Unified Framework for Driving Transformations

Labnaf PowerShell

User Guide

Labnaf PowerShell Commands

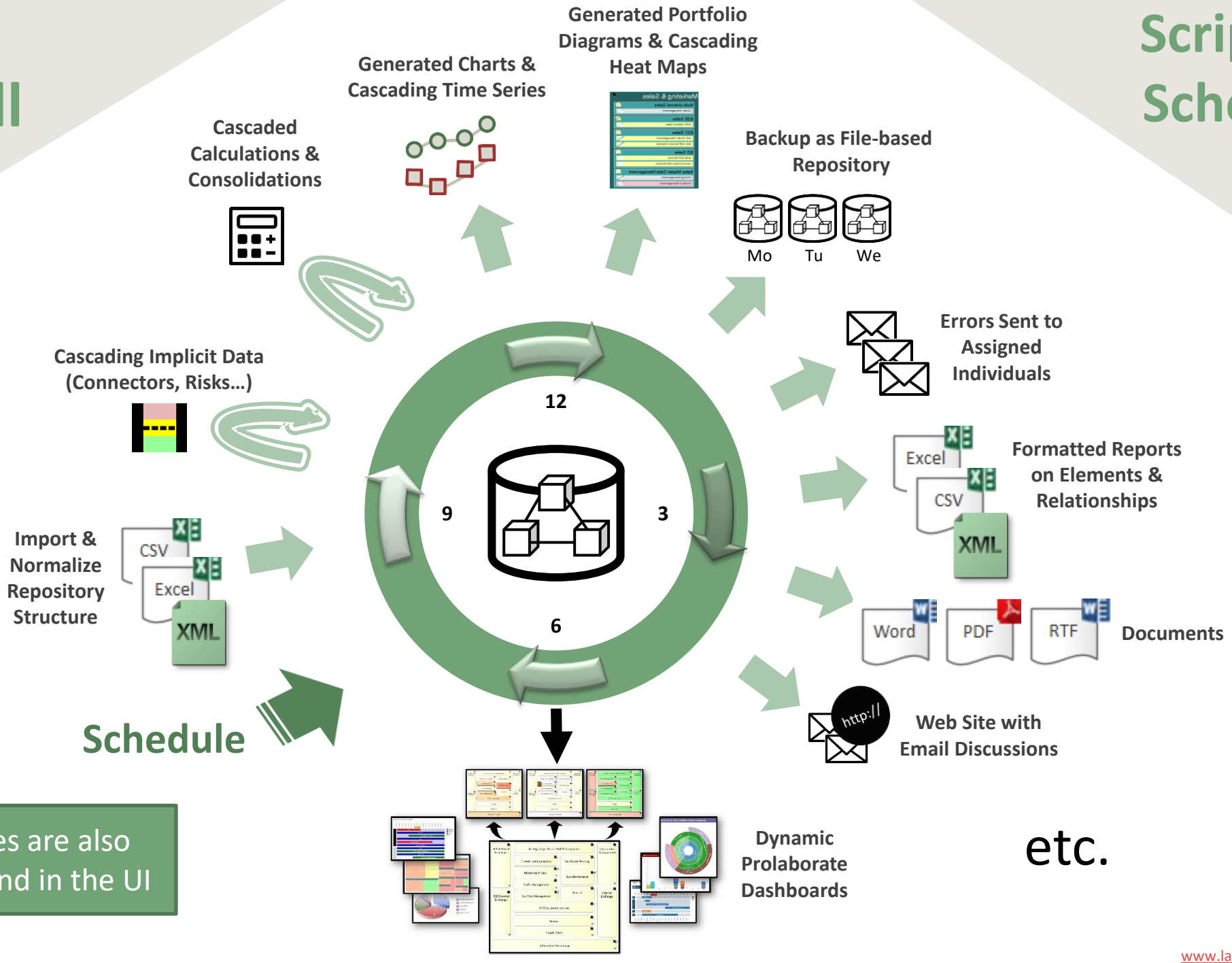
1. Overview
2. Strategy and Architecture Operations
3. Systems Integrations and Content Refactoring
4. Command Compatibility Matrix

Related resources (latest versions)

- **Labnaf PowerShell Reference Guide:** https://www.labnaf.one/EndUserMaterial/Labnaf_PowerShell/Labnaf%20PowerShell%20-%20Reference%20Guide.pdf
- **Labnaf On-line Guidance:** https://www.labnaf.one/guidance/index.html?guid=569FF62A-5210-4359-923F-4EB00EB03D61
- **Sample data:** Provided with the Labnaf PowerShell software

Labnaf PowerShell

Scripting & Scheduling



The same features are also available on demand in the UI

Running commands on the command line

lnps

Labnaf PowerShell Program

- "**LNPS**" is the name of the Labnaf PowerShell program.
- Full path is "**C:\Program Files (x86)\Labnaf\PowerShell\lnps.exe**"

```
Usage : "C:\Program Files (x86)\Labnaf\PowerShell\lnps.exe" [-]Command [arguments]

Available Commands: (Prefix the command name with '-' to run in non verbose mode)

   AutoConnectorsDelete

   AutoConnectorsGenerate

   BackupToAccessFile

   CalculateTaggedValues

   ClonePackage

   CreatePackage

   DeleteGeneratedCharts

   DeleteGeneratedDiagrams

   ExportToXmi

   GenerateCharts

   GenerateDiagrams

   GenerateDoc

   GenerateHTML

   GenerateImplicitData

   GenerateTabularReports

   ImportConnections

   ImportFromXmi

   ImportTabularReport

   MoveElementsToCalculatedParent

   MoveElementsToPackage

   MovePackagesToPackage

   RenameItem

   ScheduleCommand

   SetDiagramProperty

   SqlExportToCsv

   Validate

   ? => Show details for all commands
```

Running commands on the command line

lnps ?

Shows a detailed description of all commands and their usage

```
C:\Program Files (x86)\Labnaf\PowerShell>lnps ?

Command: AutoConnectorsDelete

Description: Delete generated connectors for child elements following defined element stereotype hierarchies.
Usage : lnps AutoConnectorsDelete [arguments]

Arguments:
   SourceRepoPathName: Path name of the source repository (EAP file).


Command: AutoConnectorsGenerate

Description: Generate connectors for child elements following defined element stereotype hierarchies.
Usage : lnps AutoConnectorsGenerate [arguments]

Arguments:
   SourceRepoPathName: Path name of the source repository (EAP file).


Command: BackupToAccessFile

Description: Backup a DBMS or Access repository to an Access Repository.
Usage : lnps BackupToAccessFile [arguments]

Arguments:
   SourceRepoPathName: Path name of the source repository (EAP file).
   DestEapPathName: Path name of the destination Access repository (EAP file).
   LogFilePath: Path name of the log file name.


Command: CalculateTaggedValues

Description: Calclute values for some defined tags and elements. The elements to be selected, the tags to be updated and the calculation formulas are all defined in the model repository
```

Running commands on the command line

lnps [-]{command name}

Example: if you type « lnps GenerateTabularReports » you get the following info.

```
Command: GenerateTabularReports

Description: Generate spreadsheets from a model repository based on configuration stored in that same repository.

Usage : lnps GenerateTabularReports [arguments]

Arguments:

   SourceRepoPathName: Path name of the source model repository (EAP file).

   OutputDirectoryPath: Directory path name where the spreadsheets must be generated.The name of each spreadsheet file is the name of the template report.

   [ElementPrototypeName]: The name of a specific element prototype name for which all embedded tabular report templates must be applied.

   [TabularReportTemplateName]: The name of a specific tabular report template to be applied.
```

Prefix the command name with '-' to run in non verbose mode

Running multiple commands from a Labnaf Power Script file (.lpsc)

lnps {LabnafPowerScriptFileName} .lpsc {RepoPathName} [SecondsBeforeRestart] [MaxMinutesRestartAfterFirstRun]

This executes commands from a '**{LabnafPowerScriptFileName}.lpsc**' file within a **single Sparx EA session**.

All commands refer to the **same repository {RepoPathName}** specified in the Windows command line,
so there's no need to include a repository argument inside the Labnaf script file itself.

Sample Labnaf Power Script:

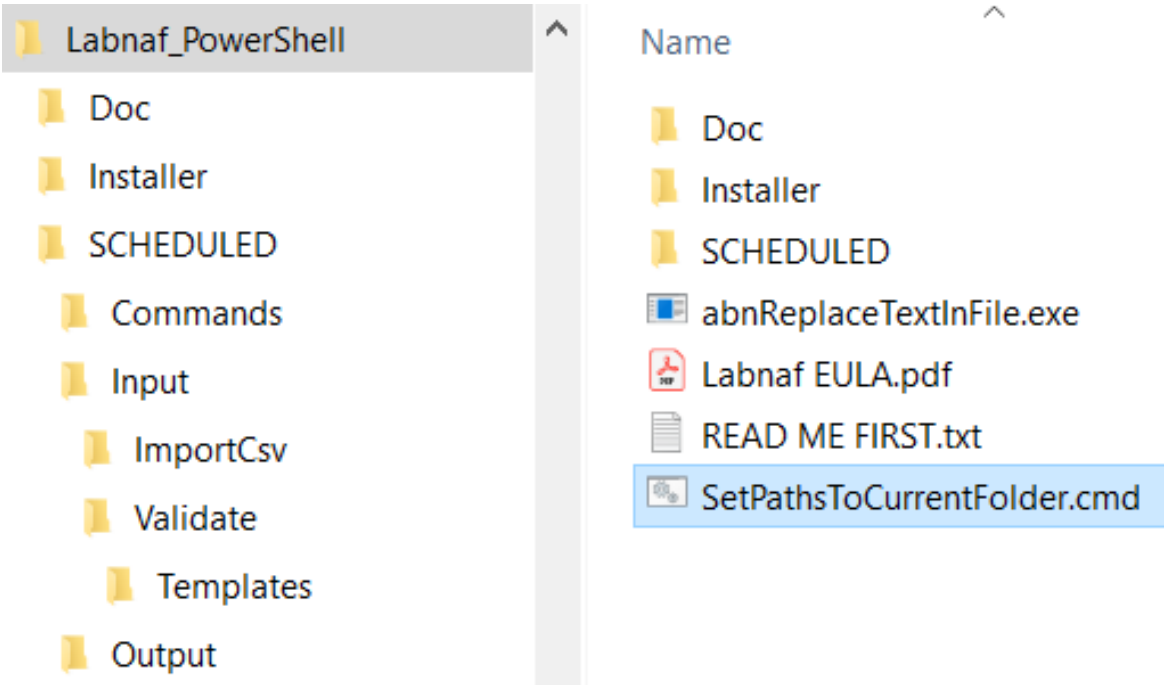
```
// Script files can include environment variables and comments. Same for the SQL queries used by the PowerScript commands.
GenerateTabularReports %OUTPUT_DIR% Capability    /* OUTPUT_DIR is a Windows environment variable that was set earlier */
GenerateTabularReports %OUTPUT_DIR% LABN_Application "Application List[CP=utf-8].CSV“
```

If specified as lnps arguments, the script will run iteratively:
- Resumes after **{SecondsBeforeRestart}**
- Concludes after **{MaxMinutesRestartAfterFirstRun}**

If only {SecondsBeforeRestart} is provided, the script continues until halted by the admin via the ESC key in the console window.

