Productivity Tools

Cascaded Value Calculations

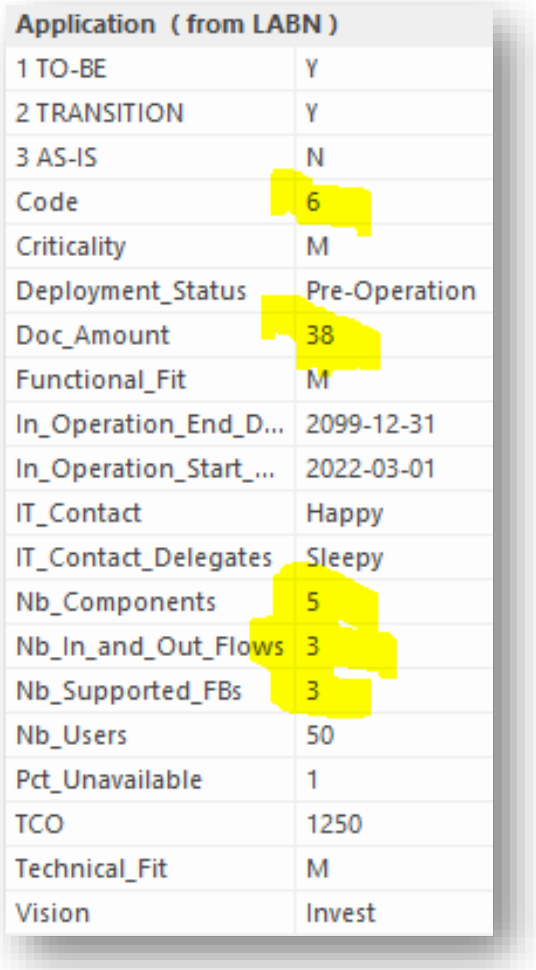
www.labnaf.one

Value Calculation

1. Overview
2. Defining the type of element needing calculation
3. Scenarios for creating new value controls
4. Selecting the collection of elements to be calculated
5. Defining the tagged values to calculated
6. Making calculated tagged values read only
7. Value calculation summary
8. Triggering periodical value calculations

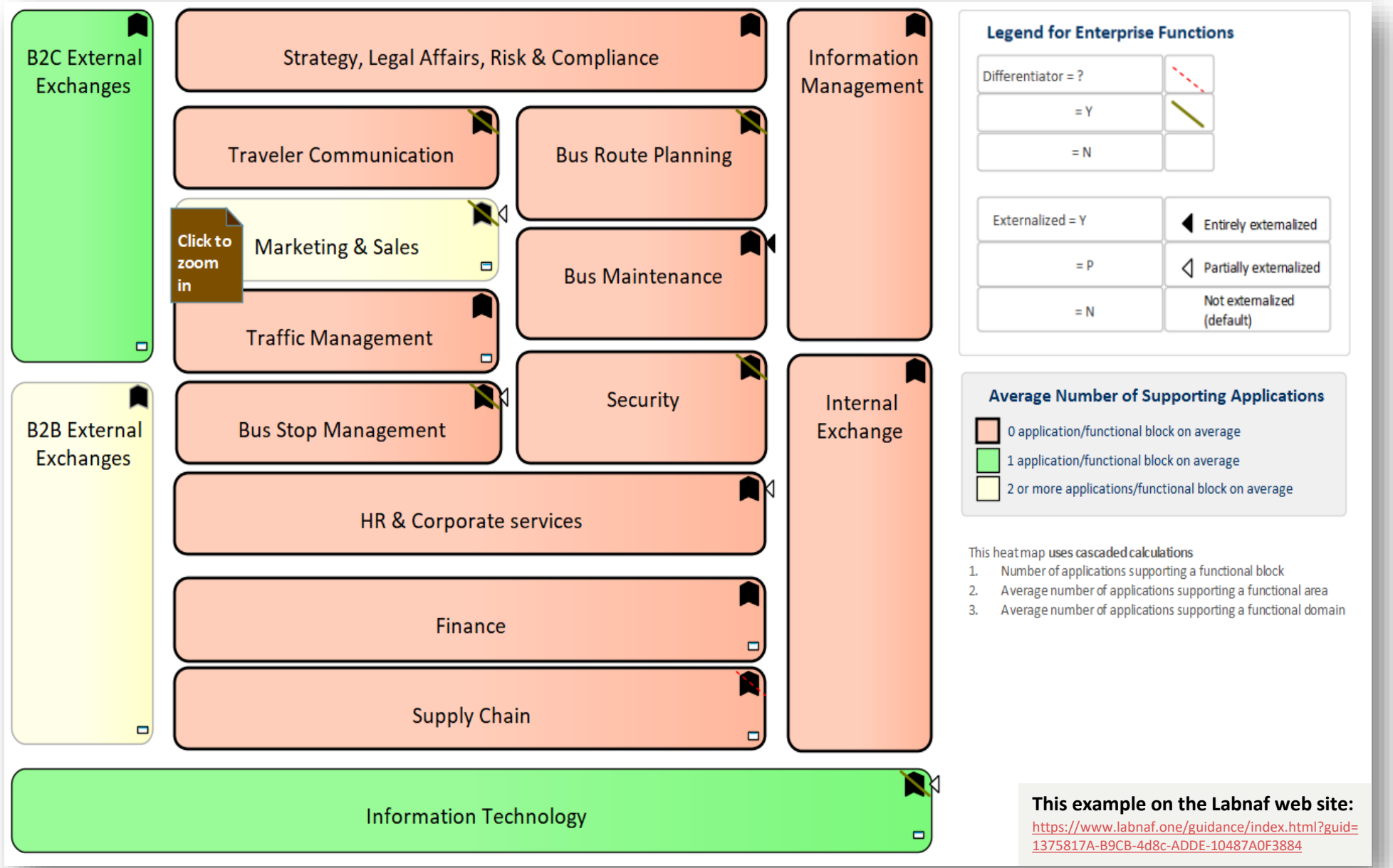
The purpose of value calculation

- Calculate element properties/tagged values
- Value can be in any format or any combination of formats (text, numeric, date, time series…)
- Calculations can be based on any propertys, any elements, any connectors, or any other information stored in the repository
- Calculations can be based on the result of other calculations
- If the calculate tagged value does not exist, then it is automatically added

