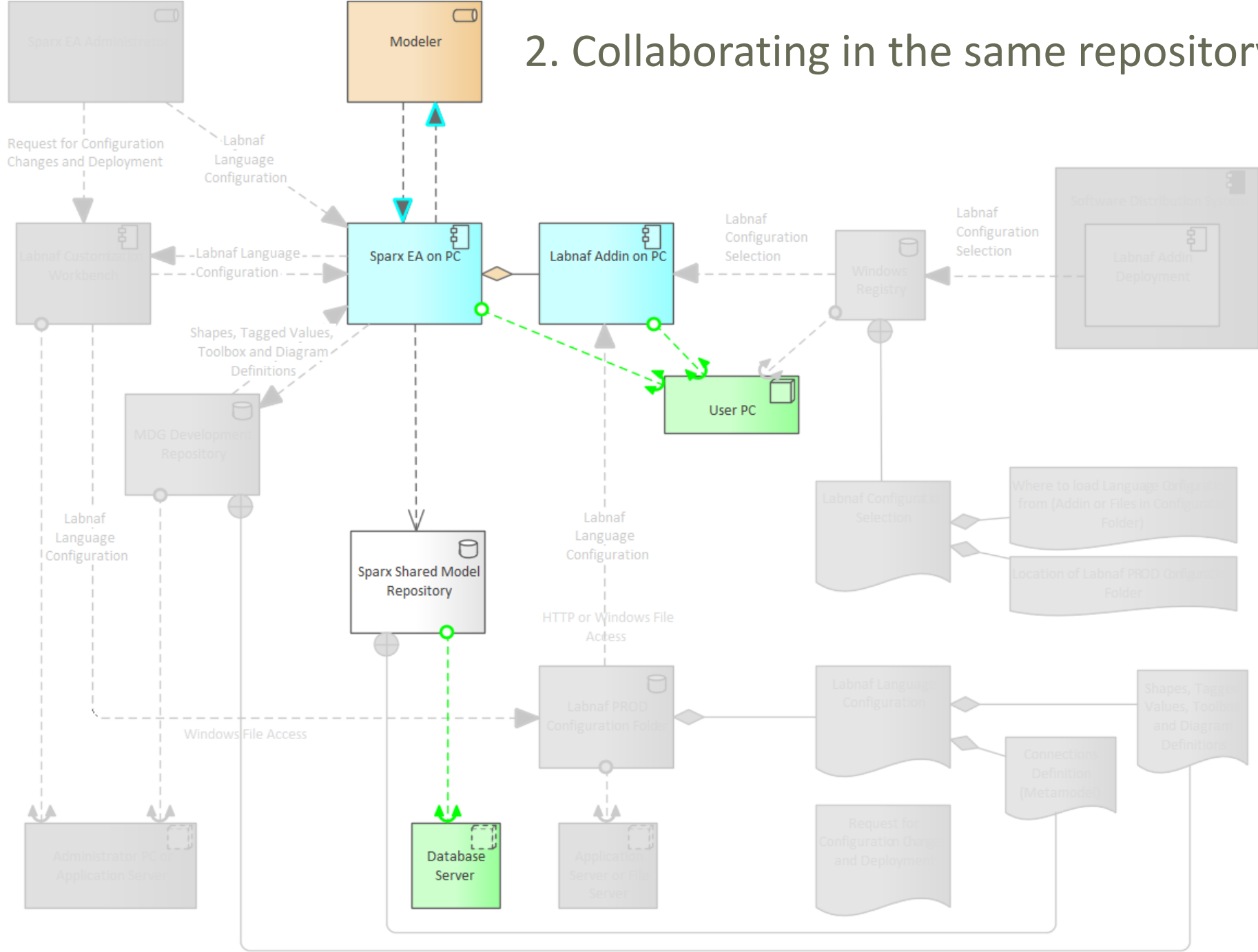
Prevent creation of invalid connectors



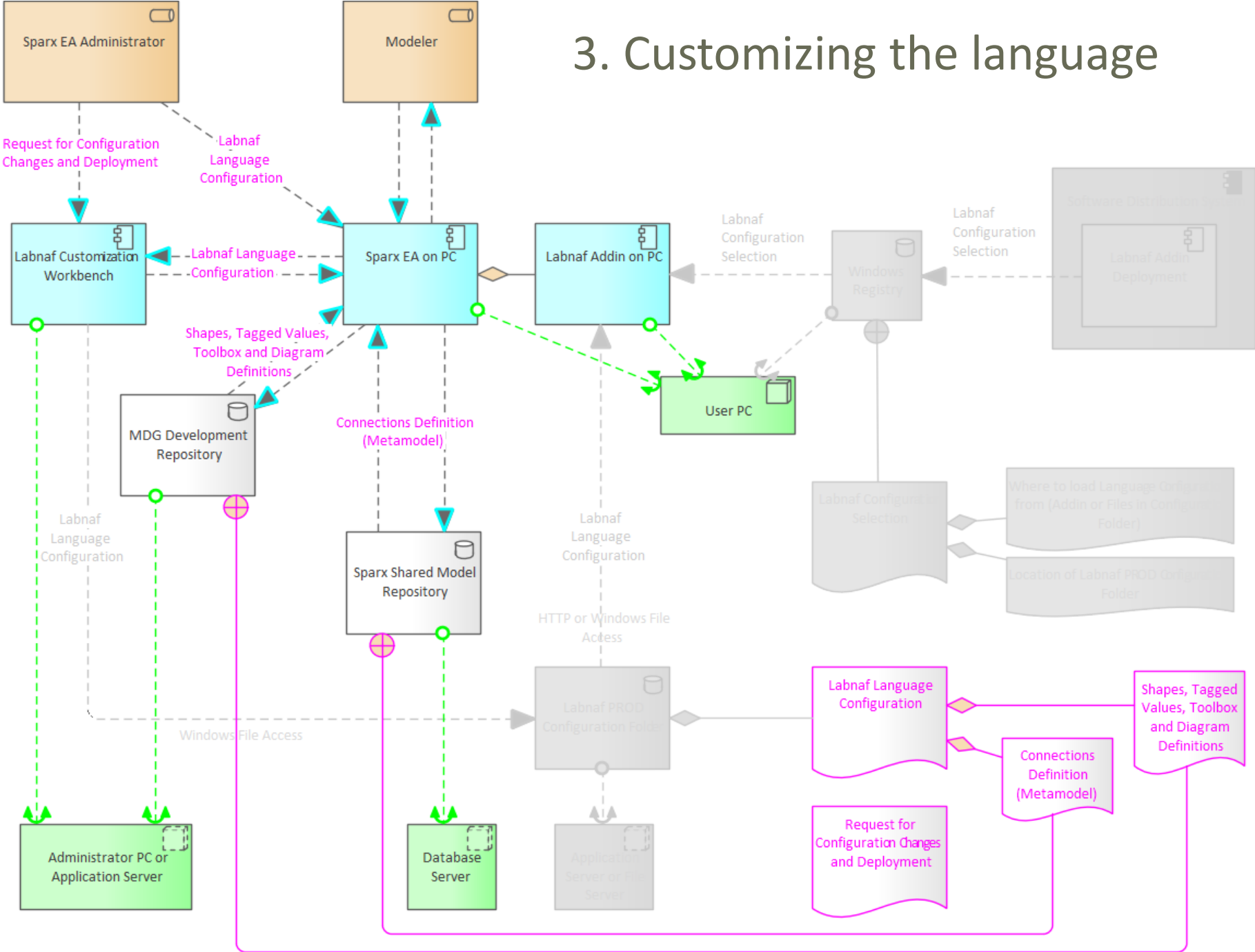
Send Error Emails to Relevant Recipients