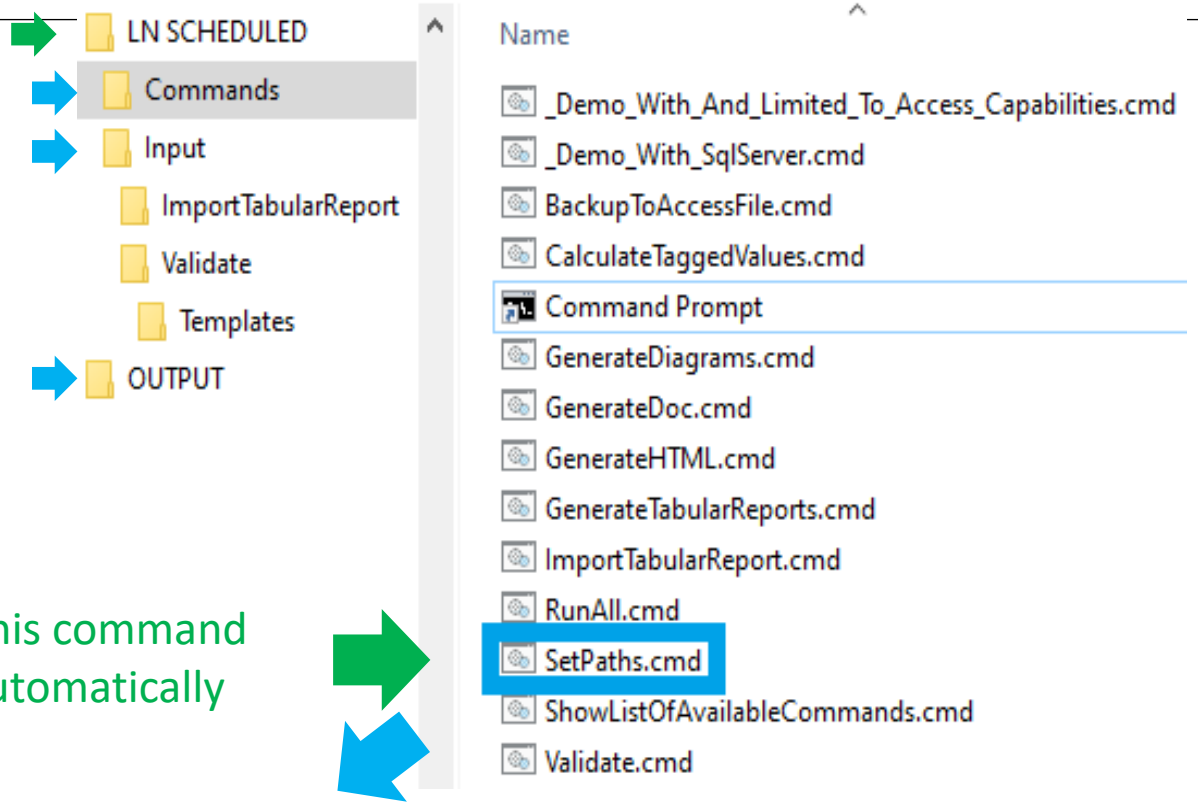
Automatic configuration

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



This updates the Labnaf PowerShell configuration files following the “Labnaf_PowerShell” folder location.

Preconfigured batches calling commands with predefined settings

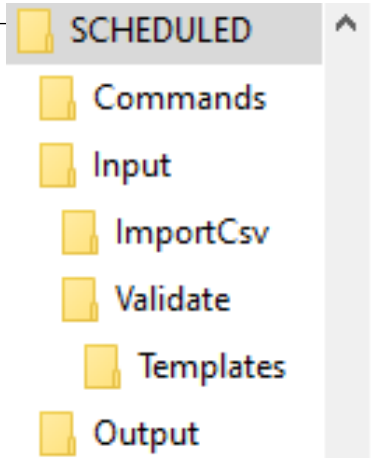


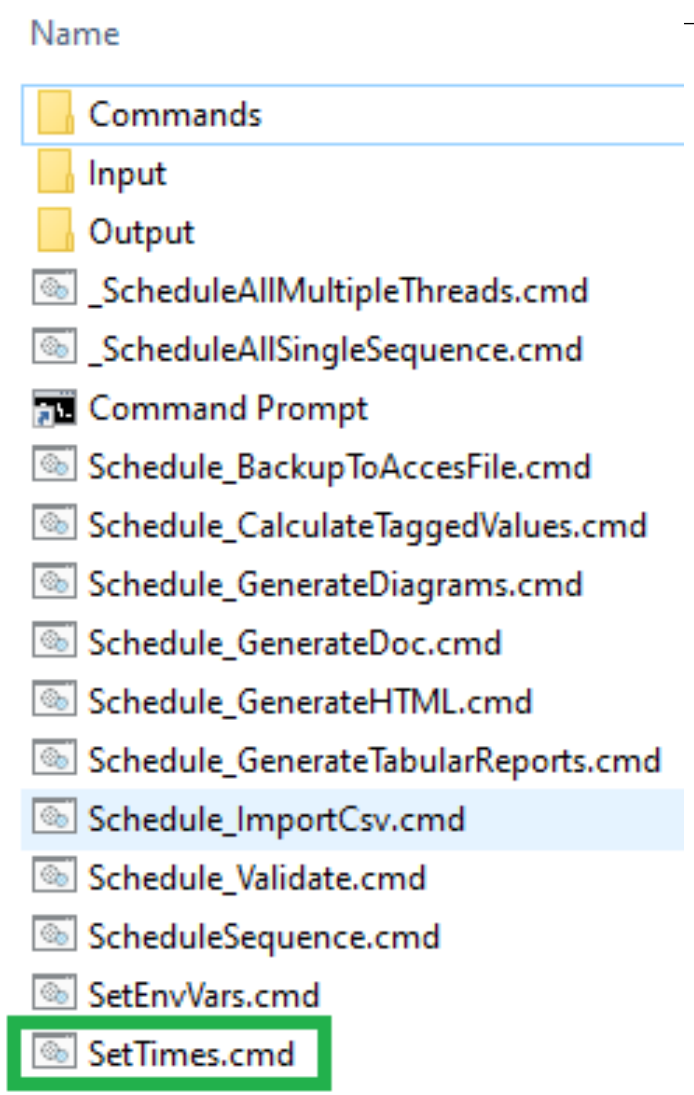
The paths in this command were reset automatically

SetPaths.cmd

```
set LABNAF_POWERSHELL=C:\Program Files (x86)\Labnaf\PowerShell\lnps.exe

set SCHEDULED_DIR=C:\Users\User\Desktop\Labnaf_PowerShell\SCHEDULED
set COMMANDS_DIR=%SCHEDULED_DIR%\Commands

set INPUT_DIR=%SCHEDULED_DIR%\Input
set OUTPUT_DIR=C:\Users\User\Desktop\Labnaf_PowerShell\SCHEDULED\Output

set REPOSITORY=%INPUT_DIR%\Repository.eap
```

Preconfigured Command Scheduling





SetTimes.cmd

```
REM -- SINGLE START TIME --
Set StartTime_AllSingleSequence=00:00:00


REM -- SPECIFIC START TIME FOR EACH TASK --

Set StartTime_Cleanup_BackupToAccesFile=22:00:00
Set StartTime_Cleanup_GenerateHTML=22:00:05

Set StartTime_ImportCsv=22:30:00

Set StartTime_CalculateTaggedValues=23:00:00
Set StartTime_GenerateDiagrams=23:30:00

Set StartTime_BackupToAccessFile=00:00:00
Set StartTime_Validate=01:00:00

Set StartTime_GenerateTabularReports=02:00:00
Set StartTime_GenerateDoc=02:30:00
Set StartTime_GenerateHTML=03:00:00


REM --------------------------

set SCHEDULED_MINUTES_UNTIL_RESTART=1440
```

Labnaf PowerShell Commands

1. Overview
2. Strategy and Architecture Operations
3. Systems Integrations and Content Refactoring
4. Command Compatibility Matrix

Labnaf PowerShell commands for **Strategy and architecture operations**

- Import Tabular Report (Excel, CSV)
- Auto Connectors Generate / Delete
- Generate Implicit Data
- Calculate Values
- Validate and send emails to assigned individuals
- Generate Charts / Delete Generated Charts
- Generate Diagrams
- Generate Tabular Report (Excel, CSV), Doc (Word, RTF, PDF), Html
- Backup To File
- Schedule Command (not only Labnaf PowerShell commands)

ImportTabularReport

Sample input data for updating tagged values of existing elements

File to be imported can be .CSV or .XLSX (Excel)

	A	B	C	D	E
1	**guid**	**application_owner**	**application_owner_delegates**	**it_responsible_service**	**legal_owner**
2	{D303A068-2CAA-438d-9E81-287EE9777F1D}	homer.simpson@labnaf.local		Microsoft development	Labnaf
3	{305AA65E-A3F8-435b-81EC-C22EB7DF01C4}	marge.simpson@labnaf.local	lisa.simpson@labnaf.local	Enterprise Architecture	Labnaf
4	{07F7FA8B-A01C-4aed-B5C2-80C9D62BD3FF}	bart.simpson@labnaf.local		SAP development	Labnaf

OPTIONAL repository column mappings are stored in a .CSV file

	A	B
1	**Input_Column_Names**	**Target_Column_Names**
2	guid	ea_guid
3	application_owner	IT_Contact
4	application_owner_delegates	IT_Contact_Delegates
5	it_responsible_service	IT_ResponsibleService
6	legal_owner	Legal_Owner

Import Tabular Report (cont.)

To start the import:

```
C:\Program Files (x86)\Labnaf\PowerShell>lnps ImportTabularReport

Command: ImportTabularReport

Description: Import elements, properties and tagged values from a CSV or Excel file into a SQL Server database.

        If a field name mapping file (CSV) is provided, the first line must contain the following headers:
                Input_Column_Names, Target_Column_Names
        Works also with Access databases but only for updating existing elements.

        Identify existing elements using
           either the ea_guid,
           or an alternative unique key.
           To define a unique key, you simply add a '#' in front of the property or tag name
           If multiple unique keys are provided, they are searched in this order: Tagged Value, Name, Alias.

        Create the element if the element is not found and if 'EnableCreate' option is present on the command line.
        The package where new elements are added is identified by a package guid provided on the command line
        Initial value calculation applies to any imported new element.


Usage : lnps ImportTabularReport [arguments]

Arguments:

   RepoPathName: Path name of the model repository (EAP file).

   SourceFile: A CSV or Excel file containing the data that needs to be imported.

   ColumnMappingFile: An optional CSV file containing the mapping between the input and output column namesor '-' if all input and output column names are the same.

   ElementStereotypeName: The stereotype of the elements that need to be updated.

   [EnableCreate]: Enable creation of new elements if they are not found.

   [TargetPackageGuidForNewElements]: The package where the new elements must be stored.
```

Import Tabular Report (cont.)

Beware of the different definitions of the “Notes” / “Note” property name in Sparx EA.

In the Sparx EA API, the property name is called “Notes”.

In the database, the corresponding column name is called “Note”.

For Imports

- In the import file use the word « **Notes** » (and not « Note »)

For SELECT statements

- In SQL Server, use the word « Note » in queries.

  Example: `select Name, Note from t_Object`

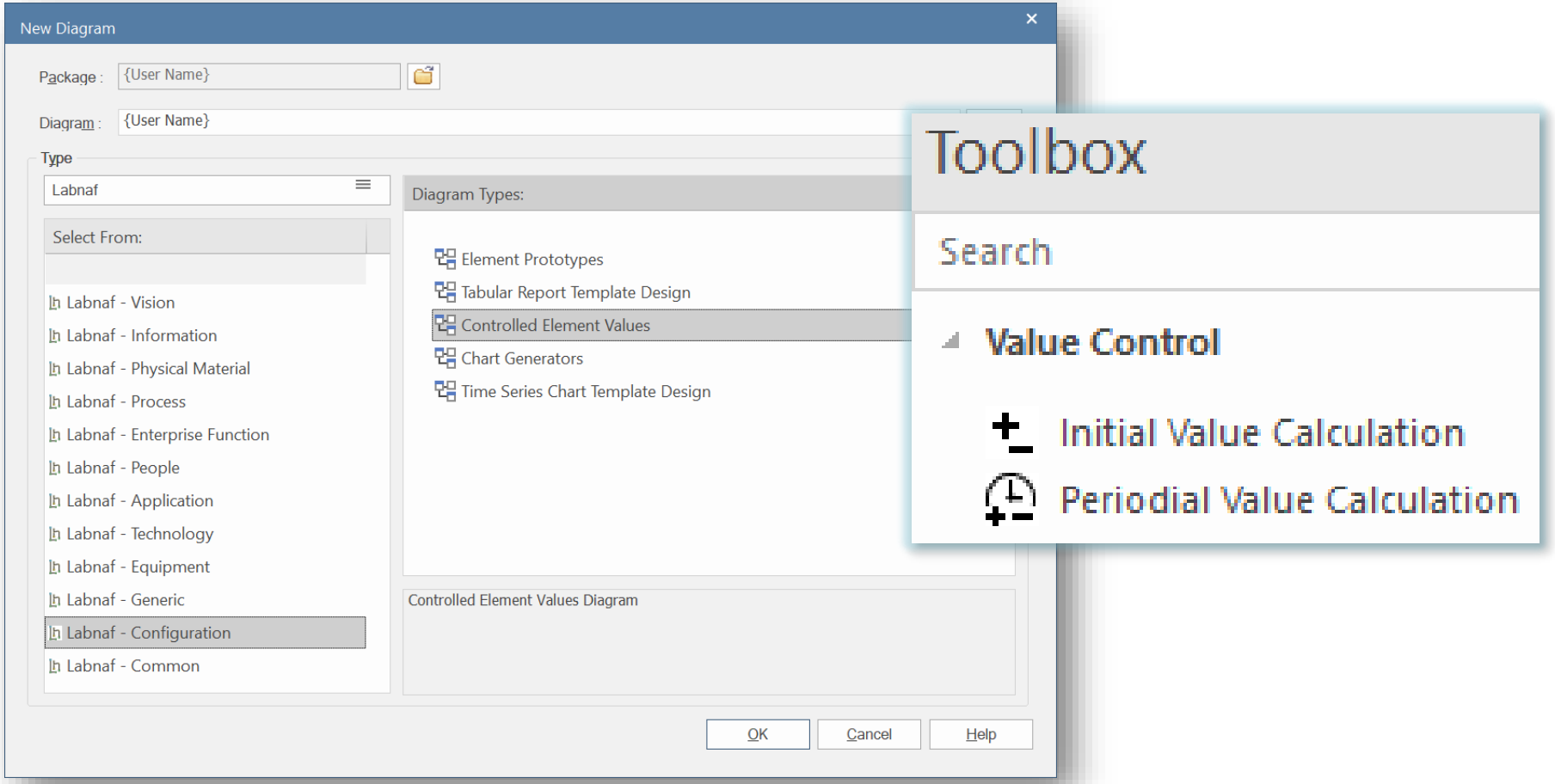
- In the Sparx EA user interface, use « Note as {Something} ». Indeed « Note » is a reserved word.