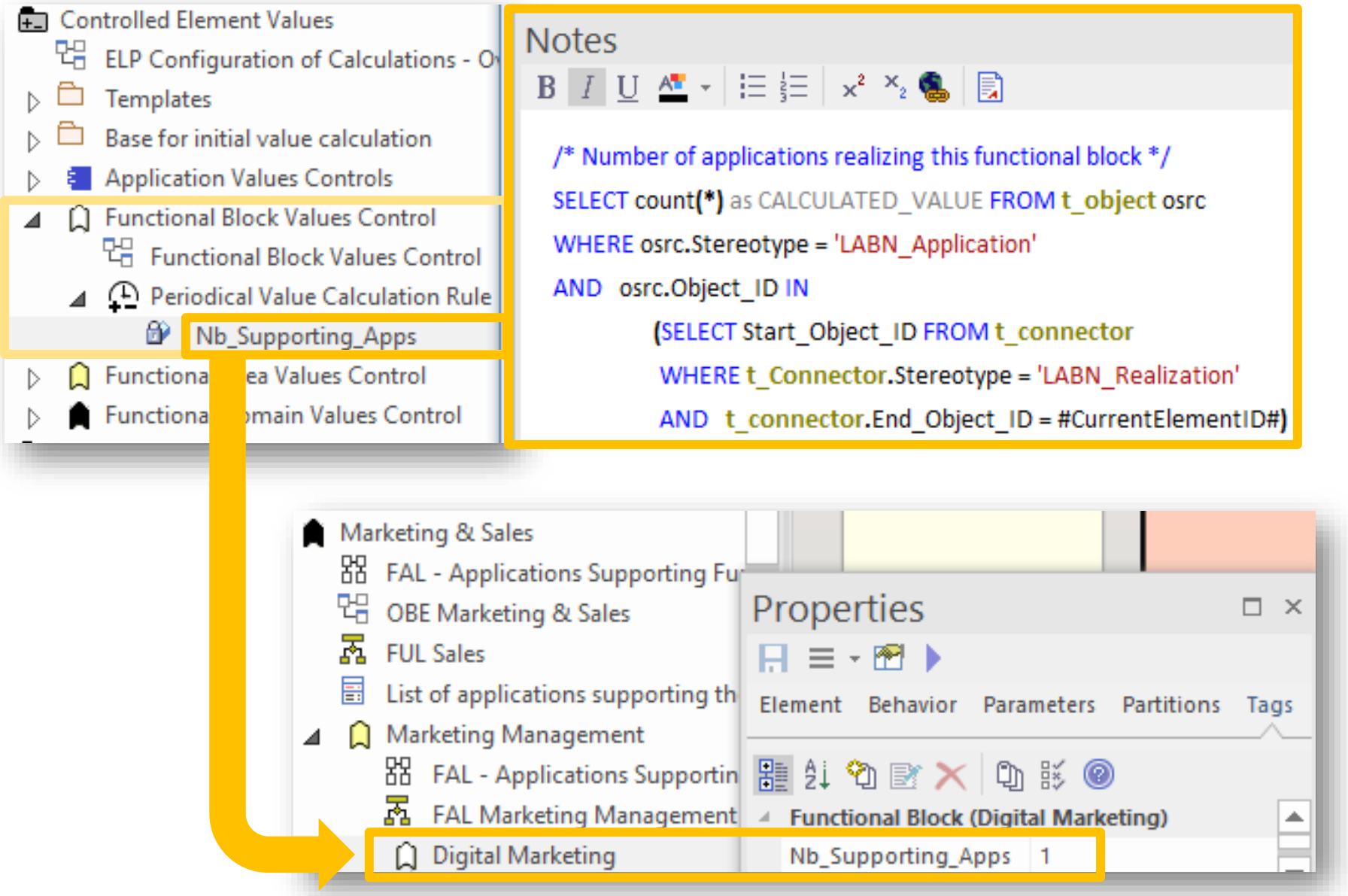


Application (from LABN)	
1 TO-BE	Y
2 TRANSITION	Y
3 AS-IS	N
Code	6
Criticality	M
Deployment_Status	Pre-Operation
Doc_Amount	38
Functional_Fit	M
In_Operation_End_D...	2099-12-31
In_Operation_Start_...	2022-03-01
IT_Contact	Happy
IT_Contact_Delegates	Sleepy
Nb_Components	5
Nb_In_and_Out_Flows	3
Nb_Supported_FBs	3
Nb_Users	50
Pct_Unavailable	1
TCO	1250
Technical_Fit	M
Vision	Invest