The evolving environment

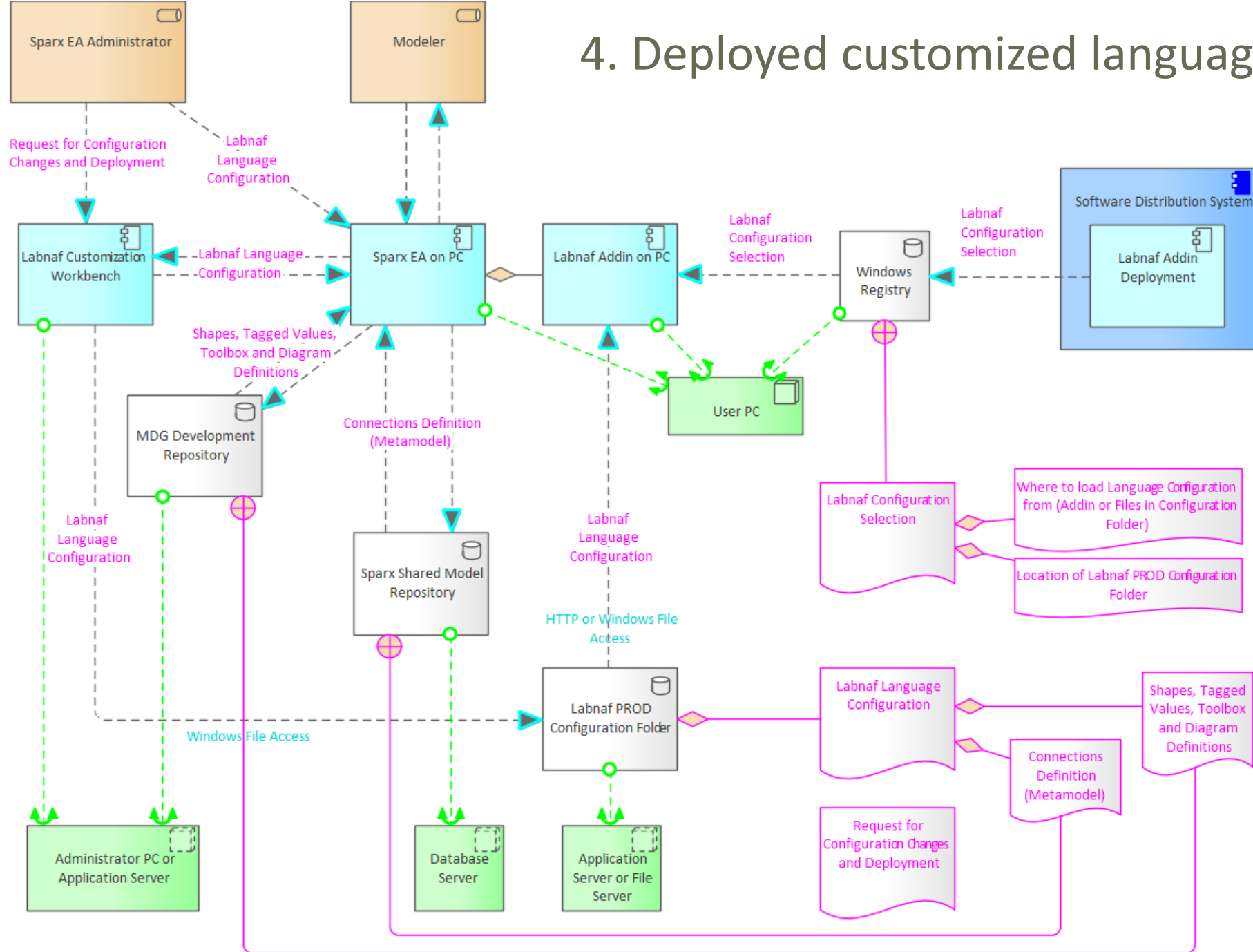
2. Collaborating in the same repository



3. Customizing the language



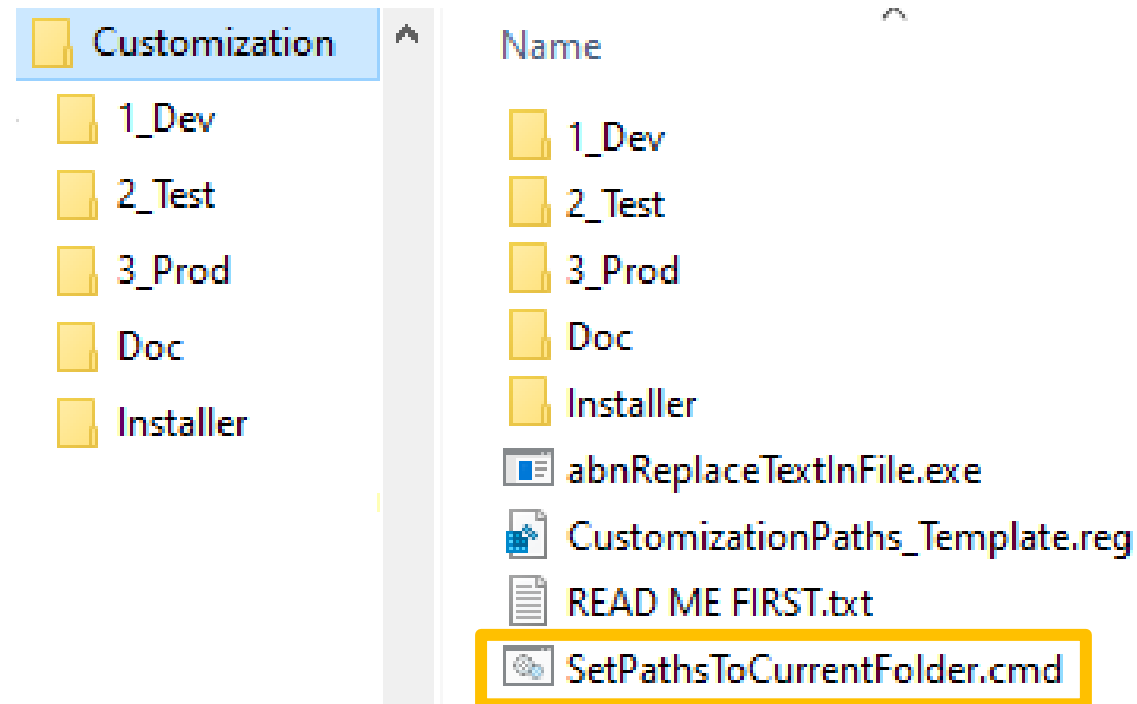
4. Deployed customized language



How to proceed in practice...

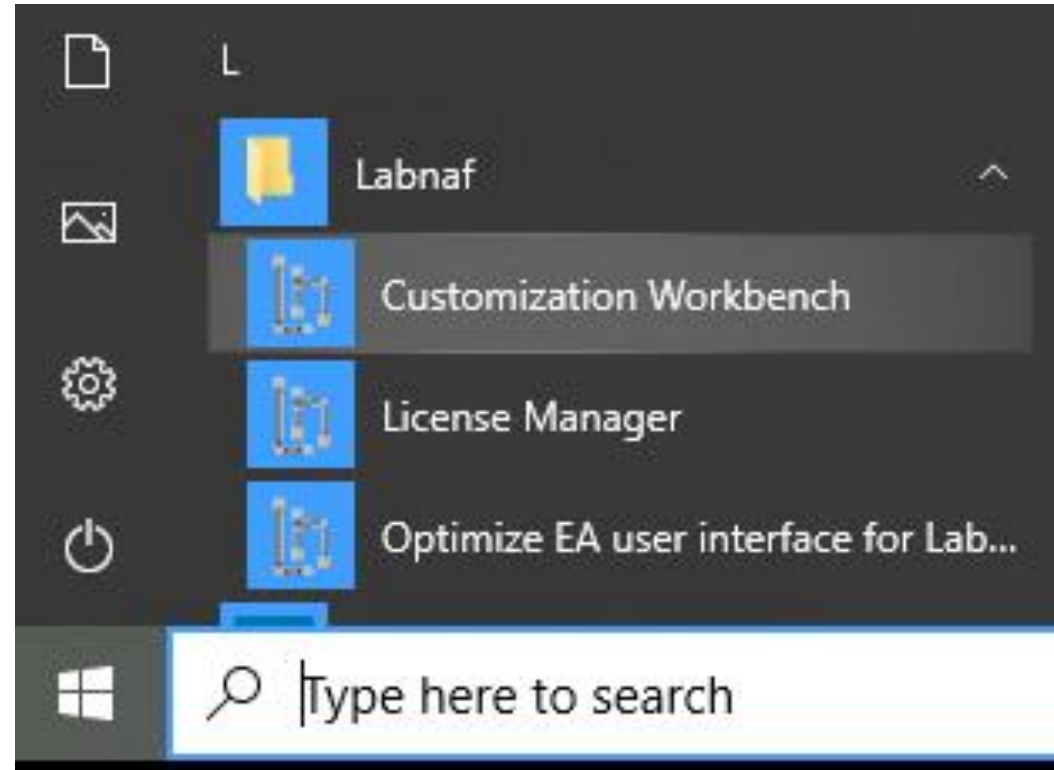
Automatic configuration to start customization

- Copy the Labnaf_Customization folder anywhere you want on your file system
- Double-click on “SetPathsToCurrentFolder.cmd”