  Example: `select Name, Note as Element_Notes from t_Object`

CalculateTaggedValues

You first need to model your value calculations

See the **Value Calculation** User Guide document and the examples on the Guidance Web Site



Calculate Tagged Values (cont.)

To start calculation:

```
Command: CalculateTaggedValues

Description: Calclute values for some defined tags and elements.The elements to be selected, the tags to
 be updated and the calculation formulas are all defined in the model repository.

Usage : lnps CalculateTaggedValues [arguments]

Arguments:

   RepoPathName: Repository path name EAP file).

   [ElementPrototypeName]: A specific element stereotype for which tagged values must be calculated.

   [TagName]: The name of aspecific tagged value that must be calculated.
```

AutoConnectorsGenerate

To start implicit connector generation:

```
Command: AutoConnectorsGenerate

Description: Generate implicit connectors following numerous patterns and options, including:
- for child elements following defined element stereotype hierarchies (this option is enabled by default)
- for information elements (entities) used, owned, aggregated or exchanged directly or indirectly by any other element.
See 'Implicit Connectors' on the Labnaf guidance web site to learn more about enabling and disabling options.

Usage : lnps AutoConnectorsGenerate [arguments]

Arguments:

   SourceRepoPathName: Path name of the source repository (EAP file).
```

The generated implicit connectors are aggregations.

See the different implicit connector generation options on the guidance web site.

Benefits of implicit connector generation

- Know which information is used by whom, by which role, organization, function, process, activity, data flow, application, component, data store, server, equipment, network, etc
- Know which information is stored where (for example in which country)
- Dramatically simplifies information security and GDPR compliance analysis
- Simplifies traceability as embedded elements get aggregation connectors
- Normalizes the way elements are related in a repository i.e. based on connectors
- Enables powerful reporting capabilities including Prolaborate charts

AutoConnectorsDelete

To start deletion of generated connectors:

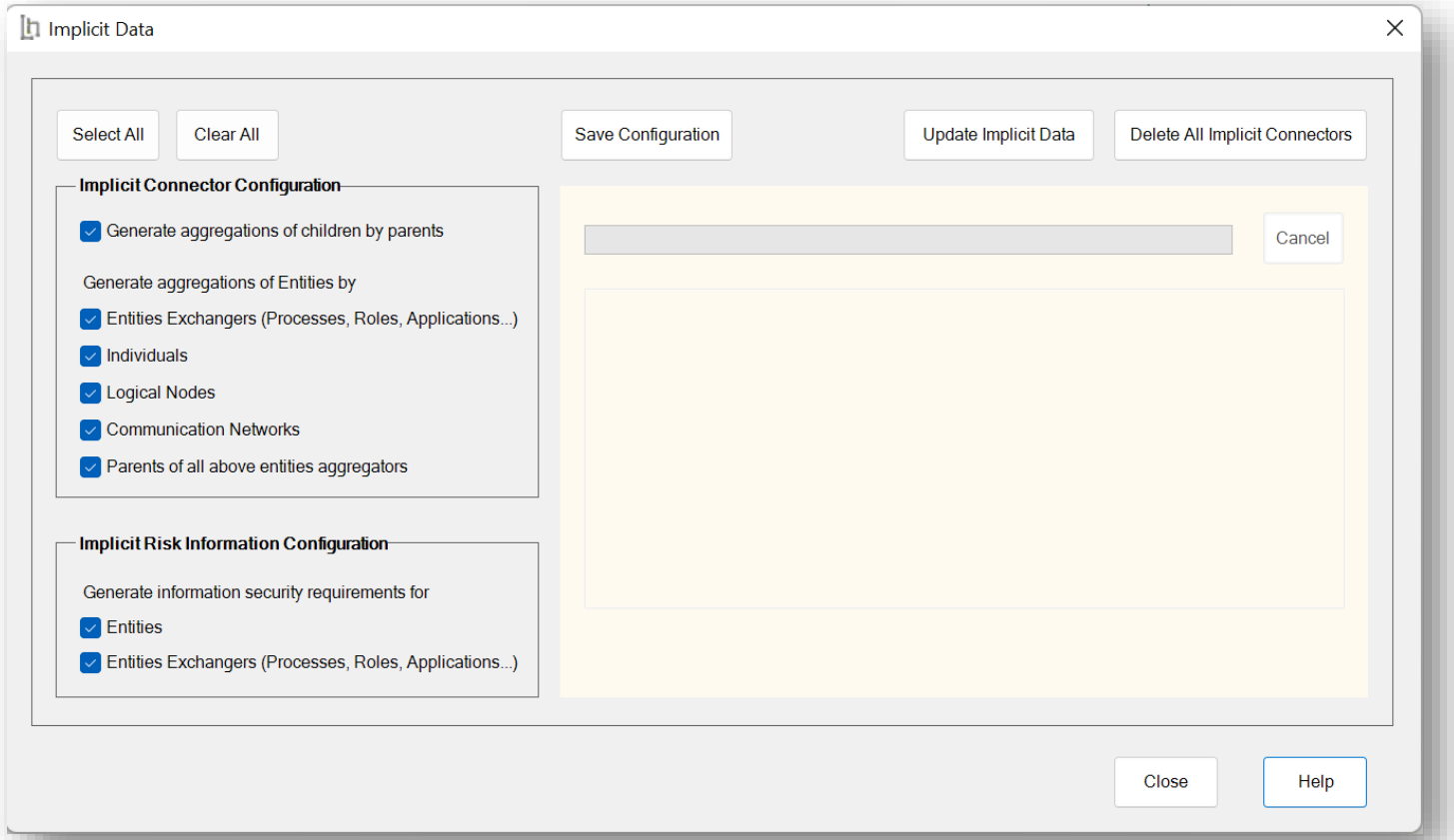
```
Command: AutoConnectorsDelete

Description: Delete generated connectors for child elements following defined element stereotype hierarchies.

Usage : lnps AutoConnectorsDelete [arguments]

Arguments:

   SourceRepoPathName: Path name of the source repository (EAP file).
```

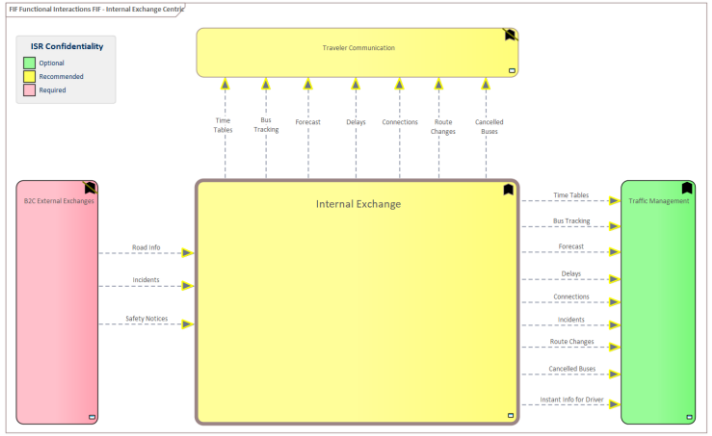
The generated connectors are aggregations.

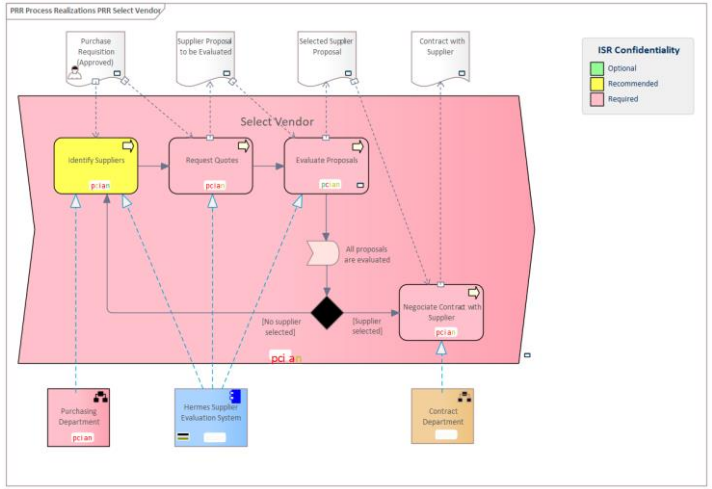
GenerateImplicitData

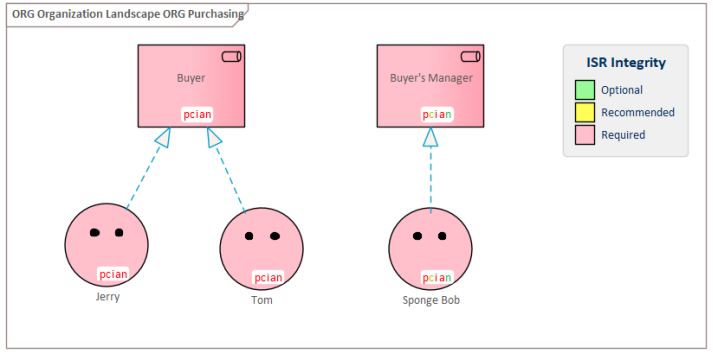
Implicit Data Generation is configured using Labnaf AddIn user interface