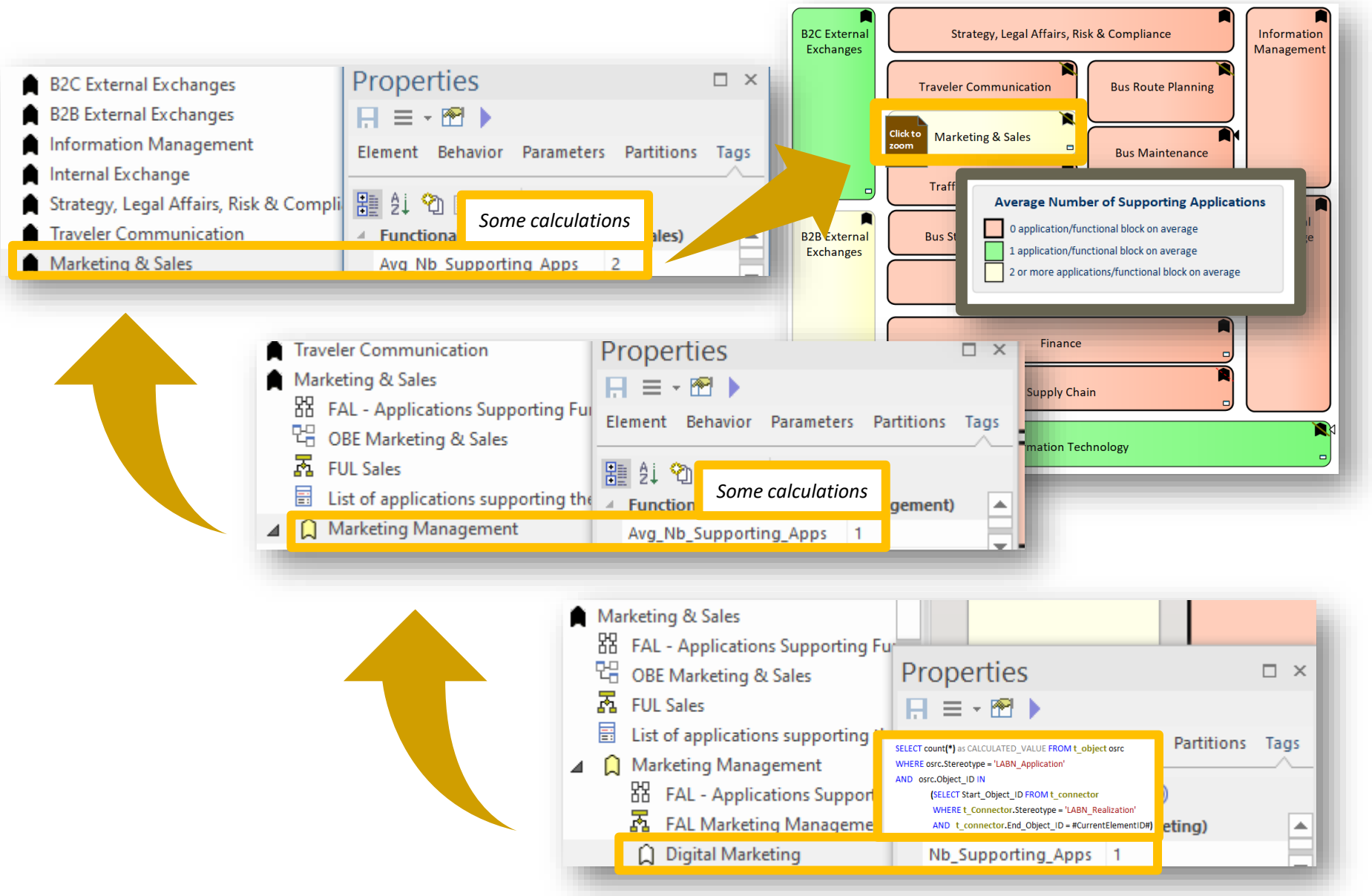
Example: Heat map using cascaded calculations



Example: Heat map using cascaded calculations (cont.)