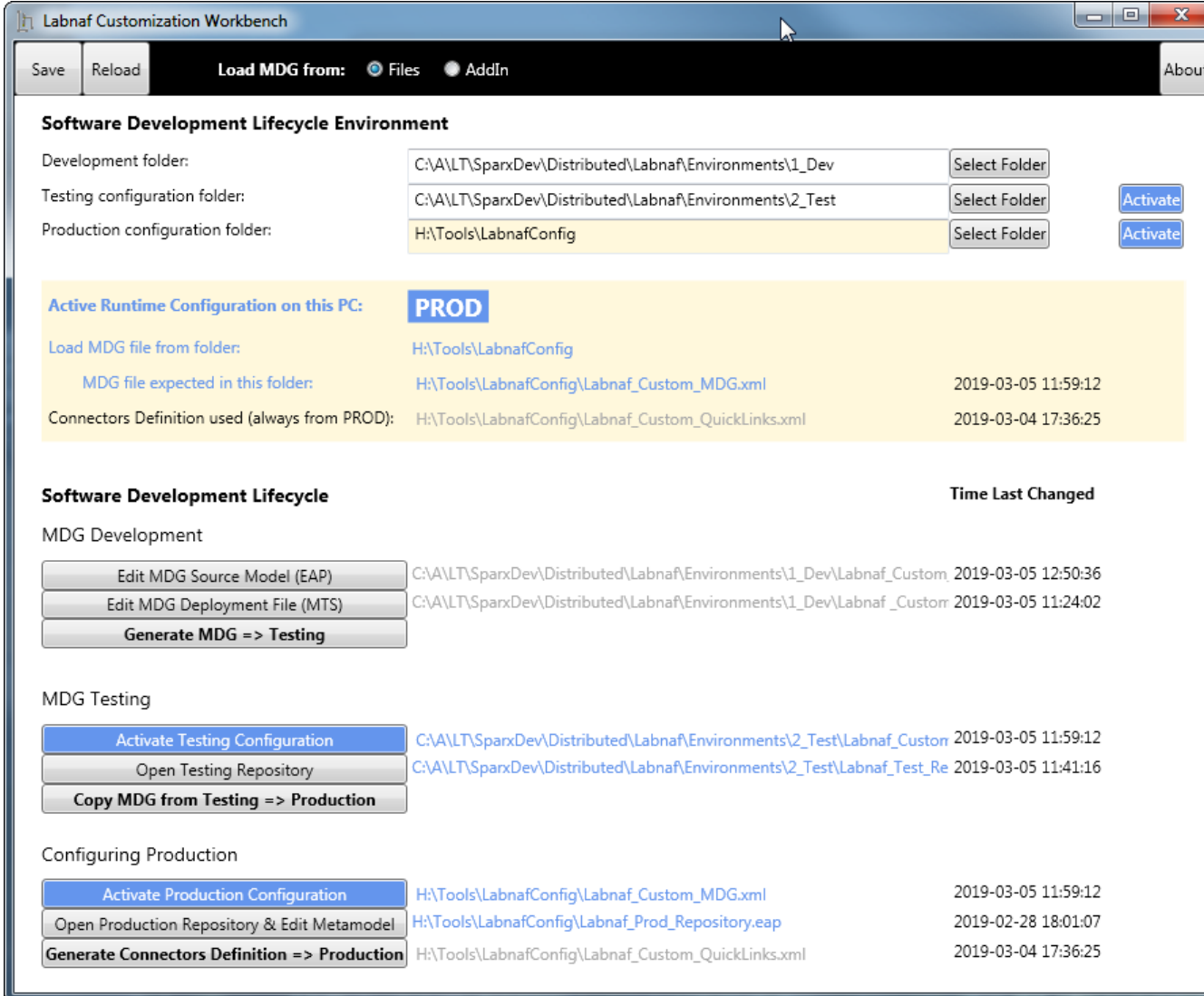


This updates the Labnaf configuration files and registry keys following the “Labnaf_Customization” folder location.

Start the Customization Workbench...



The Customization Workbench guides you throughout the **Language customization** lifecycle

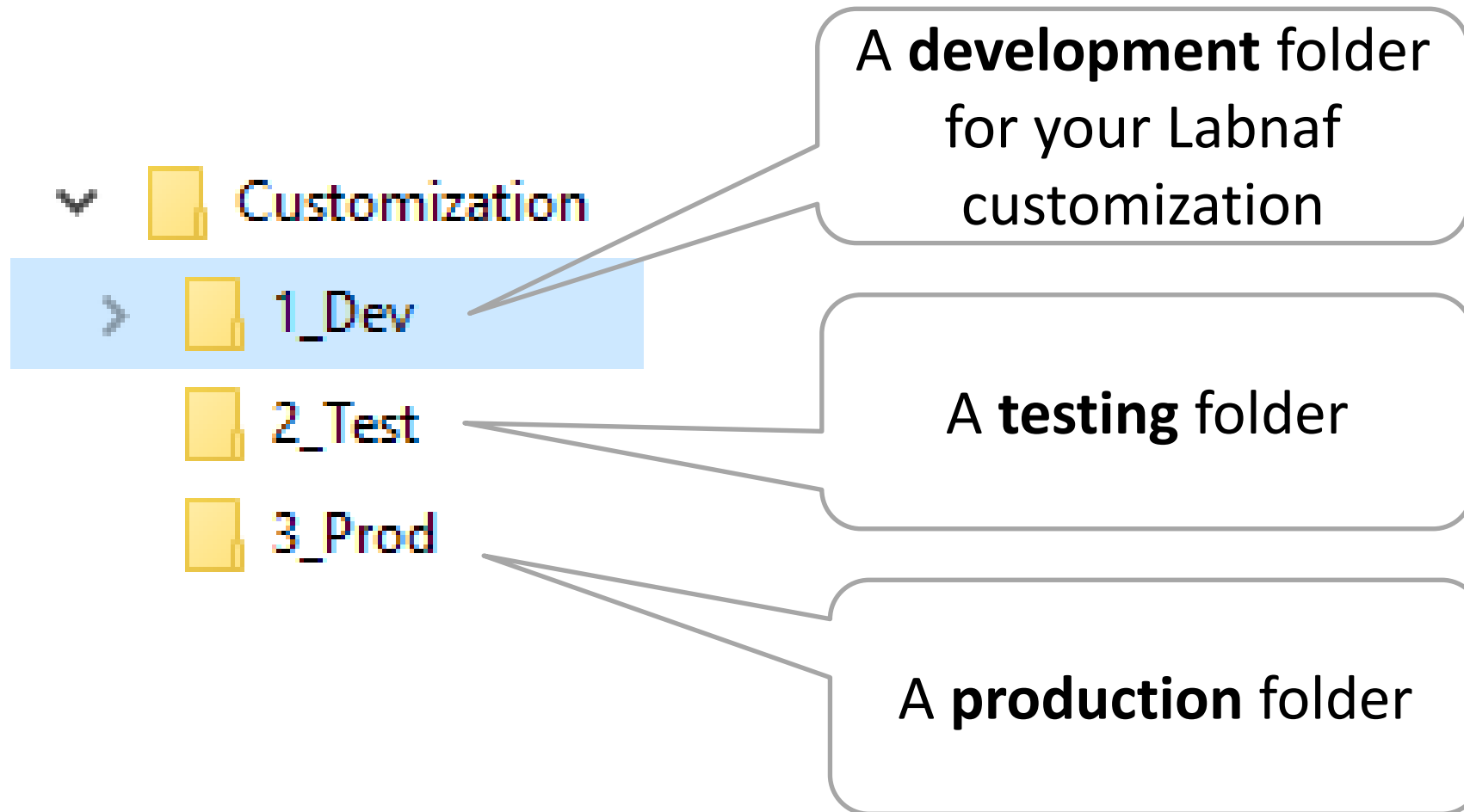


The screenshot displays the Labnaf Customization Workbench interface. At the top, there are buttons for 'Save', 'Reload', and 'About', along with a 'Load MDG from:' section with radio buttons for 'Files' and 'AddIn'. The main area is divided into several sections:

- Software Development Lifecycle Environment:** Contains three rows for 'Development folder', 'Testing configuration folder', and 'Production configuration folder'. Each row has a text input field, a 'Select Folder' button, and an 'Activate' button.
- Active Runtime Configuration on this PC:** A yellow highlighted section showing the active configuration as 'PROD'. It lists 'Load MDG file from folder', 'MDG file expected in this folder', and 'Connectors Definition used (always from PROD)' with their respective paths and timestamps.
- Software Development Lifecycle:** A table-like view with a 'Time Last Changed' column. It is organized into three categories: 'MDG Development', 'MDG Testing', and 'Configuring Production'. Each category contains several actions with buttons and their corresponding file paths and timestamps.