See **Implicit Data Generation** on the Guidance Web Site



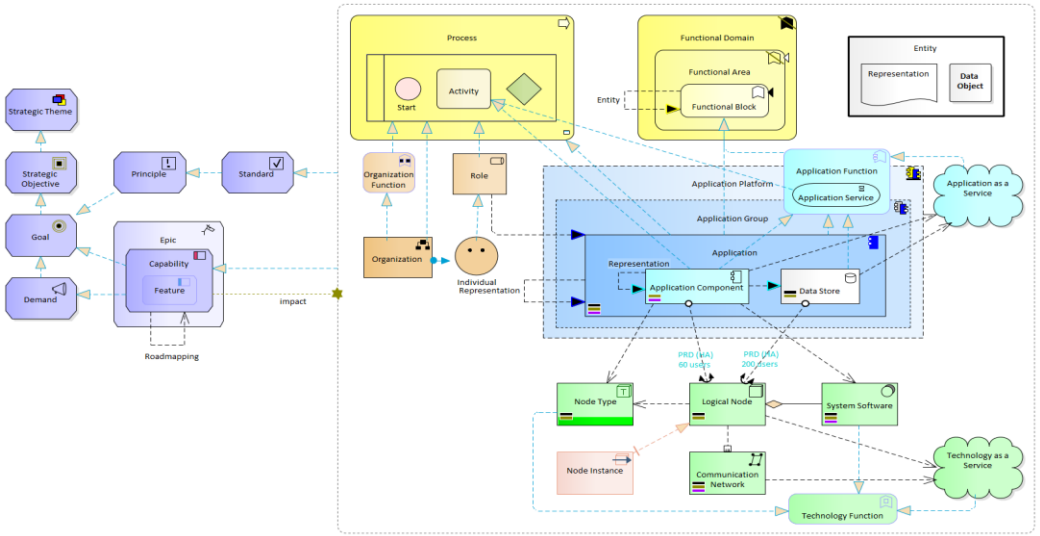


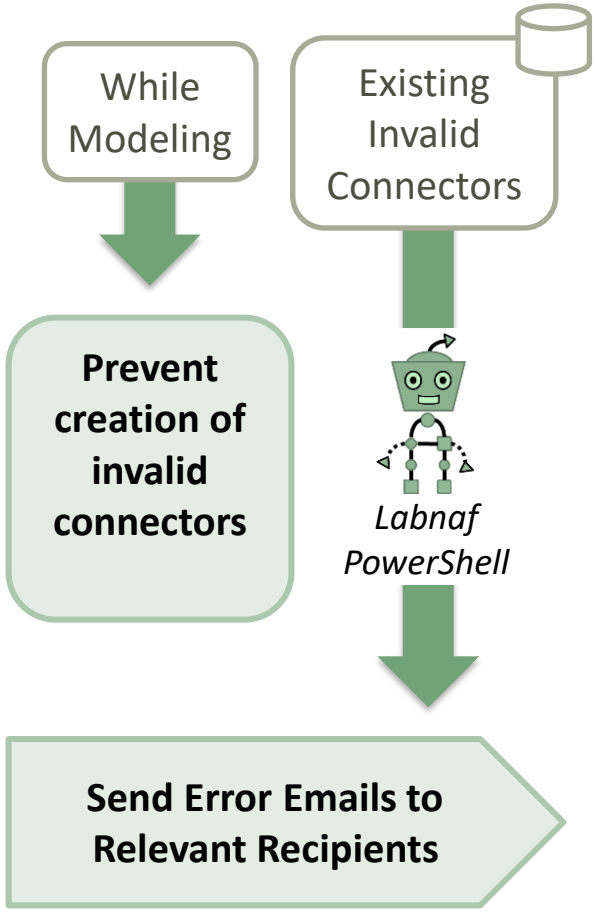




Validate

The Language Metamodel is used both for documentation & automatic model validation





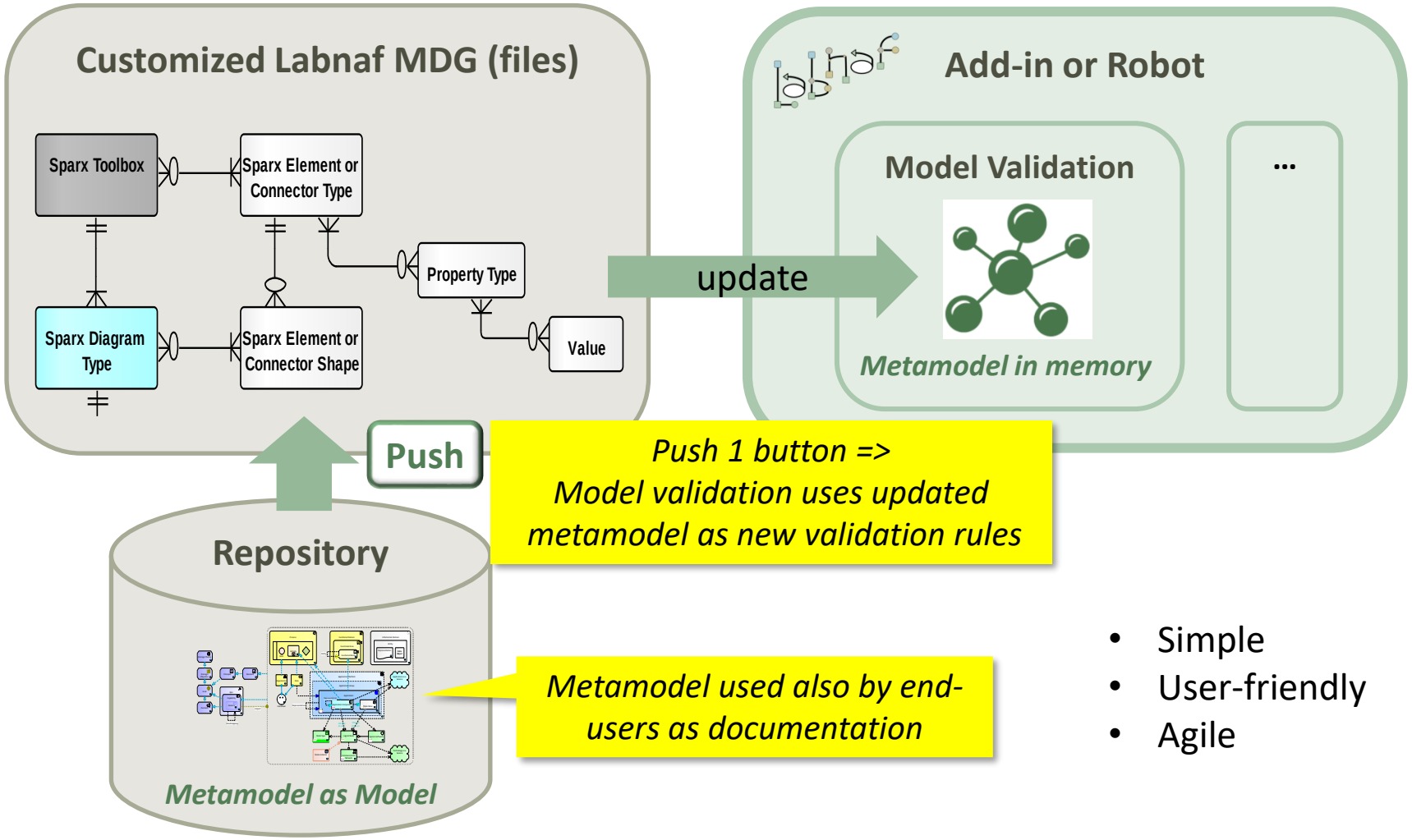
Why do we need periodical validation?

How could we have errors if we prevent users from entering errors?

- At the beginning, informal (invalid) models can be imported and their language can be transformed e.g. from ArchiMate to Labnaf.

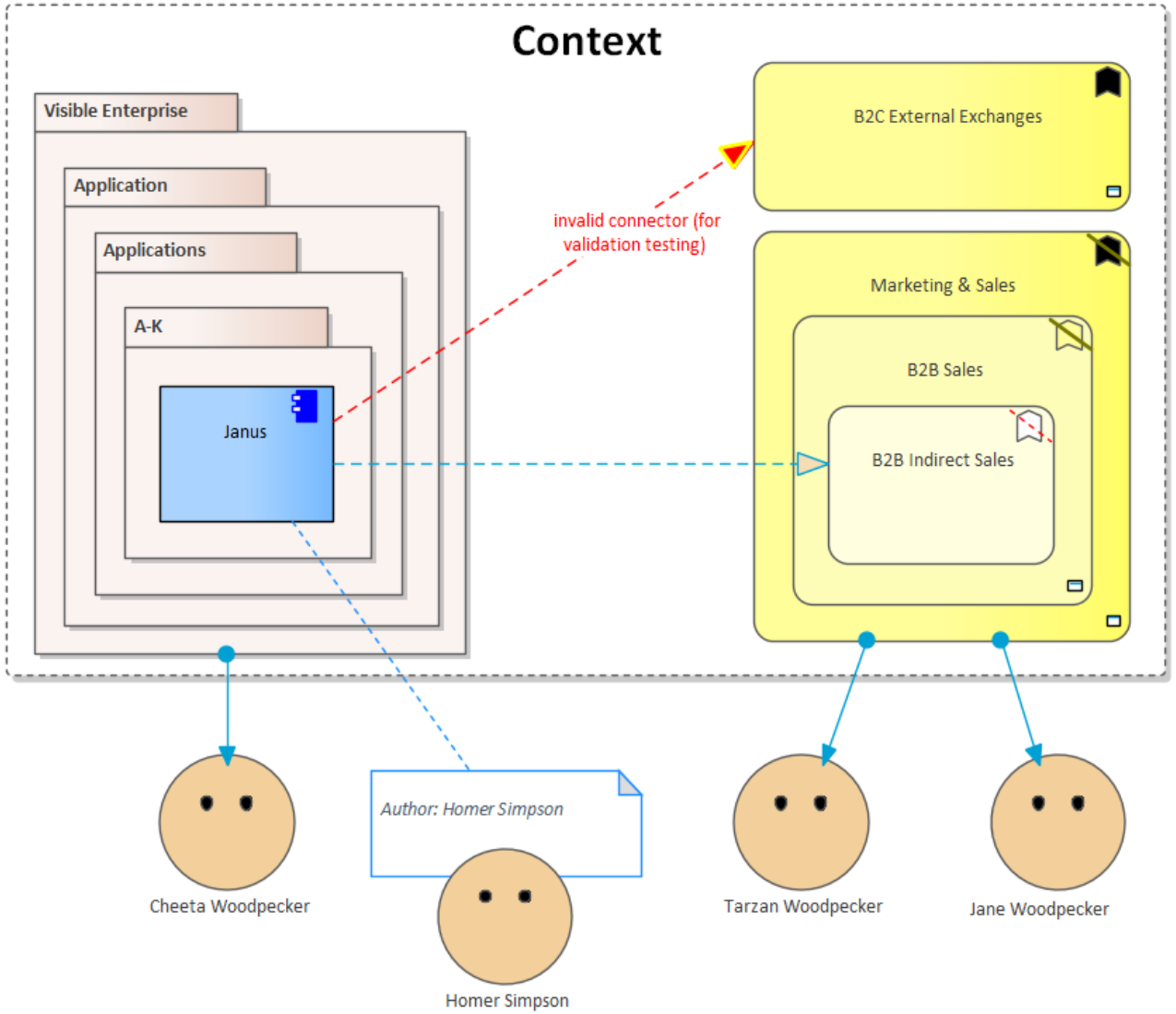
- Then, every time you update the metamodel to adapt to your enterprise specificities, some existing model repository content becomes invalid… according to your new rules.

The default metamodel can be **very** easily updated: One click on a connection in your production repository.