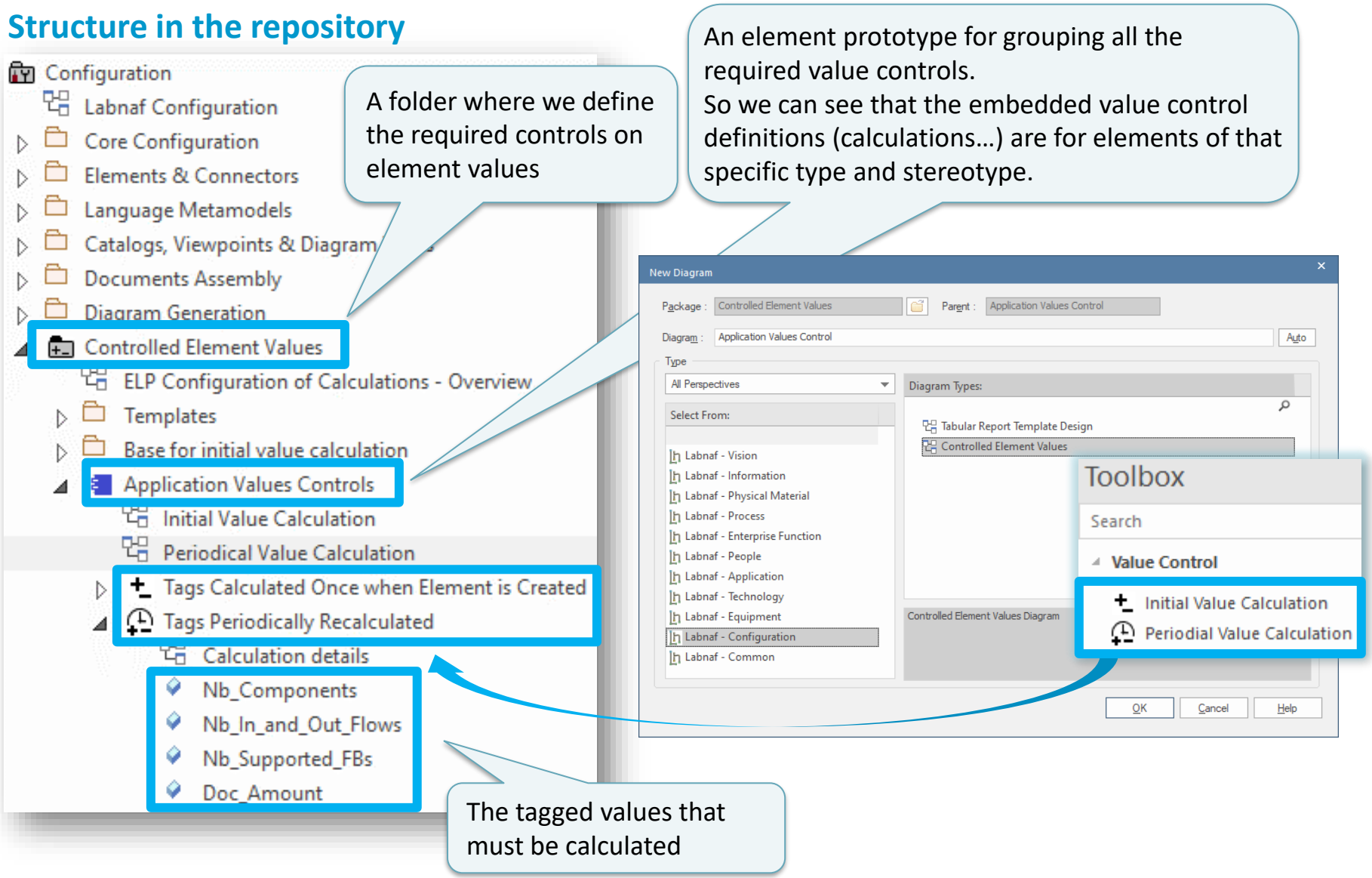


Example: Heat map using cascaded calculations (cont.)



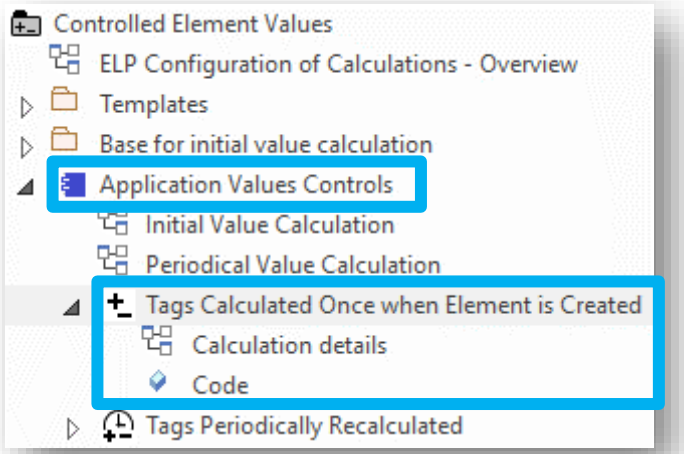
Value Calculation - Overview

Structure in the repository

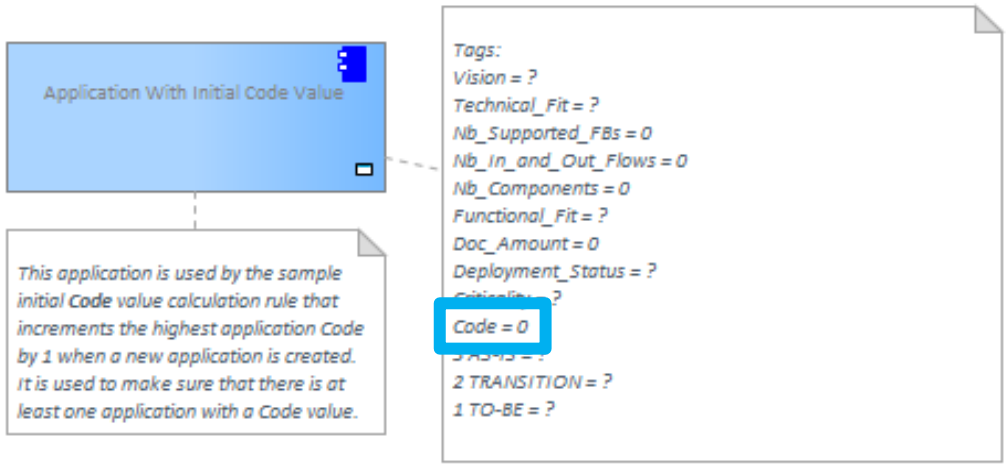


Initial Value Calculation Overview

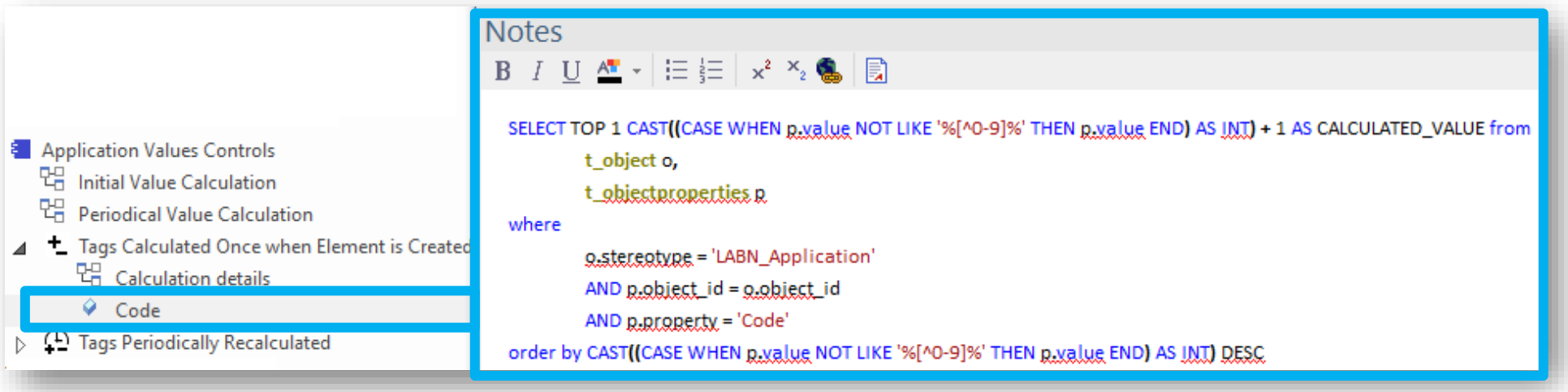
1. Structure: What tagged values need to be calculated for which stereotype?