Like in any SDLC (Software Development Lifecycle) , there is one environment for each Labnaf customization stage



SDLC Environment Setting

Select the **Development**, **Testing** and **Production** configuration folders

The screenshot shows the 'Labnaf Customization Workbench' window. At the top, there are 'Save' and 'Reload' buttons, and a 'Load MDG from:' section with radio buttons for 'Files' (selected) and 'AddIn'. An 'About' button is on the right. The main content area is titled 'Software Development Lifecycle Environment' and contains three rows of settings:

Development folder:	C:\A\LT\SparxDev\Distributed\Labnaf\Environments\1_Dev	Select Folder	
Testing configuration folder:	C:\A\LT\SparxDev\Distributed\Labnaf\Environments\2_Test	Select Folder	Activate
Production configuration folder:	H:\Tools\LabnafConfig	Select Folder	Activate

Below this, a yellow box indicates the 'Active Runtime Configuration on this PC: **PROD**'. It lists the following settings:

Load MDG file from folder:	H:\Tools\LabnafConfig	
MDG file expected in this folder:	H:\Tools\LabnafConfig\Labnaf_Custom_MDG.xml	2019-03-05 11:59:12
Connectors Definition used (always from PROD):	H:\Tools\LabnafConfig\Labnaf_Custom_QuickLinks.xml	2019-03-04 17:36:25

The bottom section is titled 'Software Development Lifecycle' and includes a 'Time Last Changed' column. It lists three actions:

	Time Last Changed
Edit MDG Source Model (EAP)	C:\A\LT\SparxDev\Distributed\Labnaf\Environments\1_Dev\Labnaf_Custom 2019-03-05 12:50:36
Edit MDG Deployment File (MTS)	C:\A\LT\SparxDev\Distributed\Labnaf\Environments\1_Dev\Labnaf_Custorr 2019-03-05 11:24:02
Generate MDG => Testing	

The Development Stage

Development Folder

- Pictures
- WSL
- LABN_Diagrams_01_Vision.xml
- LABN_Diagrams_02_Information.xml
- LABN_Diagrams_03_Physical_Material.xml
- LABN_Diagrams_04_Process.xml
- LABN_Diagrams_05_Enterprise_Function.xml
- LABN_Diagrams_06_People.xml
- LABN_Diagrams_07_Application.xml
- LABN_Diagrams_08_Technology.xml
- LABN_Diagrams_09_Equipment.xml
- LABN_Diagrams_10_Configuration.xml
- LABN_Diagrams_11_Common.xml
- LABN_Diagrams_99_All.xml
- LABN_TB_Activities.xml
- LABN_TB_Any.xml
- LABN_TB_App_Dep.xml
- LABN_TB_App_Func.xml
- LABN_TB_App_Func_Interactions.xml
- LABN_TB_App_Func_Realizations.xml
- LABN_TB_App_Inter_Details.xml
- LABN_TB_App_Interactions.xml
- LABN_TB_App_Land.xml
- LABN_TB_Archi_Management.xml
- LABN_TB_Connectivity.xml
- LABN_TB_ContractsAndImplications.xml
- LABN_TB_ControlledElementValues.xml
- LABN_TB_Corp_Strategy_Map.xml
- LABN_TB_Demands.xml
- LABN_TB_Distribution.xml
- LABN_TB_Entities.xml
- LABN_TB_Equipment_Func.xml
- LABN_TB_Equipment_Func_Relizations.xml
- LABN_TB_Equipment_Land.xml
- LABN_TB_Evolution.xml
- LABN_TB_FreeText.xml
- LABN_TB_Func_App_Land.xml
- LABN_TB_Func_Eqp_Land.xml
- LABN_TB_Func_Interactions.xml
- LABN_TB_Func_Land.xml
- LABN_TB_Func_Org_Land.xml
- LABN_TB_Goals.xml
- LABN_TB_HLReqRoadmap.xml
- LABN_TB_Info_Prod_and_Usage.xml
- LABN_TB_Info_Storage.xml
- LABN_TB_Information.xml
- LABN_TB_Instance_Dep.xml
- LABN_TB_Locations.xml
- LABN_TB_Material.xml
- LABN_TB_Motivations.xml
- LABN_TB_Org_Func.xml
- LABN_TB_Org_Func_Interactions.xml
- LABN_TB_Org_Func_Realizations.xml
- LABN_TB_Org_Interactions.xml
- LABN_TB_Org_Land.xml
- LABN_TB_Owned_By_Ent_Func.xml
- LABN_TB_Owned_By_Organizations.xml
- LABN_TB_Phys_Func_Interactions.xml
- LABN_TB_Phys_Interactions.xml
- LABN_TB_Principles.xml
- LABN_TB_Process_Realizations.xml
- LABN_TB_Processes.xml
- LABN_TB_Standards.xml
- LABN_TB_StdTechSvc.xml
- LABN_TB_Tabular_Report_Template_Design.xml
- LABN_TB_Tech_Func.xml
- LABN_TB_Tech_Func_Realizations.xml
- LABN_TB_Tech_Land.xml
- LABN_TB_Tech_Usage.xml
- Labnaf_Custom_Dev.eap
- Labnaf_Custom_MDG.xml
- Labnaf_Custom_Profile.xml
- Labnaf_Custom_Template.MTS

Step 1: Develop your customized Labnaf language

- **Labnaf_Custom_Dev.eap:** A Labnaf model repository for customizing your Labnaf modeling language
- **Pictures:** Folder for storing custom language images

Step 2: Generate language “profiles”

- **LABN_Diagrams_...xml:** Diagram types
- **LABN_Custom_Profile.xml:** Element & connector types
- **LABN_TB....xml:** Toolboxes

Step 3: Use the wizard to update

- **Labnaf_Custom.MTS:** The binding of all profile files needed
- **Labnaf_Custom_MDG.xml:** Your resulting customized Labnaf modeling language (generated in the Test folder)

For further details, see Sparx’Systems MDG Technologies User’s Guide @ <https://sparxsystems.com/resources/user-guides/modeling/mdg-technologies.pdf>

Customization

1_Dev

2_Test

3_Prod

Develop your customized Labnaf language

The screenshot shows the Labnaf Customization Workbench interface. At the top, there are buttons for 'Save' and 'Reload', and a 'Load MDG from:' section with radio buttons for 'Files' (selected) and 'AddIn'. An 'About' button is in the top right. The main area is titled 'Software Development Lifecycle Environment' and contains three rows for environment configuration: 'Development folder' (C:\A\LT\SparxDev\Distributed\Labnaf\Environments\1_Dev), 'Testing configuration folder' (C:\A\LT\SparxDev\Distributed\Labnaf\Environments\2_Test), and 'Production configuration folder' (H:\Tools\LabnafConfig). Each row has a 'Select Folder' button, and the testing and production rows also have 'Activate' buttons. Below this is a yellow-highlighted section for 'Active Runtime Configuration on this PC: PROD'. It lists 'Load MDG file from folder' (H:\Tools\LabnafConfig), 'MDG file expected in this folder' (H:\Tools\LabnafConfig\Labnaf_Custom_MDG.xml, 2019-03-05 11:59:12), and 'Connectors Definition used (always from PROD):' (H:\Tools\LabnafConfig\Labnaf_Custom_QuickLinks.xml, 2019-03-04 17:36:25). The bottom section is 'Software Development Lifecycle' with a 'Time Last Changed' column. A yellow box highlights the 'MDG Development' section, which includes buttons for 'Edit MDG Source Model (EAP)', 'Edit MDG Deployment File (MTS)', and 'Generate MDG => Testing'. The first two buttons have associated file paths and timestamps.

Software Development Lifecycle Environment	
Development folder:	C:\A\LT\SparxDev\Distributed\Labnaf\Environments\1_Dev
Testing configuration folder:	C:\A\LT\SparxDev\Distributed\Labnaf\Environments\2_Test
Production configuration folder:	H:\Tools\LabnafConfig

Active Runtime Configuration on this PC: PROD		
Load MDG file from folder:	H:\Tools\LabnafConfig	
MDG file expected in this folder:	H:\Tools\LabnafConfig\Labnaf_Custom_MDG.xml	2019-03-05 11:59:12
Connectors Definition used (always from PROD):	H:\Tools\LabnafConfig\Labnaf_Custom_QuickLinks.xml	2019-03-04 17:36:25

Software Development Lifecycle		Time Last Changed
MDG Development		
Edit MDG Source Model (EAP)	C:\A\LT\SparxDev\Distributed\Labnaf\Environments\1_Dev\Labnaf_Custom	2019-03-05 12:50:36
Edit MDG Deployment File (MTS)	C:\A\LT\SparxDev\Distributed\Labnaf\Environments\1_Dev\Labnaf_Custom	2019-03-05 11:24:02
Generate MDG => Testing		