- Simple
- User-friendly
- Agile

Error message routing is based on architecture management assignments

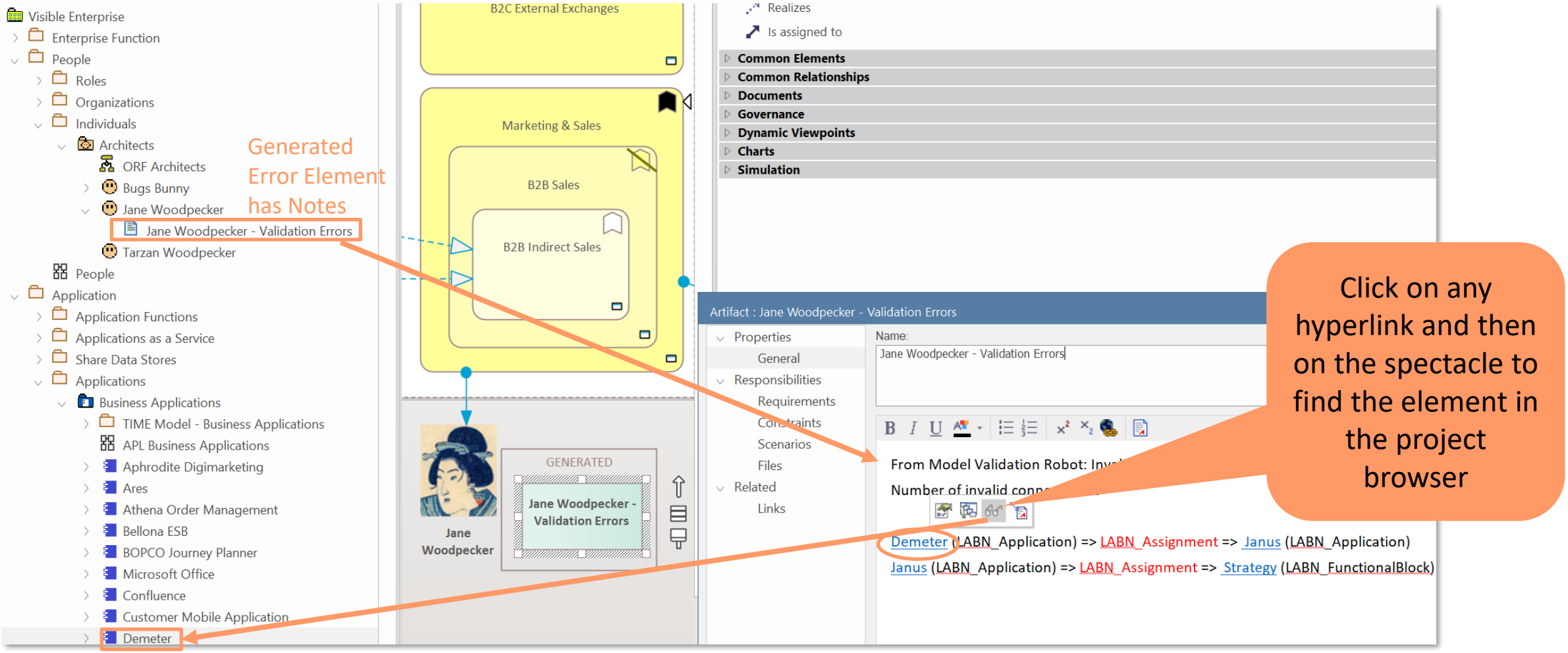


Error message distribution depends on

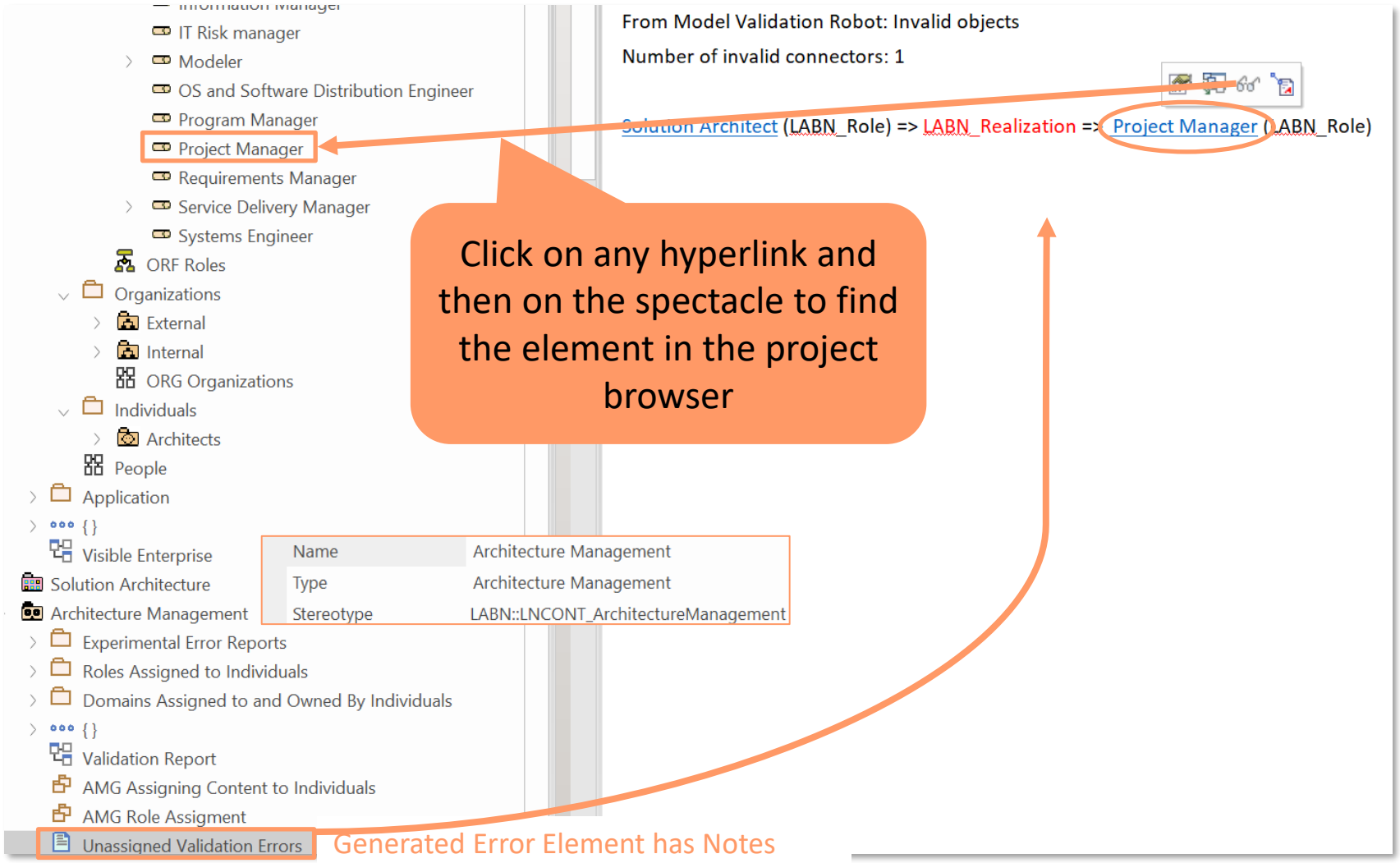
- recent authors
- individuals assigned to packages
- individuals assigned to domains
- default recipients
- and rules that you can use to combine these different combinations

The catalog of individuals **includes email addresses**

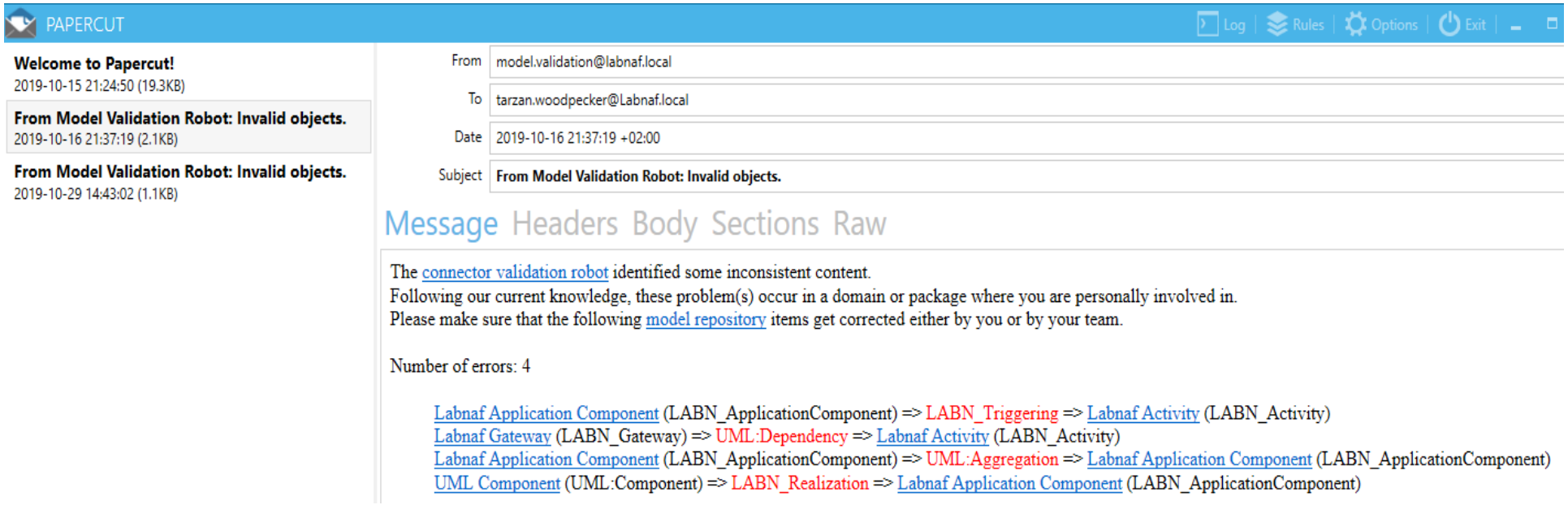
Error messages are generated as an error element with notes under each assigned individual **in the model repository**



If the error cannot be assigned to any specific individual, then the error element is generated under the unique package of type “Architecture Management”



Error messages are also sent to assigned individuals **by email,** if email is configured



Validation rules can be further customized

```
<ValidationConfiguration xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd=
"http://www.w3.org/2001/XMLSchema">
  <SelectElementsInScope>
SELECT * FROM t_object WHERE stereotype like 'LABN_%' AND Package_ID IN (SELECT PDATA1
from t_object where Object_Type='Package' and Stereotype like 'LNCAT_%') ORDER BY Name
  </SelectElementsInScope>
  <Sender>model.validation@labnafdemo.com</Sender>
  <SendTo>
    <FirstAvailableAlternativeOnly>true</FirstAvailableAlternativeOnly>
    <PeopleAssignedToDomain>true</PeopleAssignedToDomain>
    <AuthorDuringMonthsAfterElementCreated>120</AuthorDuringMonthsAfterElementCreated>
    <PeopleAssignedToPackage>true</PeopleAssignedToPackage>
    <DefaultEmailAddresses>lisa.simpson@labnaf.local</DefaultEmailAddresses>
  </SendTo>
  <PublishedRepositoryWebSiteUrl>http://localhost/guidance</PublishedRepositoryWebSiteUrl>
  <DocumentationReferences>
    <GuidanceWebSiteUrl>http://www.Labnaf.one/guidance</GuidanceWebSiteUrl>
    <DiagramGuids>
      <ConnectorValidation>{269E2D0C-3B9E-4d85-915A-87905EB7271F}</ConnectorValidation>
      <ModelRepository>{EF41E336-AC6B-4407-88D9-3ECC41725132}</ModelRepository>
    </DiagramGuids>
  </DocumentationReferences>
</ValidationConfiguration>
```

If you want to be specific about the elements to be validated. By default all Labnaf elements are validated.

Error messages are sent from this email address.

Who will receive the error messages.

Error messages contains urls to invalid elements. HTML publication should be scheduled as well

Smtp Server configuration is straightforward

```
<?xml version="1.0" encoding="utf-8"?>
<SmtpServerConfiguration xmlns:xsi="http://ww
  <Host>127.0.0.1</Host>
  <ClientPort>25</ClientPort>
  <EnableSSL>false</EnableSSL>
  <UserName>alain@labnafdemo.com</UserName>
  <Password></Password>
</SmtpServerConfiguration>
```

Needed to send error messages to assigned individuals

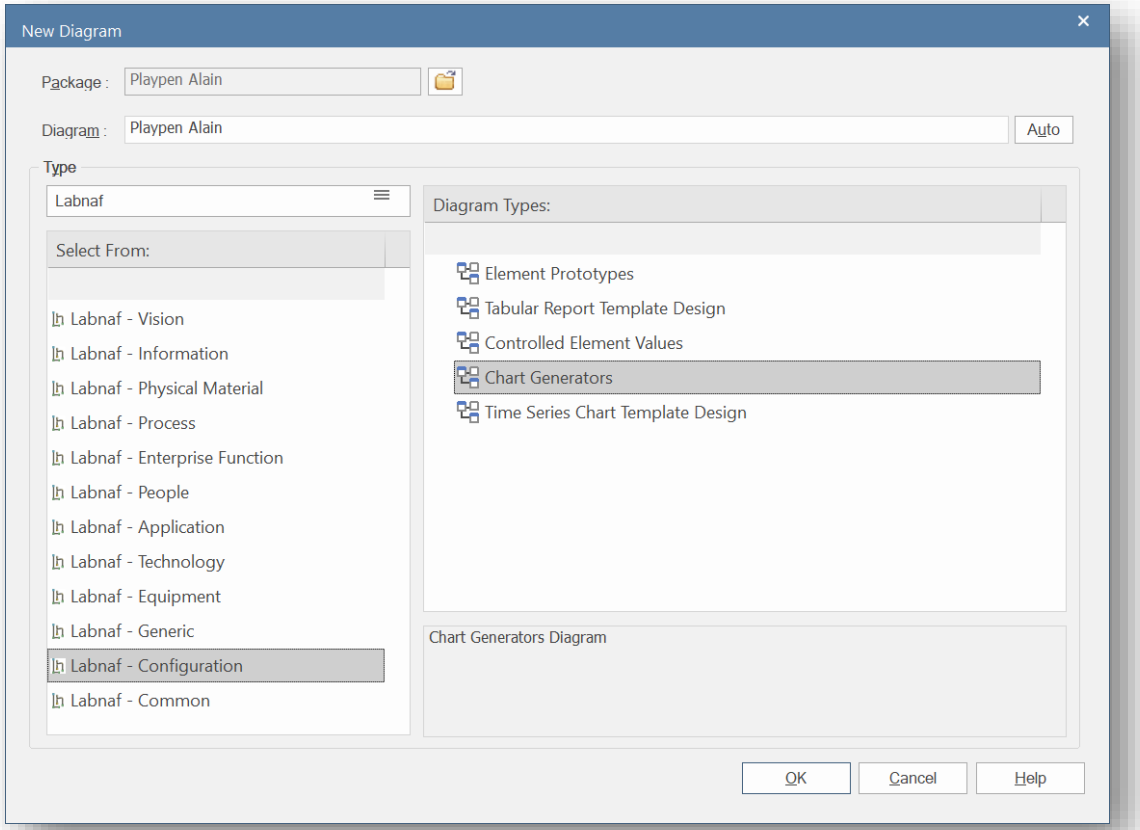
To start validation:

```
C:\A\LT\SparxDev\VSProjects\Alain\LabNaf\PowerShell\bin\Release\Dotfuscated>lnps validate

Command: Validate

Description: Validate model repository.

Usage : lnps Validate [arguments]

Arguments:

   RepoPathName: Repository path name (EAP file).

   ValidationConfigurationFile: Path name of the  model validation configuration file.

   SmtpServerConfigurationFile: Path name of the  SMTP Server configuration file.
```