2. Base: What is the referenced base element in case there is no other element with an initial value to be incremented?

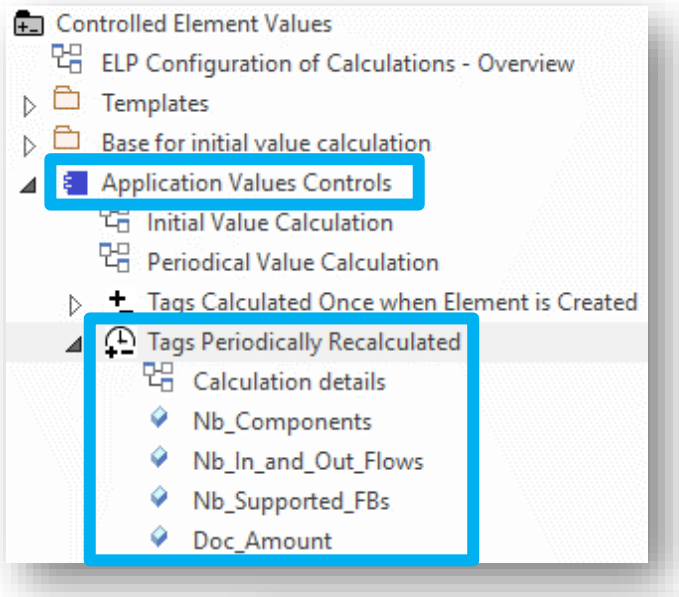


3. Calculation: How shall we calculate the value?

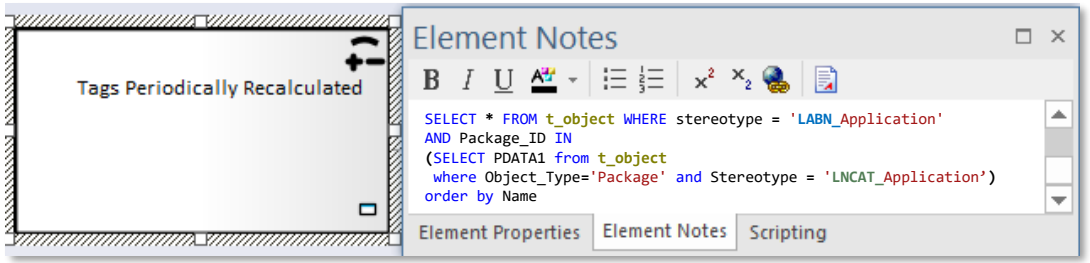


Periodical Value Calculation - Overview

1. Structure: What tagged values need to be calculated for which stereotype?

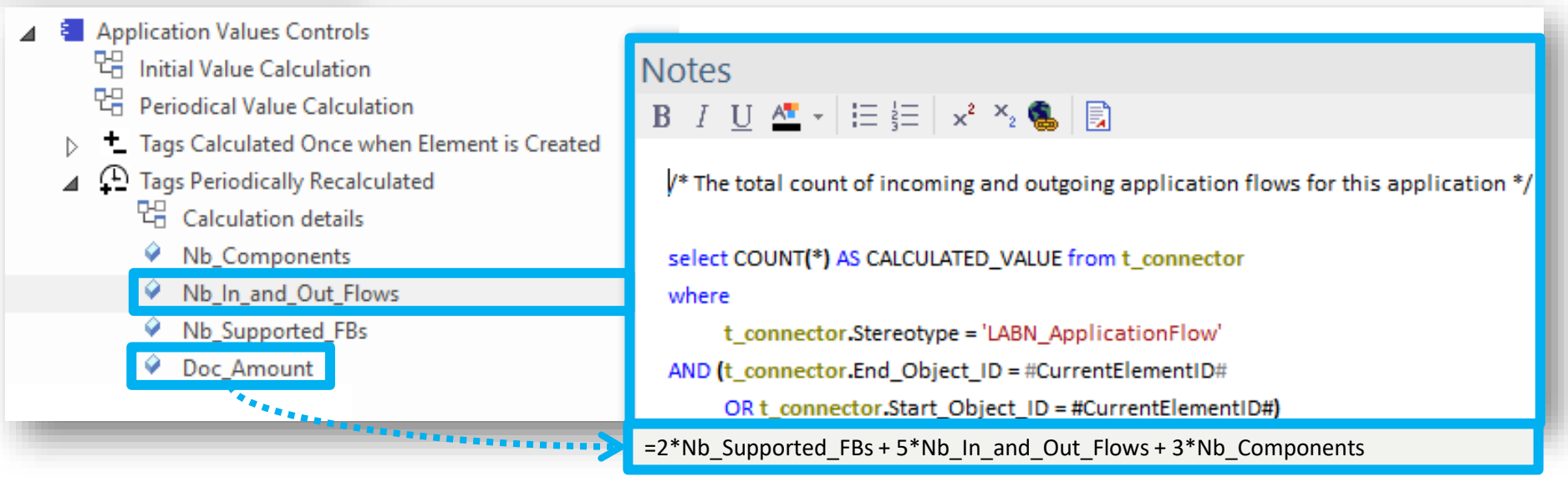


2. Scope (optional): Which elements need to be updated?



By default, all elements with the same stereotype "LABN_xxx" as the element prototype are selected from the related catalog packages with stereotype "LNCAT_xxx".