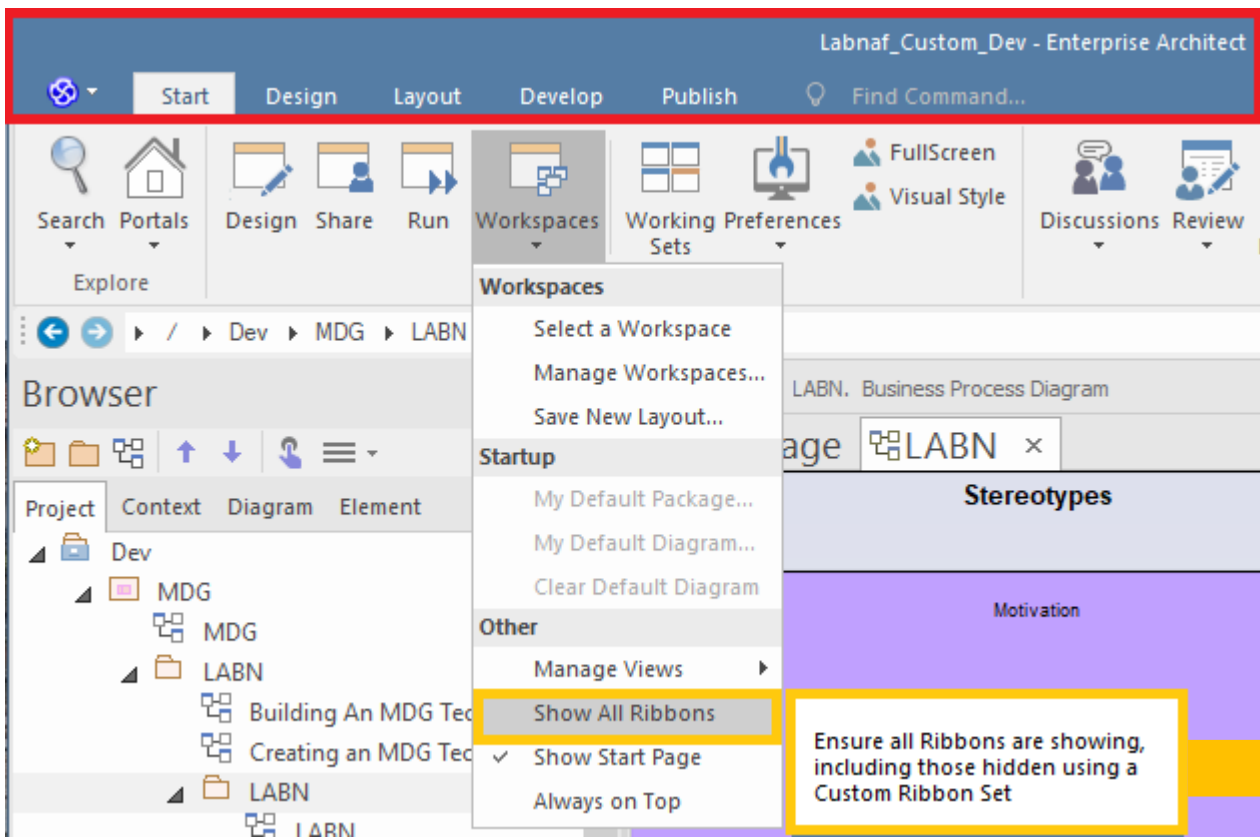
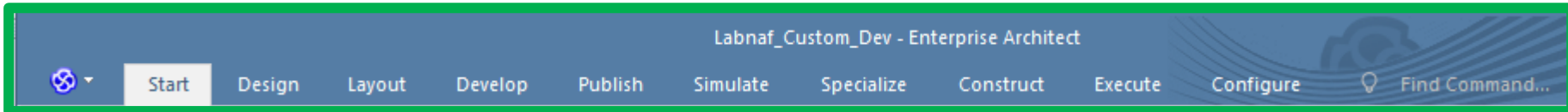
Customization

1_Dev

2_Test

3_Prod

As soon as the Labnaf_Custom_dev.eap in open **ensure all ribbons are showing**



Ensure all Ribbons are showing, including those hidden using a Custom Ribbon Set

Customization

1_Dev

2_Test

3_Prod

Set up your EA workspace layout for Labnaf



- This feature automatically opens and organizes all the EA windows that are useful with Labnaf including Labnaf customization
- Select the “**Start**” ribbon.
- Select the option “**Workspaces > Manage Workspaces > My Workspaces > Labnaf Workspace Layout**”.
- Press the “**Apply**” button.

Things you can change or add to the Labnaf_Custom_dev.eap

- Element & Connector types
- Tagged Values

Complex Sparx **Metamodeling** approach **replaced by Labnaf => Simplified, readable, dynamically customizable and upgradable at runtime**

The screenshot shows a configuration window with the following sections and options:

- Metamodel**
 - Profiles
 - Patterns
 - Diagram Types
 - Toolboxes
 - Tagged Value Types
- Code**
 - Code Modules
 - DDL Modules
 - MDA Transforms
- Reports**
 - RTF Templates
 - Linked Document Templates
- Other**
 - Images
 - Scripts
 - Workspace Layouts
- Model Views**
 - Model Views
 - Searches

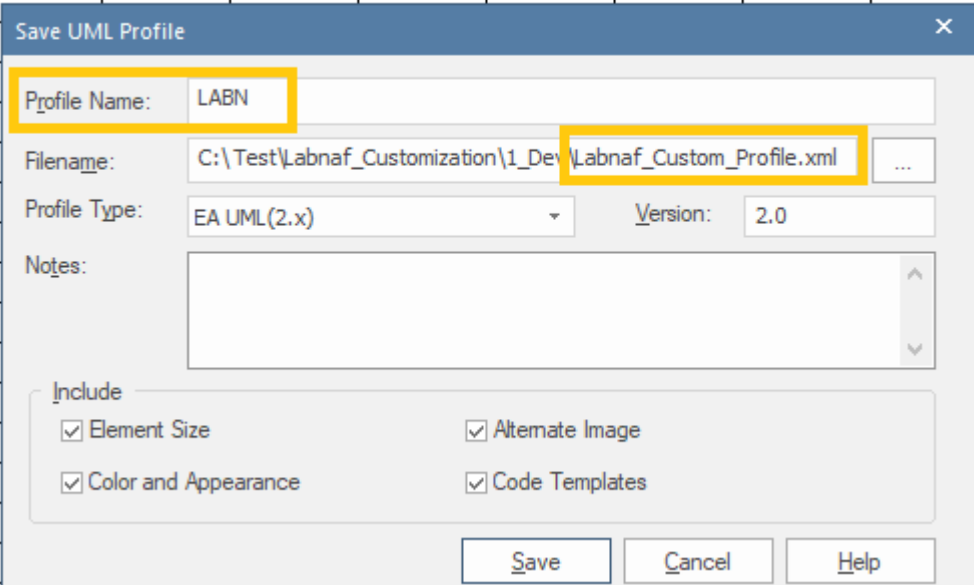
Updating the Labnaf MDG Profiles

See Sparx System's EA documentation about updating

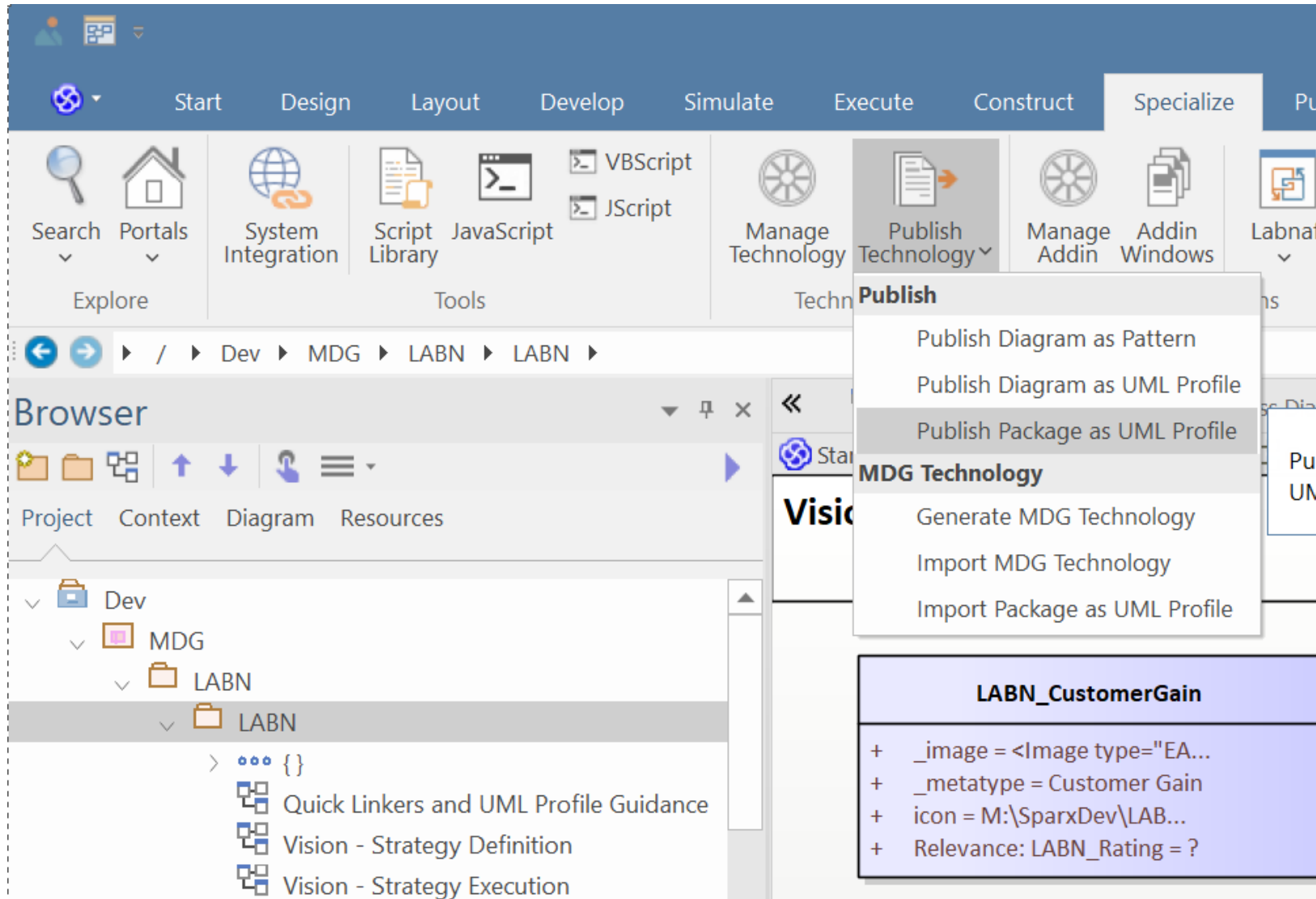
- [Stereotype Profiles](#)
- [Toolbox Profiles](#)
- [Diagram Profiles](#)