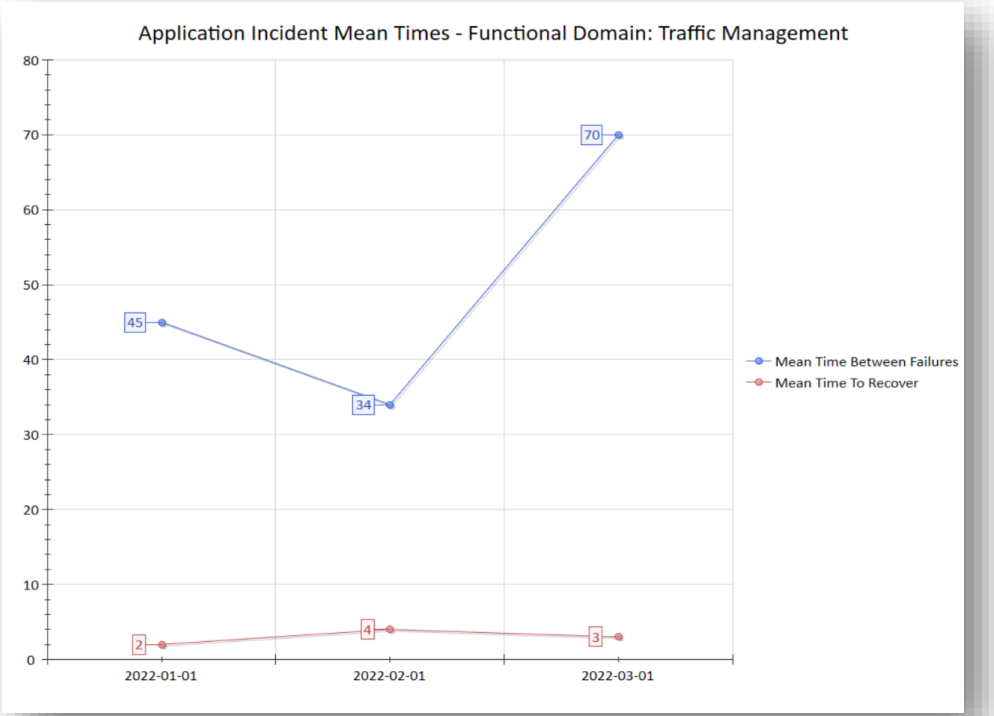
GenerateCharts

Labnaf comes with predefined chart templates, but you might want model your own chart templates

See **Chart Generation** on the Guidance Web Site



Sample Generated Time Series Chart



Generate Charts (cont.)

To start chart generation:

```
Command: GenerateCharts

Description: Generate charts for some configured types of element.
        The elements to be selected and the charts to be generated are all defined in the model repository.
        Chart Generator elements that have a name starting with "--" will be ignored.
Usage : "C:\Program Files (x86)\Labnaf\PowerShell\lnps.exe" GenerateCharts [arguments]

Arguments:
   RepoPathName: Repository path name (EAP file).
   [ElementPrototypeName]: A specific element stereotype for which charts must be generated.
   [TemplateChart]: The template chart name defining the type of chart diagram and chart element to be generated.
```

To delete all generated charts:

```
Command: DeleteGeneratedCharts

Description: Delete generated charts.
Usage : "C:\Program Files (x86)\Labnaf\PowerShell\lnps.exe" DeleteGeneratedCharts [arguments]

Arguments:
   SourceRepoPathName: Path name of the source repository (EAP file).
```

GenerateDiagrams

Sample Results

Enterprise Function Taxonomy & Applications Supporting Level 1

List of applications supporting the domain.

'Application Name'
Ares
Athena Cash Desk
Customer Mobile Application
Demeter
Hera
Janus
Jupiter Cash Desk
LOGIN B2B
Neptune
Venus Cash Desk
Zeus Pricing
Zeus Sales Records Management

Legend for Business Functions

Differentiator = ?	
= Y	
= N	

Externalized = Y	Entirely externalized
= P	Partially externalized
= N	Not externalized (default)

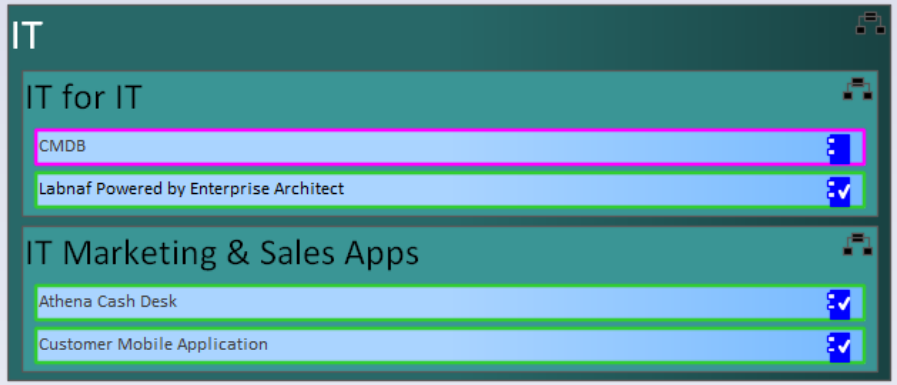
Nb of Applications / Business function

- 0
- 1
- 2 or more

Other diagram(s) for this enterprise function:

FULSales

Applications managed by organizations



Application Lifecycle / Vision

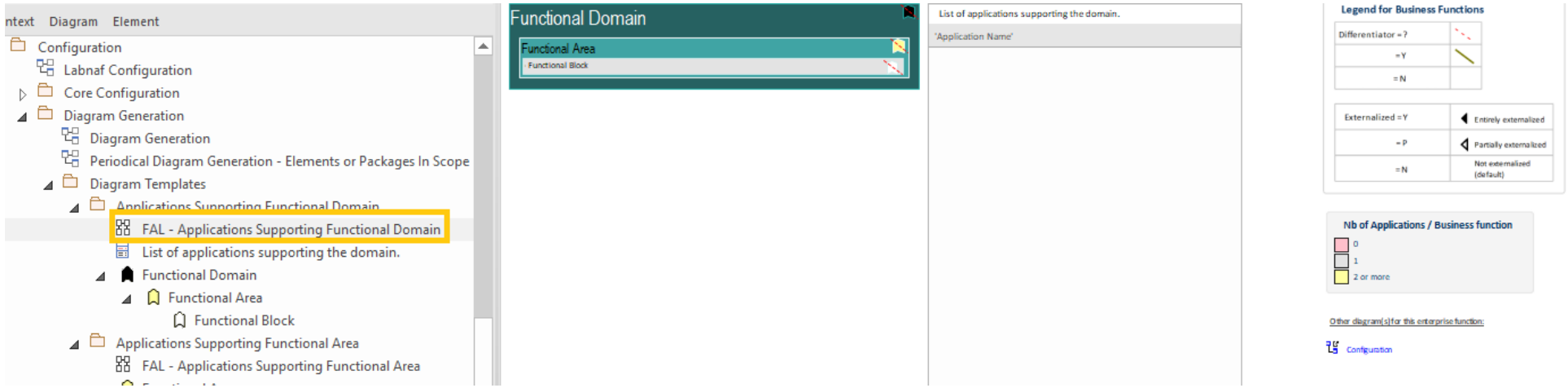
- New
- Invest
- Maintain
- Phase Out
- ?

Other diagram(s) for this organization:

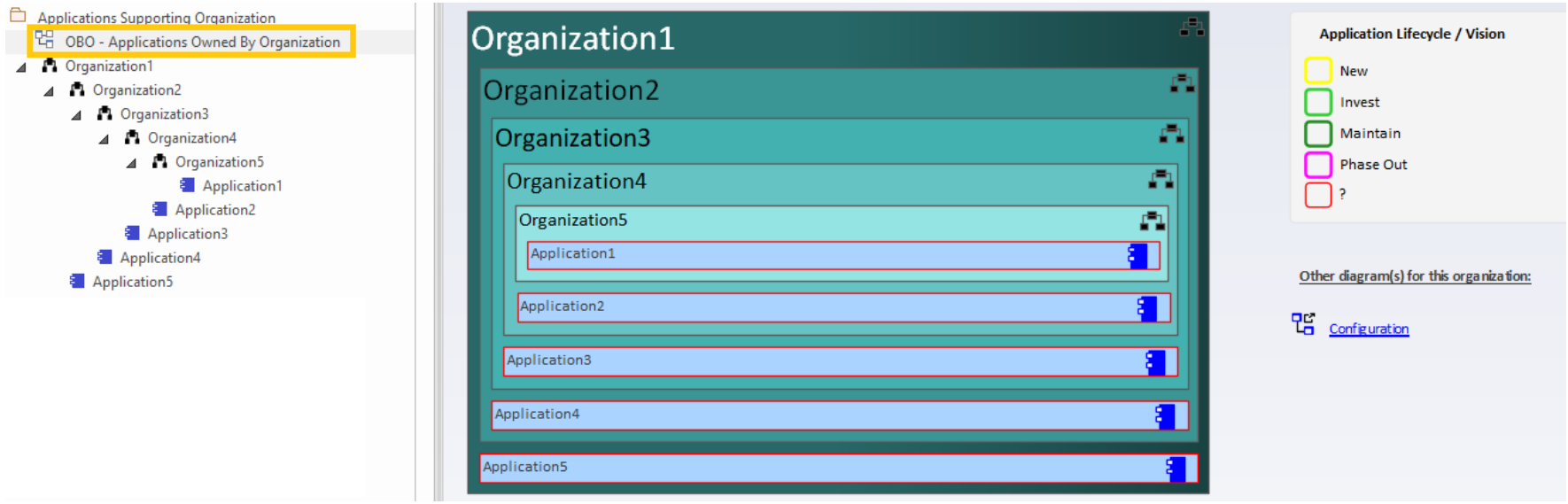
Configuration

Sample Diagram Templates

Enterprise Function Taxonomy & Applications Supporting Level 1

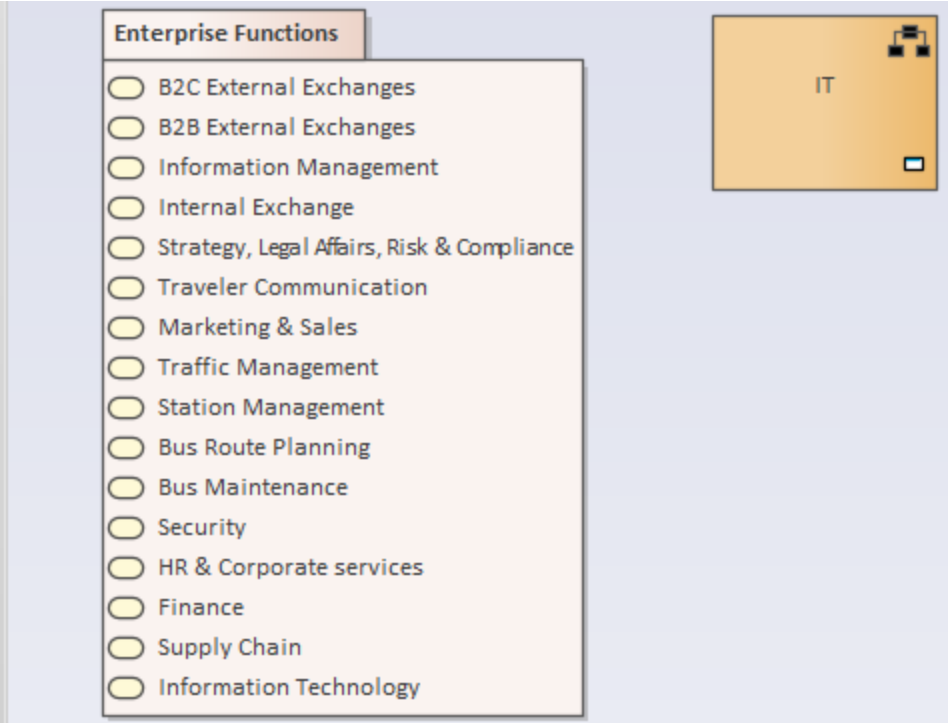


Applications managed by organizations



Items in Scope for Generation

- Configuration
  - Labnaf Configuration
  - Core Configuration
  - Diagram Generation
    - Diagram Generation
    - Periodical Diagram Generation - Elements or Packages In Scope
    - Diagram Templates
      - Applications Supporting Functional Domain
        - FAL - Applications Supporting Functional Domain
        - List of applications supporting the domain.
        - Functional Domain
          - Functional Area
            - Functional Block
      - Applications Supporting Functional Area
        - FAL - Applications Supporting Functional Area
        - Functional Area
          - Application2
          - Functional Block
            - Application1
      - Applications Supporting Organization
        - OBO - Applications Owned By Organization
        - Organization1
          - Organization2
            - Organization3
              - Organization4
                - Organization5
                - Application1
                - Application2
              - Application3
              - Application4
            - Application5



This is optional

By default, all elements are selected from the related catalog

Generate Diagrams (cont.)