3. Calculation: How shall we calculate the values?

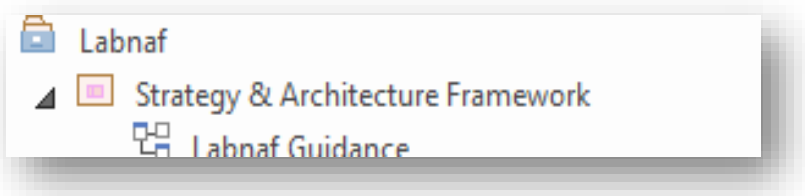


Value Calculation

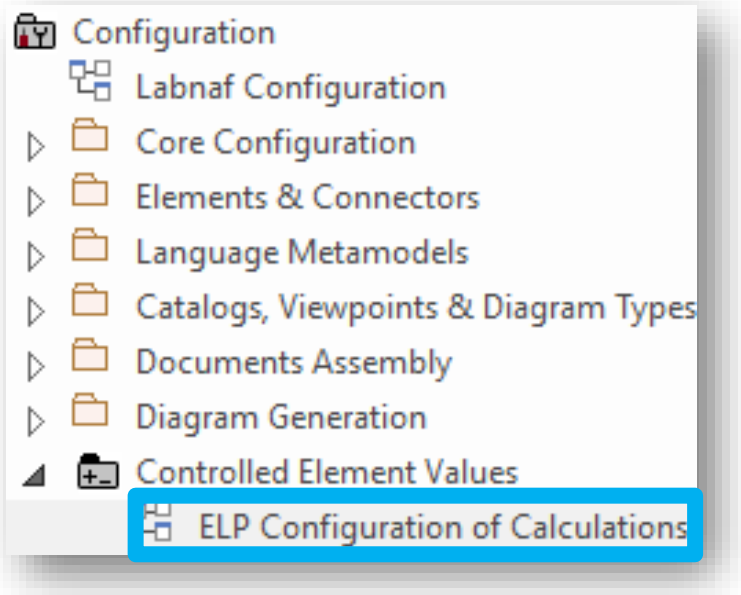
1. Overview
2. Defining the type of element needing calculation
3. Scenarios for creating new value controls
4. Selecting the collection of elements to be calculated
5. Defining the tagged values to calculated
6. Making calculated tagged values read only
7. Value calculation summary
8. Triggering periodical value calculations

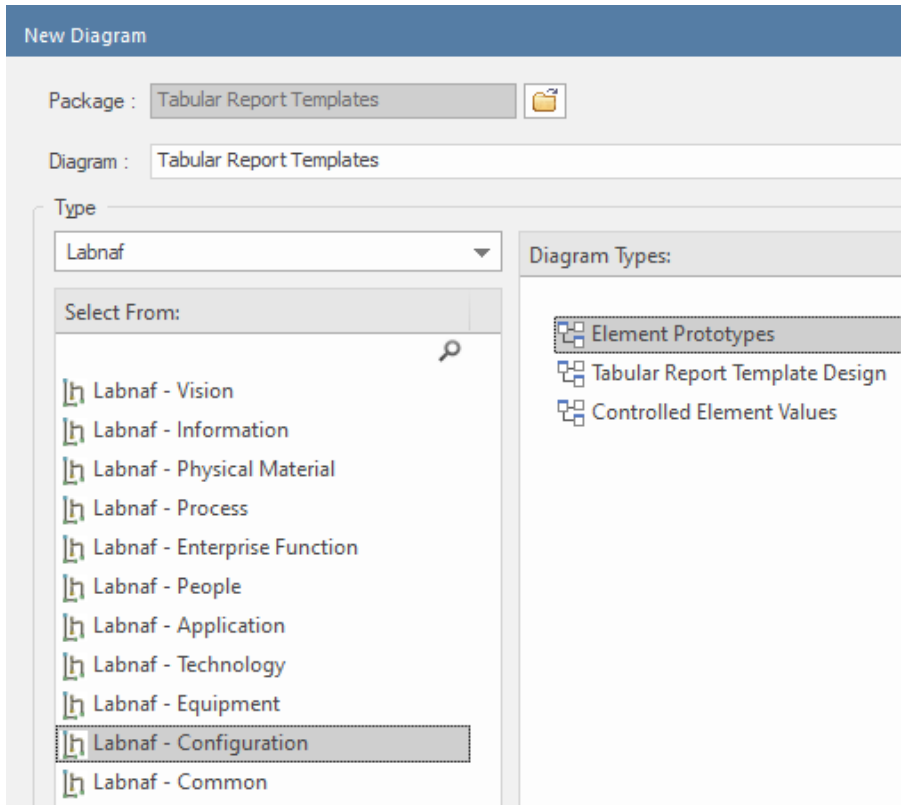
Define the type of element needing calculation…

- If it does not exist yet, create a diagram of type “Element Prototypes” or any diagram type containing the elements you need in the toolbox.