Refer to “**Labnaf Customization - Updating Profiles.xlsx**” to get the **Labnaf MDG profile** names and corresponding XML file names

	A	B	C	D	E	F	G	H	I	J
1	Profile Name	XML File Name								
2	LABN	Labnaf_Custom_Profile.xml								
3										
4										
5										
6										
7										
8										
9										
10										
11										
12										
13										
14										
15										



Saving the Labnaf MDG Profiles



Generate the Customized Labnaf MDG file (Language configuration)

The screenshot shows the Labnaf Customization Workbench interface. At the top, there are buttons for 'Save', 'Reload', and 'About'. The 'Load MDG from:' section has radio buttons for 'Files' (selected) and 'AddIn'. Below this, the 'Software Development Lifecycle Environment' section contains three rows for 'Development folder', 'Testing configuration folder', and 'Production configuration folder', each with a text input field and a 'Select Folder' button. The 'Production configuration folder' is highlighted in yellow. To the right of these fields are 'Activate' buttons. Below this is a yellow box titled 'Active Runtime Configuration on this PC:' with a 'PROD' label. It lists 'Load MDG file from folder' as 'H:\Tools\LabnafConfig', 'MDG file expected in this folder' as 'H:\Tools\LabnafConfig\Labnaf_Custom_MDG.xml' (dated 2019-03-05 11:59:12), and 'Connectors Definition used (always from PROD):' as 'H:\Tools\LabnafConfig\Labnaf_Custom_QuickLinks.xml' (dated 2019-03-04 17:36:25). The 'Software Development Lifecycle' section has a 'Time Last Changed' column. Under 'MDG Development', there are buttons for 'Edit MDG Source Model (EAP)', 'Edit MDG Deployment File (MTS)', and 'Generate MDG => Testing' (highlighted with a yellow box and an arrow). Under 'MDG Testing', there are buttons for 'Activate Testing Configuration' (highlighted with a yellow box), 'Open Testing Repository', and 'Copy MDG from Testing => Production'. The 'Activate Testing Configuration' button's tooltip shows the path 'C:\ALT\SparxDev\Distributed\Labnaf\Environments\2_Test\Labnaf_Custom...' and the date '2019-03-05 11:59:12'. Under 'Configuring Production', there are buttons for 'Activate Production Configuration', 'Open Production Repository & Edit Metamodel', and 'Generate Connectors Definition => Production'.

Customization

1_Dev

2_Test

3_Prod

The Testing Stage

Contents of the Testing Folder

 Labnaf_Custom_MDG.xml

 Labnaf_Test_Repository.eap

Labnaf_Custom_MDG.xml: Your customized Labnaf modeling language

Labnaf_Test_Repository.eap: Your model repository that you will use to test your customized Labnaf modeling language

Customization

1_Dev

2_Test

3_Prod

Activate the Testing Configuration and create some diagrams in the Testing Repository

The screenshot shows the Labnaf Customization Workbench interface. At the top, there are buttons for 'Save', 'Reload', and 'About'. The 'Load MDG from:' section has radio buttons for 'Files' (selected) and 'AddIn'. Below this, the 'Software Development Lifecycle Environment' section contains three rows for 'Development folder', 'Testing configuration folder', and 'Production configuration folder', each with a text input field, a 'Select Folder' button, and an 'Activate' button. The 'Testing configuration folder' is highlighted in yellow. Below this, a yellow box displays 'Active Runtime Configuration on this PC: TESTING' and lists 'Load MDG file from folder' and 'MDG file expected in this folder' with their respective paths and timestamps. The bottom section, 'Software Development Lifecycle', is a table with columns for actions and 'Time Last Changed'. It is divided into three sub-sections: 'MDG Development', 'MDG Testing', and 'Configuring Production', each with several action buttons and their corresponding paths and timestamps.

Software Development Lifecycle	Time Last Changed
MDG Development	
Edit MDG Source Model (EAP)	C:\A\LT\SparxDev\Distributed\Labnaf\Environments\1_Dev\Labnaf_Custom 2019-03-29 12:32:17
Edit MDG Deployment File (MTS)	C:\A\LT\SparxDev\Distributed\Labnaf\Environments\1_Dev\Labnaf_Custom 2019-03-29 12:36:02
Generate MDG => Testing	
MDG Testing	
Activate Testing Configuration	M:\SparxDev\Distributed\Labnaf\Environments\2_Test\Labnaf_Custom_MI 2019-03-29 12:36:02
Open Testing Repository	M:\SparxDev\Distributed\Labnaf\Environments\2_Test\Labnaf_Test_Reposit 2019-03-05 11:41:16
Copy MDG from Testing => Production	
Configuring Production	
Activate Production Configuration	M:\SparxDev\Distributed\Labnaf\Environments\3_Prod\Labnaf_Custom_MI 2019-03-10 12:15:46
Open Production Repository & Edit Metamodel	M:\SparxDev\Distributed\Labnaf\Environments\3_Prod\Labnaf_Prod_Repos 2019-03-27 17:18:17
Generate Connectors Definition => Production	M:\SparxDev\Distributed\Labnaf\Environments\3_Prod\Labnaf_Custom_Qu 2019-03-04 17:36:25

Customization

1_Dev

2_Test

3_Prod

Copy the Customized Labnaf MDG file to Production

The screenshot shows the Labnaf Customization Workbench interface. The 'Load MDG from:' dropdown is set to 'Files'. Under 'Software Development Lifecycle Environment', the 'Production configuration folder' is set to 'H:\Tools\LabnafConfig' and is highlighted in yellow. Below this, the 'Active Runtime Configuration on this PC:' is set to 'PROD'. A table shows the 'Load MDG file from folder:' as 'H:\Tools\LabnafConfig' with an expected MDG file 'H:\Tools\LabnafConfig\Labnaf_Custom_MDG.xml' last changed on 2019-03-05 11:59:12. The 'Connectors Definition used' is 'H:\Tools\LabnafConfig\Labnaf_Custom_QuickLinks.xml' last changed on 2019-03-04 17:36:25.

Software Development Lifecycle

	Time Last Changed
MDG Development	
Edit MDG Source Model (EAP)	C:\ALT\SparxDev\Distributed\Labnaf\Environments\1_Dev\Labnaf_Custom 2019-03-05 12:50:36
Edit MDG Deployment File (MTS)	C:\ALT\SparxDev\Distributed\Labnaf\Environments\1_Dev\Labnaf_Custom 2019-03-05 11:24:02
MDG Testing	
Activate Testing Configuration	C:\ALT\SparxDev\Distributed\Labnaf\Environments\2_Test\Labnaf_Custom 2019-03-05 11:59:12
Open Testing Repository	C:\ALT\SparxDev\Distributed\Labnaf\Environments\2_Test\Labnaf_Test_Re 2019-03-05 11:41:16
Copy MDG from Testing => Production	
Configuring Production	
Activate Production Configuration	H:\Tools\LabnafConfig\Labnaf_Custom_MDG.xml 2019-03-05 11:59:12
Open Production Repository & Edit Metamodel	H:\Tools\LabnafConfig\Labnaf_Prod_Repository.eap 2019-02-28 18:01:07
Generate Connectors Definition => Production	H:\Tools\LabnafConfig\Labnaf_Custom_QuickLinks.xml 2019-03-04 17:36:25

Customization

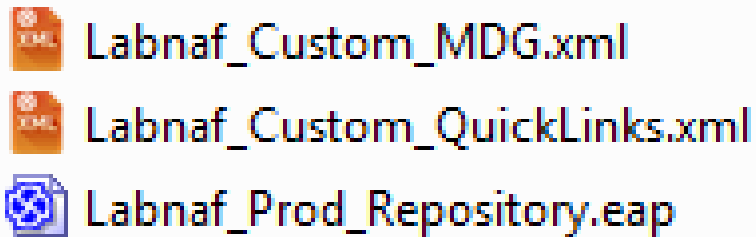
1_Dev

2_Test

3_Prod

The Production Stage

Contents of the Production Folder



The production folder can be located on a web server folder. In which case:

- The Customization Workbench accesses this folder directly on the file system.
- End users, using the Labnaf Addin, access the corresponding web folder url instead.