To start diagram generation:

```
Command: GenerateDiagrams

Description: Generate diagrams in a model repository.

Usage : lnps GenerateDiagrams [arguments]

Arguments:

   RepoPathName: Path name (EAP file) of the repository where the diagrams must be generated.

   GenerationScopeDiagramGUID: A diagram containing organizations elements and/or a package of
enterprise functions for which diagram generation is required.
```

GenerateTabularReports

Sample Result

Reported collection of elements (e.g. applications) selected following any kind of rule

Element properties and/or tagged values with optional renaming and coloring

Specific connections in specific direction to specific types of elements.
Automatic connection **consolidation** into parent element relationships.

Properties

Relationships

Overview:	Guid	AppNo	Name	Notes	IT_Contact	IT_Contact_Delegates	Value:	Criticality	Functional_Fit	Technical_Fit	Scores:	TCO	Doc_Amount	Nb_Supported_FBs	Nb_Components	Nb_In_and_Out_Flows	Pct_Unavailable	Nb_Users	Plateaus:	AS-IS	TRANSITION	TO-BE	Lifecycle:	Vision	Deployment_Status	Start_Date	End_Date
2	1DAC6]	4	A													1	0	8		Y	Y	Y		Maintain	In Operation	2005-01-01	2099-12-31
3	C23963]	34	Aphrodite													0	1	5		Y	N	N		Maintain	In Operation	2005-01-01	2099-12-31
4	64E0A9]	5														0	0	100		Y	Y	Y		Maintain	In Operation	2005-01-01	2099-12-31
5	78F998]	6	Athena Order Management		Happy	Sleepy		H	H	M		1250	38	4	5	3	1	50		N	Y	Y		Invest	re-Operation	2022-03-01	2099-12-31
6	50A8C2]	8	Bellona ESB		Sneezy	Dopey		H	M	H		69	0	0	0	0	0	50		Y	Y	Y		Maintain	In Operation	2005-01-01	2099-12-31
7	B84B3B]	58	BOPCO Journey Planner					H	L	L		20	4	2	0	0	2	120		Y	N	N		Phase Out	In Operation	2020-03-02	2023-06-28
8	D85A4C]	59	BOPCO Travel Info					H	M	M		45	6	3	0	0	5	120		Y	Y	N		Maintain	In Operation	2016-01-30	2089-09-09
9	4168FC]	7	CMDB		Sleepy	Bashful		M	L	M		258	50	4	4	6	5	100		Y	Y	N		Phase Out	In Operation	2005-01-01	2022-12-31
10	469E12]	64	Confluence					?	?	?			5	0	0	1		40		Y	Y	Y		?	?		
11	82A4FA]	10	Customer Mobile Application		Dopey	Doc		H	M	M		5	20	1	6	0	3	50		N	N	Y		Invest	re-Operation	2022-01-01	2099-12-31
12	E3ACE5]	11	Demeter		Bashful	Grumpy		M	M	M		545	4	2	0	0	0	62		Y	Y	N		Maintain	In Operation	2005-01-01	2099-12-31
13	C19E8D]	12	Generic Web Browser		Doc	Happy		M	M	H		62	3	0	1	0	0	100		Y	Y	Y		Maintain	In Operation	2005-01-01	2099-12-31
14	594294]	13	Hera		Grumpy	Sneezy		H	H	M		254	4	2	0	0	0	8		Y	Y	Y		Invest	In Operation	2005-01-01	2099-12-31
15	F876B9]	48	Hermes Supplier Evaluation System					L	H	H		56	2	1	0	0		23		N	Y	Y		Invest	?	2022-05-01	2099-12-31
16	BCF60F]	15	Janus		Happy	Sleepy		M	H	H		455	2	1	0	0	0	62		N	Y	Y		New	Idea	2023-01-01	2099-12-31
17	52DF5A]	65	Jira					?	?	?			10	0	0	2		15		Y	Y	Y		?	?		
18	8C4817]	16	Jupiter Cash Desk		Sneezy	Dopey		M	M	M		66	6	3	0	0	0	100		Y	Y	N		Phase Out	In Operation	2005-01-01	2023-12-31
19	E176F7]	9	Labnaf Powered by Sparx Systems Platform		Sleepy	Bashful		M	H	H		2	136	4	11	19	0	80		Y	Y	Y		Invest	In Operation	2005-01-01	2099-12-31
20	BAFF55]	17	LOGIN		Dopey	Doc		L	M	M		5	2	1	0	0	0	50		Y	Y	Y		Maintain	In Operation	2005-01-01	2099-12-31
21	241C10]	14	Mail Server		Bashful	Grumpy		M	H	H		9	13	0	1	2	0	8		Y	Y	Y		Maintain	In Operation	2005-01-01	2099-12-31
22	2E9A2E]	18	Mars		Doc	Happy		H	M	L		78	2	1	0	0	0	50		Y	Y	?		Maintain	In Operation	2005-01-01	2099-12-31
23	9E8DC2]	19	Mars WebApp		Grumpy	Sneezy		M	M	M		65	0	0	0	0	1	100		Y	Y	?		Maintain	In Operation	2005-01-01	2099-12-31
24	B6F870]	3	Microsoft Office		Happy	Sleepy		M	M	H		49	3	0	1	0	0	100		?	?	?		Maintain	In Operation	2005-01-01	2099-12-31
25	245816]	20	Minerva Card Payment		Sneezy	Dopey		H	M	L		867	15	1	1	2	0	8		Y	Y	Y		Invest	In Operation	2005-01-01	2099-12-31
26	80CE1B]	21	MyCo		Sleepy	Bashful		M	L	M		240	0	0	0	0	0	100		Y	N	N		Phase Out	In Operation	2005-01-01	2022-12-31
27	58F850]	22	Neptune Sales		Dopey	Doc		M	M	M		510	2	1	0	0	2	62		Y	N	N		Phase Out	In Operation	2005-01-01	2023-06-01
28	01CC8C]	23	Poseidon		Bashful	Grumpy		M	M	M		95	0	0	0	0	1	50		Y	?	?		Maintain	In Operation	2005-01-01	2099-12-31
29	08C79E]	66	Service Now					?	?	?			10	0	0	2		12		Y	Y	Y		?	?		
30	31D341]	24	Venus Cash Desk		Doc	Happy		M	L	L		1503	17	1	0	3	0	62		Y	N	N		Phase Out	In Operation	2005-01-01	2022-08-01
31	9D0EC2]	26	Vesta Web		Grumpy	Sneezy		H	M	M		57	2	1	0	0	1	50		Y	N	N		Maintain	In Operation	2005-01-01	2099-12-31
32	DC3FE5]	27	Vulcan Communication BE		Happy	Sleepy		M	M	M		1564	25	2	2	3	1	50		Y	Y	?		Maintain	In Operation	2005-01-01	2099-12-31
33	C70C6F]	68	Warehouse Plus					?	?	?		4856	0	0	0	0		3		?	?	?		?	?		
34	6FF363]	28	Zeus CCE		Sneezy	Dopey		M	M	H		56	3	0	1	0	4	8		Y	Y	Y		Maintain	In Operation	2005-01-01	2099-12-31
35	265413]	32	Zeus Convergent Mediation		Sleepy	Bashful		M	M	H		173	0	0	0	0	1	50		Y	Y	Y		Maintain	In Operation	2005-01-01	2099-12-31
36	12DE5C]	33	Zeus Mobile Synchronization		Dopey	Doc		M	M	H		674	0	0	0	0	0	62		Y	Y	Y		Maintain	In Operation	2005-01-01	2099-12-31
37	15E7F3]	29	Zeus MRC		Bashful	Grumpy		M	M	M		901	0	0	0	0	0	50		Y	Y	Y		Maintain	In Operation	2005-01-01	2099-12-31
38	31D8A7]	31	Zeus Pricing		Doc Brown	Bashful		H	L	L		2412	12	1	0	2	1	50		Y	Y	Y		Invest	In Operation	2005-01-01	2099-12-31
39	DA152A]	30	Zeus Sales Records Management		Grumpy	Doc Brown		M	L	H		55	18	2	3	1	0	100		Y	Y	Y		Maintain	In Operation	2005-01-01	2099-12-31

Overview:	Guid	AD	AE	AF	AG	AH	AI	AJ	AK	AL	AM	AN	AO	AP	AQ	AR	AS	AT	AU	AV	AW	AX	AY	AZ	BA	BB	BC	BD	BE	BF	BG	BH	BI	BJ	BK	BL	BM	BN	BO	BP	BQ	BR	BS	BT	BU	BV	BW	BX	BY	BZ	CA	CB	CC	CD	CE	CF	CG	CH	CI	CJ	CK	CL	CM	CN	CO	CP	CQ	CR	CS	CT	CU
		plications REALIZING Functional Dom	B2B External Exchanges	B2C External Exchanges	Bus Maintenance	Bus Route Planning	Bus Stop Management	Finance	HR & Corporate services	Information Management	Information Technology	Internal Exchange	Marketing & Sales	Security	rategy, Legal Affairs, Risk & Compliance	Supply Chain	Traffic Management	Traveler Communication	plications REALIZING Functional Area	Financial Exchanges	Digital Communication	Customer Payment	Cash Management	Accounts Payable	Accounts Receivable	t with other operators/third p	General Ledger - Tax - Closure	ations & Benefits Management	HR Reporting	HR Strategic Planning	Learning and Development	Recruitment and Hiring	Talent Management	Artificial Intelligence	Big Data	Business Intelligence	Corporate Repository	rprise Information Management	eographic Information System	Strategy to Portfolio (S2P)	Requirement to Deploy (R2D)	Request to Fulfill (R2F)	Detect to Correct (D2C)	Marketing Management	Multi-channel Sales	B2B Sales	B2C Sales	B2I Sales	M&S Master Data Management	Purchasing	Shipping and Transportation	r Communication Management	Supply Chain Reporting	Warehouse Management	Bus Personnel Information Channels	D-1 Traffic Management	Daily Traffic Management	Post Processing	Reporting Traffic Management	Customer Services	Pre-Booking Communication	Post-Booking Communication	Passenger Communication	plications REALIZING Functional Bloc	Payment Exchanges	Email Exchange	SMS Exchange	Card Payment	Cash Payment	Payment Confirmation	Deep Learning Model	Expert Systems
2	1DAC6]																																																																						
3	C23963]											X																																																										
4	64E0A9]											X																																																										
5	78F998]		X									X								X																									X	X	X																		X						
6	50A8C2]																																																																						
7	B84B3B]																X																																													X								
8	D85A4C]																X																																													X	X							
9	4168FC]									X																															X																															
10	469E12]																																																																									
11	82A4FA]											X																																		X																											
12	E3ACE5]											X																																	X		X																									
13	C19E8D]																																																																						
14	594294]											X																																	X		X																									
15	F876B9]														X																																		X																									
16	BCF60F]											X																																	X																									
17	52DF5A]																																																																						
18	8C4817]		X									X								X																										X	X																		X						
19	E176F7]									X																												X	X																																
20	BAFF55]											X																																		X																									
21	241C10]																																																																						
22	2E9A2E]															X																																													X								
23	9E8DC2]																																																																						
24	B6F870]														X																																		X																									
25	245816]						X															X																																																X			
26	80CE1B]																																																																												
27	58F850]											X																																	X																											
28	01CC8C]																																																																									
29	08C79E]																																																																									
30	31D341]											X																																		X																											
31	9D0EC2]															X																																													X								
32	DC3FE5]			X																	X																																															X	X			
33	C70C6F]														X																		X																																							
34	6FF363]																																																																									
35	265413]																																																																									
36	12DE5C]																																																																									
37	15E7F3]																																																																									
38	31D8A7]											X																																				X																									
39	DA152A]											X					X																											X																		X								

Generate Tabular Report (cont.)