- Name it, for example, “ELP Configuration of Calculations” :



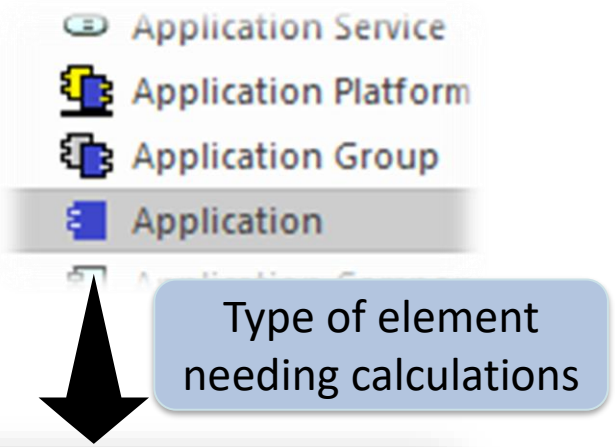


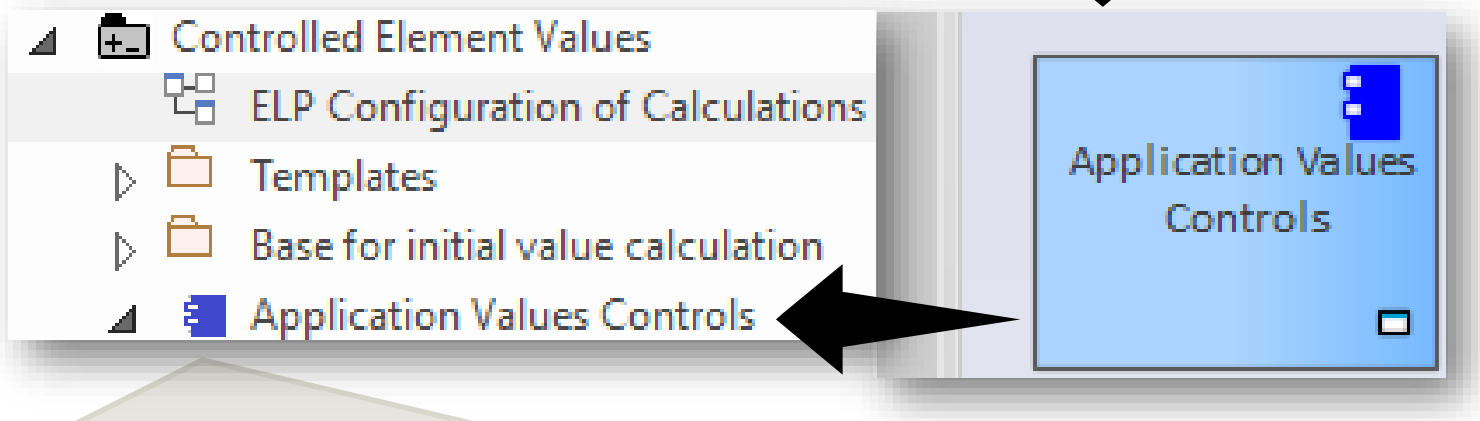
… using an element prototype

- Select an element in the toolbox

  You need only one element prototype of each type. But you can have more, for example, to further classify your calculations.

- Name the element prototype for example “Application Values Controls”





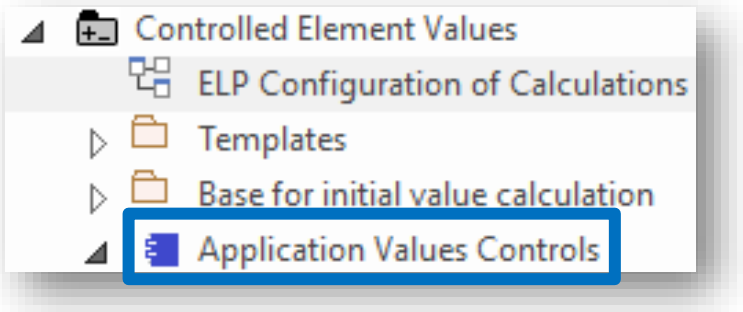
This is an **element protype** that will group all the **application calculations** (periodical or initial value).

By default, the Labnaf PowerShell periodically calculates values for all element prototypes contained in the “Controlled Element Values” folder. But you can be selective as well.

Value Calculation

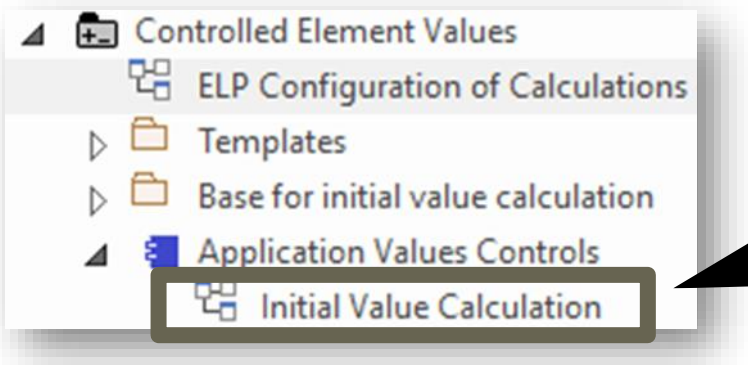
1. Overview
2. Defining the type of element needing calculation
3. Scenarios for creating new value controls
4. Selecting the collection of elements to be calculated
5. Defining the tagged values to calculated
6. Making calculated tagged values read only
7. Value calculation summary
8. Triggering periodical value calculations