Labnaf_Custom_MDG.xml: Your customized Labnaf modeling language

Labnaf_Prod_Repository.eap: Your production model repository where you can dynamically customize the language metamodel

Labnaf_Custom_Quicklinks.xml: The connection rules generated from the language metamodel

Customization

1_Dev

2_Test

3_Prod

Activate the Production Configuration and open the Production Repository

The screenshot shows the Labnaf Customization Workbench interface. The '3_Prod' folder is selected in the left sidebar. The main window displays the 'Software Development Lifecycle Environment' configuration. The 'Production configuration folder' is set to 'H:\Tools\LabnafConfig' and is highlighted in yellow. Below this, the 'Active Runtime Configuration on this PC' is set to 'PROD'. The 'Load MDG file from folder' is 'H:\Tools\LabnafConfig'. The 'MDG file expected in this folder' is 'H:\Tools\LabnafConfig\Labnaf_Custom_MDG.xml' with a timestamp of '2019-03-05 11:59:12'. The 'Connectors Definition used (always from PROD)' is 'H:\Tools\LabnafConfig\Labnaf_Custom_QuickLinks.xml' with a timestamp of '2019-03-04 17:36:25'. The 'Software Development Lifecycle' section is divided into 'MDG Development' and 'MDG Testing'. The 'MDG Development' section includes buttons for 'Edit MDG Source Model (EAP)', 'Edit MDG Deployment File (MTS)', and 'Generate MDG => Testing'. The 'MDG Testing' section includes buttons for 'Activate Testing Configuration', 'Open Testing Repository', and 'Copy MDG from Testing => Production'. The 'Configuring Production' section is highlighted with a yellow box and includes buttons for 'Activate Production Configuration', 'Open Production Repository & Edit Metamodel', and 'Generate Connectors Definition => Production'.

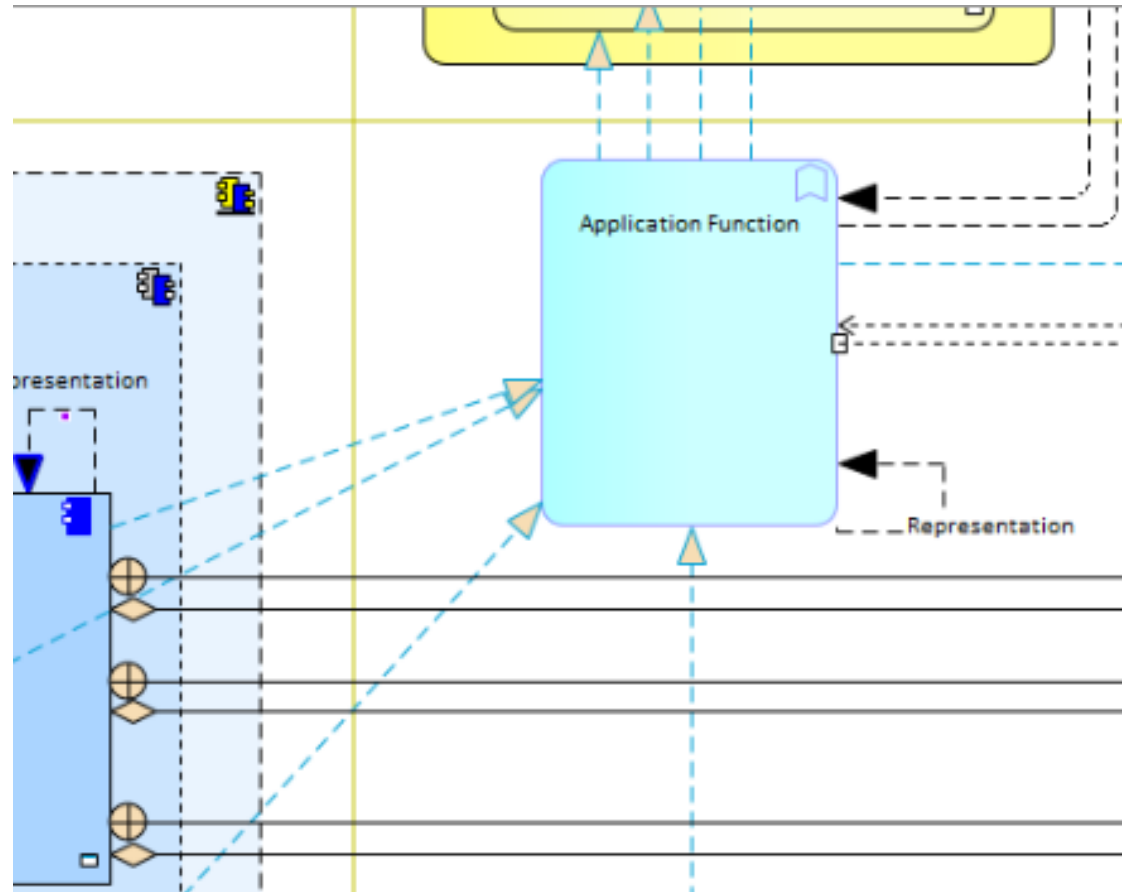
Software Development Lifecycle Environment	
Development folder:	C:\ALT\SparxDev\Distributed\Labnaf\Environments\1_Dev
Testing configuration folder:	C:\ALT\SparxDev\Distributed\Labnaf\Environments\2_Test
Production configuration folder:	H:\Tools\LabnafConfig

Active Runtime Configuration on this PC: PROD		
Load MDG file from folder:	H:\Tools\LabnafConfig	
MDG file expected in this folder:	H:\Tools\LabnafConfig\Labnaf_Custom_MDG.xml	2019-03-05 11:59:12
Connectors Definition used (always from PROD):	H:\Tools\LabnafConfig\Labnaf_Custom_QuickLinks.xml	2019-03-04 17:36:25

Software Development Lifecycle		Time Last Changed
MDG Development		
Edit MDG Source Model (EAP)	C:\ALT\SparxDev\Distributed\Labnaf\Environments\1_Dev\Labnaf_Custom	2019-03-05 12:50:36
Edit MDG Deployment File (MTS)	C:\ALT\SparxDev\Distributed\Labnaf\Environments\1_Dev\Labnaf_Custorr	2019-03-05 11:24:02
MDG Testing		
Activate Testing Configuration	C:\ALT\SparxDev\Distributed\Labnaf\Environments\2_Test\Labnaf_Custorr	2019-03-05 11:59:12
Open Testing Repository	C:\ALT\SparxDev\Distributed\Labnaf\Environments\2_Test\Labnaf_Test_Re	2019-03-05 11:41:16
Configuring Production		
Activate Production Configuration	H:\Tools\LabnafConfig\Labnaf_Custom_MDG.xml	2019-03-05 11:59:12
Open Production Repository & Edit Metamodel	H:\Tools\LabnafConfig\Labnaf_Prod_Repository.eap	2019-02-28 18:01:07
Generate Connectors Definition => Production	H:\Tools\LabnafConfig\Labnaf_Custom_QuickLinks.xml	2019-03-04 17:36:25

Update the language metamodel

- Add or delete connectors in the language metamodel



[Further details](#) on the Labnaf Guidance web site.

Customization

1_Dev

2_Test

3_Prod

(Re)generate the connector definitions

The screenshot shows the Labnaf Customization Workbench interface. The title bar reads "Labnaf Customization Workbench". The menu bar includes "Save", "Reload", "Load MDG from:" (with radio buttons for "Files" and "AddIn"), and "About".