Labnaf comes with many tabular report templates, but you might want model your own templates

See the **Tabular Report Generation** User Guide document and the examples on the Guidance Web Site

To start tabular report generation:

```
Command: GenerateTabularReports

Description: Generate spreadsheets from a model repository based on configuration stored in that
 same repository.

Usage : lnps GenerateTabularReports [arguments]

Arguments:

   SourceRepoPathName: Path name of the source model repository (EAP file).

   OutputDirectoryPath: Directory path name where the spreadsheets must be generated.The name of
 each spreadsheet file is the name of the template report.

   [ElementPrototypeName]: The name of a specific element prototype name for which all embedded
tabular report templates must be applied.

   [TabularReportTemplateName]: The name of a specific tabular report template to be applied.
```

By default, all scheduled report templates will be applied. But you can also be specific.

When a report template name ends with ‘.CSV’ a CSV file is generated instead of Excel.

Generate Tabular Report (cont.)

To start tabular report generation:

```
Command: GenerateTabularReports

Description: Generate spreadsheets from a model repository based on configuration stored in that
 same repository.

Usage : lnps GenerateTabularReports [arguments]

Arguments:

   SourceRepoPathName: Path name of the source model repository (EAP file).

   OutputDirectoryPath: Directory path name where the spreadsheets must be generated.The name of
 each spreadsheet file is the name of the template report.

   [ElementPrototypeName]: The name of a specific element prototype name for which all embedded
tabular report templates must be applied.

   [TabularReportTemplateName]: The name of a specific tabular report template to be applied.
```

By default, all report templates will be applied. But you can also be specific.

When a report template name ends with ‘.CSV’ a CSV file is generated instead of Excel.

GenerateDoc
(Word, RTF, PDF)

To start document generation:

```
Command: GenerateDoc

Description: Generate a Word, RTF or PDF document from a model repository package.

Usage : lnps GenerateDoc [arguments]

Arguments:

   SourceRepoPathName: Path name of the source model repository (EAP file).

   OutputPath: Path name of the document file to be generated.
The file extension specified will determine the format of the generated document - for example, RTF, PDF

   PackageGuid: The GUID of the package or master document to run the report on.

   TemplateName: The document report template to use; if the PackageGUID has a stereotype of MasterDocume
nt, the template is not required.
```

GenerateHtml

To start HTML generation:

```
Command: GenerateHTML

Description: Generate an HTML web site from a model repository package.

Usage : lnps GenerateHTML [arguments]

Arguments:

   SourceRepoPathName: Path name of the source model repository (EAP file).

   OutputPath: The path of the file system folder where the HTML pages must be generated.

   SourcePackageGUID: The GUID of the repository package for which HTML must be generated.

   [WebSiteTemplateName]: The optional name of a web style template used for HTML generation
(default=Sparx EA default template).
```

On the web site, you can email a stable link to the current page by clicking on the little envelope.

BackupToFile

To start the backup to a file:

```
Command: BackupToFile

Description: Backup a DBMS repository to a file-based Repository (.eap, .eapx, .eadb, .feap).

Usage : "C:\Program Files\Labnaf\PowerShell\lnps.exe" BackupToFile [arguments]

Arguments:

   SourceRepoPathName: Path name of the source repository (EAP file containing a connection string).

   DestPathName: Path name of the destination file-based repository (.eap, .eapx, .eadb, .feap).

   [DetailedLogFilePath]: Alternative path for the detailed log file (supersedes the default detailed log file path)
```

SourceRepoPathName (EAP) must point to a DBMS repository

ScheduleCommand

Alternatives to « ScheduleCommand » (If you prefer):

- **Windows Task Scheduler** (https://docs.microsoft.com/en-us/dynamics365/business-central/dev-itpro/developer/devenv-task-scheduler)
- **Your company standard scheduler**

To schedule a **nightly** command **starting at midnight**:

- **InitialStartTime = 00:00:00** — **Don't schedule 2 commands starting exactly at the same time**
- **PeriodAsMinutes = 1440** — There are 1440 minutes in one day

To start the schedular:

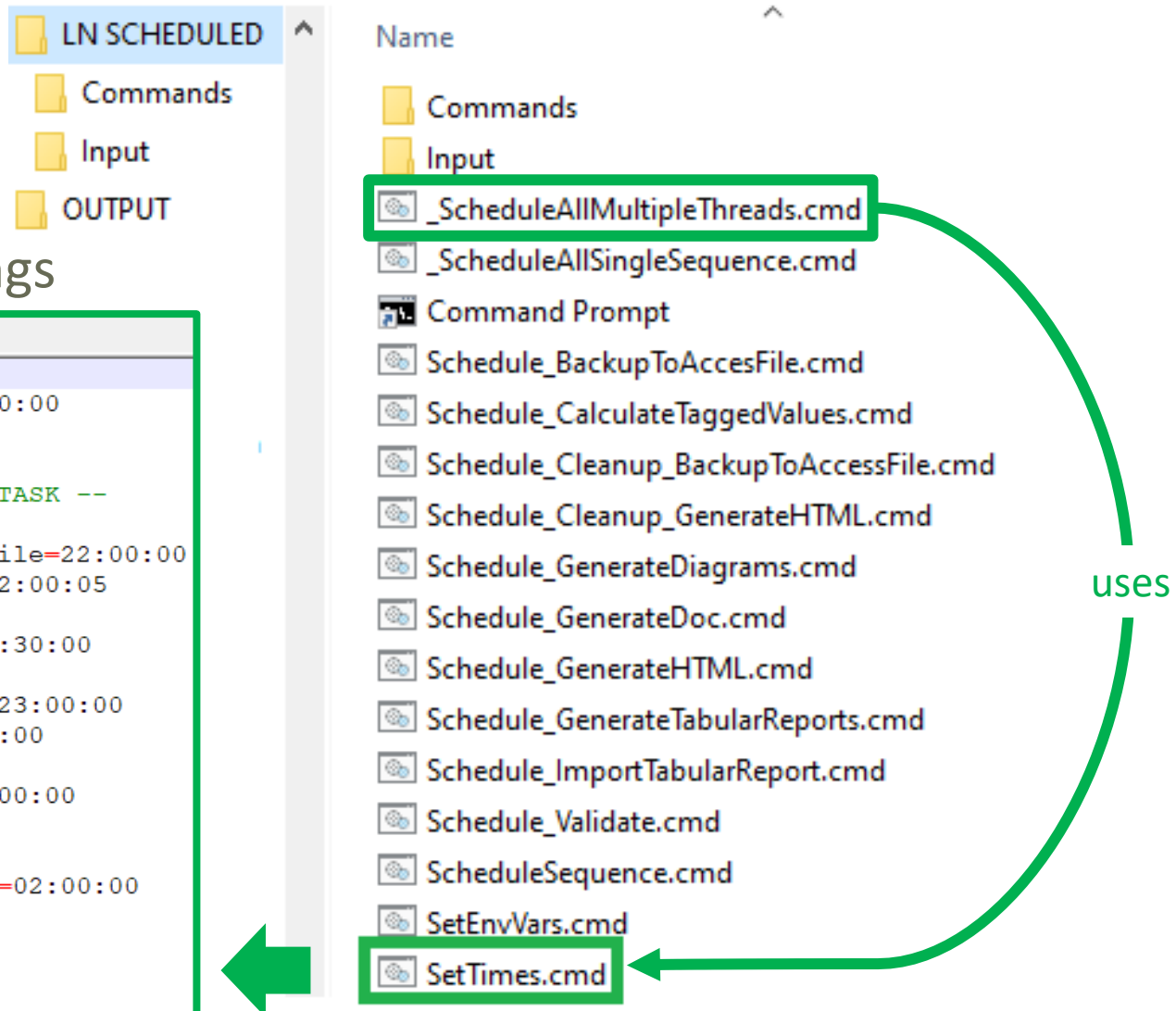
```
Command: ScheduleCommand

Description: Schedule a task to run periodically starting at a specific time.

Usage : lnps ScheduleCommand [arguments]

Arguments:

   CommandPathName: Path name of the command that needs to be periodically started (.cmd or .bat).

   InitialStartTime: The initial start time for the task (HH:MM:SS).

   [PeriodAsMinutes]: The length of a period expressed in minutes.
```

Final Recommendation in case you use “ScheduleCommand”

- Use the preconfigured batches and settings

SetTimes.cmd

```
REM -- SINGLE START TIME --
Set StartTime_AllSingleSequence=00:00:00


REM -- SPECIFIC START TIME FOR EACH TASK --

Set StartTime_Cleanup_BackupToAccesFile=22:00:00
Set StartTime_Cleanup_GenerateHTML=22:00:05

Set StartTime_ImportTabularReport=22:30:00

Set StartTime_CalculateTaggedValues=23:00:00
Set StartTime_GenerateDiagrams=23:30:00

Set StartTime_BackupToAccessFile=00:00:00
Set StartTime_Validate=01:00:00

Set StartTime_GenerateTabularReports=02:00:00
Set StartTime_GenerateDoc=02:30:00
Set StartTime_GenerateHTML=03:00:00


REM --------------------------

set SCHEDULED_MINUTES_UNTIL_RESTART=1440
```



- Take computer reboots into account

Labnaf PowerShell Commands

1. Overview
2. Strategy and Architecture Operations
3. Systems Integrations and Content Refactoring
4. Command Compatibility Matrix

Labnaf PowerShell commands for
Systems integrations and content refactoring

- ClonePackage
- CreatePackage
- ExportToXmi
- ImportConnections
- ImportFromXmi
- ImportTabularReport
- MoveElementsToCalculatedParent
- MoveElementsToPackage
- MovePackagesToPackage
- RenameItem
- ScheduleCommand
- SetDiagramProperty
- SqlExportToCsv

Detailed information in the
Labnaf PowerShell Reference Guide
Latest version:
https://www.labnaf.one/EndUserMaterial/Labnaf_PowerShell/Labnaf%20PowerShell%20-%20Reference%20Guide.pdf

Labnaf PowerShell Commands

1. Overview
2. Strategy and Architecture Operations
3. Systems Integrations and Content Refactoring
4. Command Compatibility Matrix

Labnaf PowerShell **Command Compatibility Matrix**

Power Shell Commands	Labnaf Script Compatible	Sql Server	Pro Cloud Server	SQLite (QEA)
AutoConnectorsDelete	X	X	X	-> UI
AutoConnectorsGenerate	X	X	X	-> UI
BackupToFile		X	X	
CalculateTaggedValues	X	X	X	X
ClonePackage	X	X	X	X
CreatePackage	X	X	X	X
ExportToXmi	X	X	X	X
GenerateCharts	X	X	X	X
GenerateDiagrams	X	X	X	X
GenerateDoc	X	X		X
GenerateHTML	X	X		X
GenerateImplicitData	X	X	X	
GenerateTabularReports	X	X	X	X
ImportConnections	X	X	X	X
ImportFromXmi	X	X	X	X
ImportTabularReport	X	X	X	-> UI
MoveElementsToCalculatedParent	X	X	X	X
MoveElementsToPackage	X	X	X	X
MovePackagesToPackage	X	X	X	X
RenameItem	X	X	X	X
ScheduleCommand		X	X	X
SetDiagramProperty	X	X	X	X
SetProperty	X	X	X	X
SqlExportToCsv	X	X	X	X
Validate	X	X	X	X

Legend
-> UI : Use the Labnaf AddIn
i.e. the user interface