Software Development Lifecycle Environment

Development folder: C:\ALT\SparxDev\Distributed\Labnaf\Environments\1_Dev [Select Folder]
Testing configuration folder: C:\ALT\SparxDev\Distributed\Labnaf\Environments\2_Test [Select Folder] [Activate]
Production configuration folder: H:\Tools\LabnafConfig [Select Folder] [Activate]

Active Runtime Configuration on this PC: PROD

Load MDG file from folder: H:\Tools\LabnafConfig
MDG file expected in this folder: H:\Tools\LabnafConfig\Labnaf_Custom_MDG.xml 2019-03-05 11:59:12
Connectors Definition used (always from PROD): H:\Tools\LabnafConfig\Labnaf_Custom_QuickLinks.xml 2019-03-04 17:36:25

Software Development Lifecycle Time Last Changed

MDG Development

Edit MDG Source Model (EAP)	C:\ALT\SparxDev\Distributed\Labnaf\Environments\1_Dev\Labnaf_Custom	2019-03-05 12:50:36
Edit MDG Deployment File (MTS)	C:\ALT\SparxDev\Distributed\Labnaf\Environments\1_Dev\Labnaf_Custorr	2019-03-05 11:24:02
Generate MDG => Testing		

MDG Testing

Activate Testing Configuration	C:\ALT\SparxDev\Distributed\Labnaf\Environments\2_Test\Labnaf_Custorr	2019-03-05 11:59:12
Open Testing Repository	C:\ALT\SparxDev\Distributed\Labnaf\Environments\2_Test\Labnaf_Test_Re	2019-03-05 11:41:16
Copy MDG from Testing => Production		

Configuring Production

Activate Production Configuration	H:\Tools\LabnafConfig\Labnaf_Custom_MDG.xml	2019-03-05 11:59:12
Open Production Repository & Edit Metamodel	H:\Tools\LabnafConfig\Labnaf_Prod_Repository.eap	2019-02-28 18:01:07
Generate Connectors Definition => Production	H:\Tools\LabnafConfig\Labnaf_Custom_QuickLinks.xml	2019-03-04 17:36:25

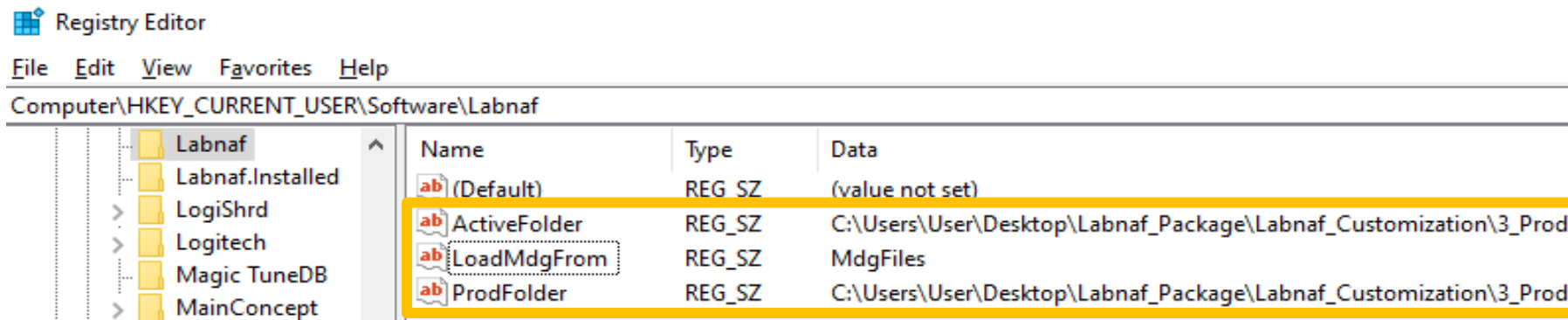
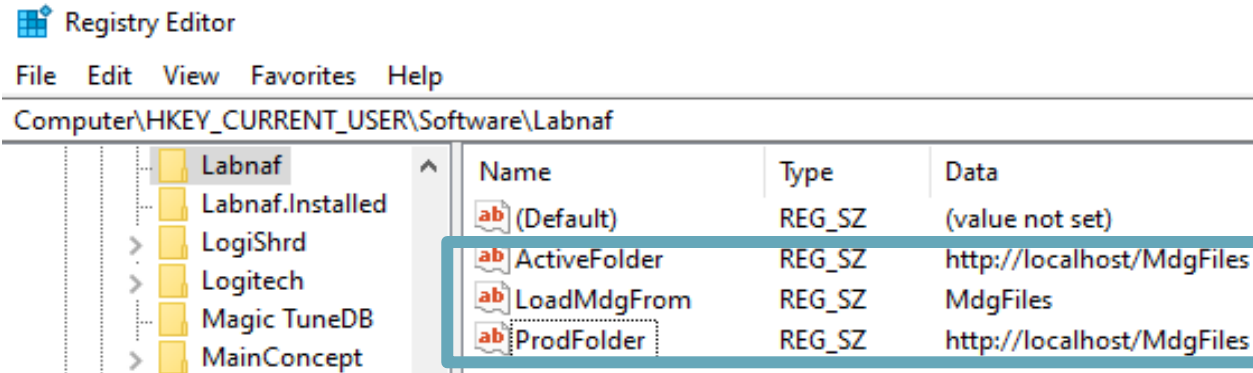
Deployment for end users of the Labnaf Addin

To deploy the customized version of the language on end users' desktops, the software distribution package must set some registry keys under

HKEY_CURRENT_USER\Software\Labnaf

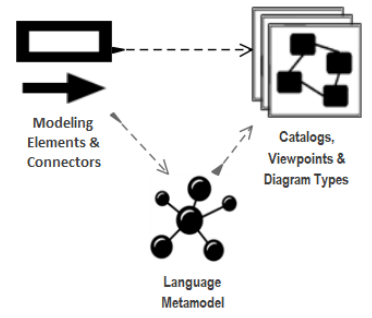
- **LoadMdgFrom** = MdgFiles
- **ActiveFolder** and **ProdFolder** point to the production folder (**web url** or **file system**)

For end users ActiveFolder and ProdFolder must have the same value.

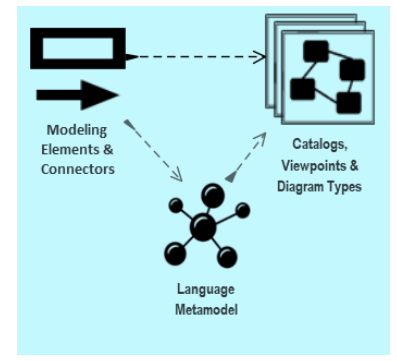


Merging your Labnaf customizations into a new version of Labnaf

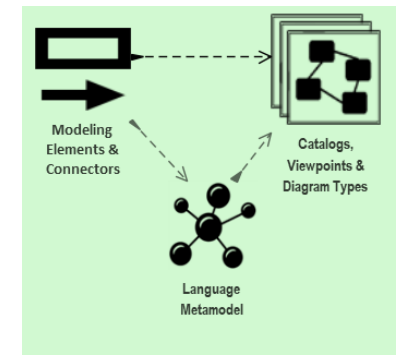
New tagged values, new elements types, new connector types



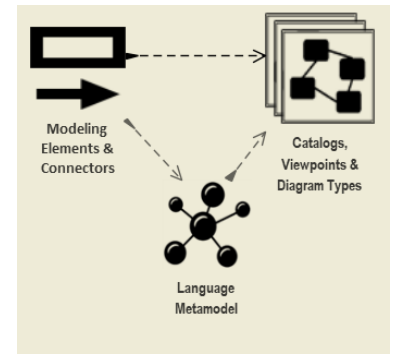
Labnaf Version X



Labnaf Version X + Customer Changes



Labnaf Version Y + Customer Changes

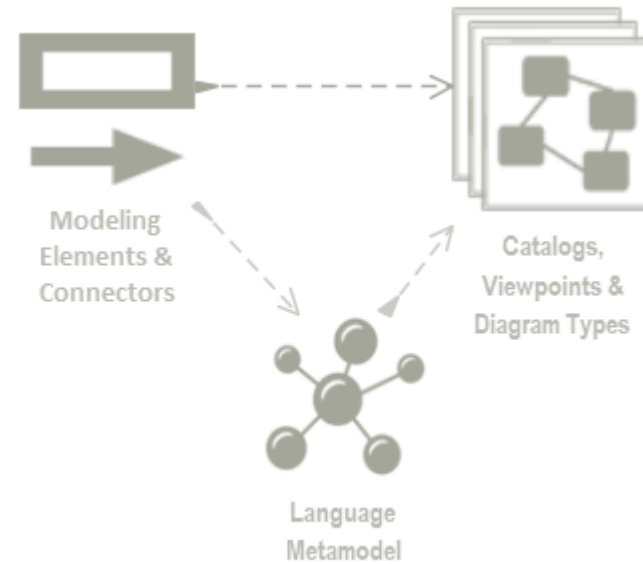


Labnaf Version Y

How to proceed

See also: Labnaf Language Transformer

1. Customize the language



2. Adapt existing repository content