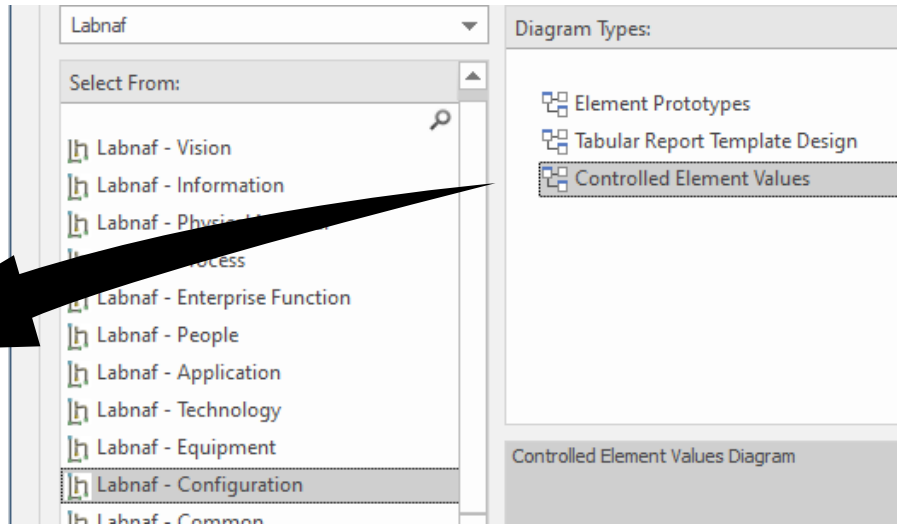
Add a diagram for creating your Value Controls (Value calculation elements)

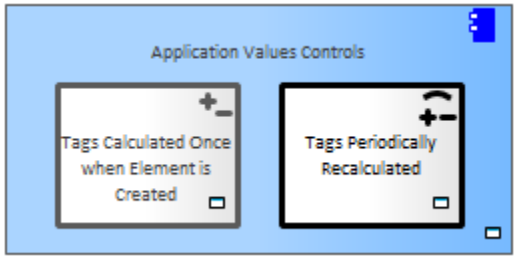


Right click on the element prototype and select "**Add Diagram**"

Under "Labnaf Configuration", select "**Controlled Element Values**"







2 possible scenarios for **Creating a New Value Control**

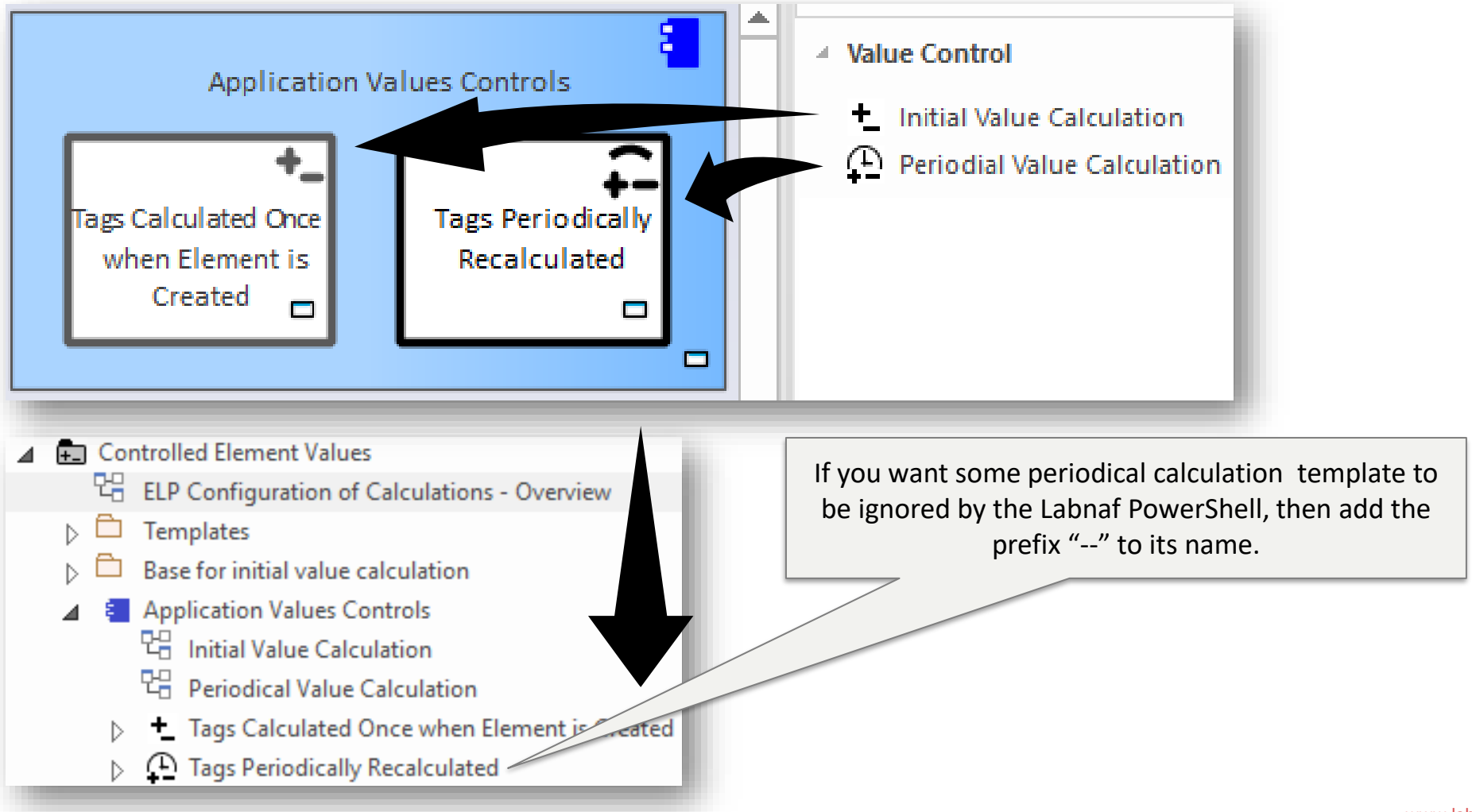
From Scratch Add a value control from the toolbox

Fast Track Copy and adapt an existing value control

From Scratch

Add a value control from the toolbox

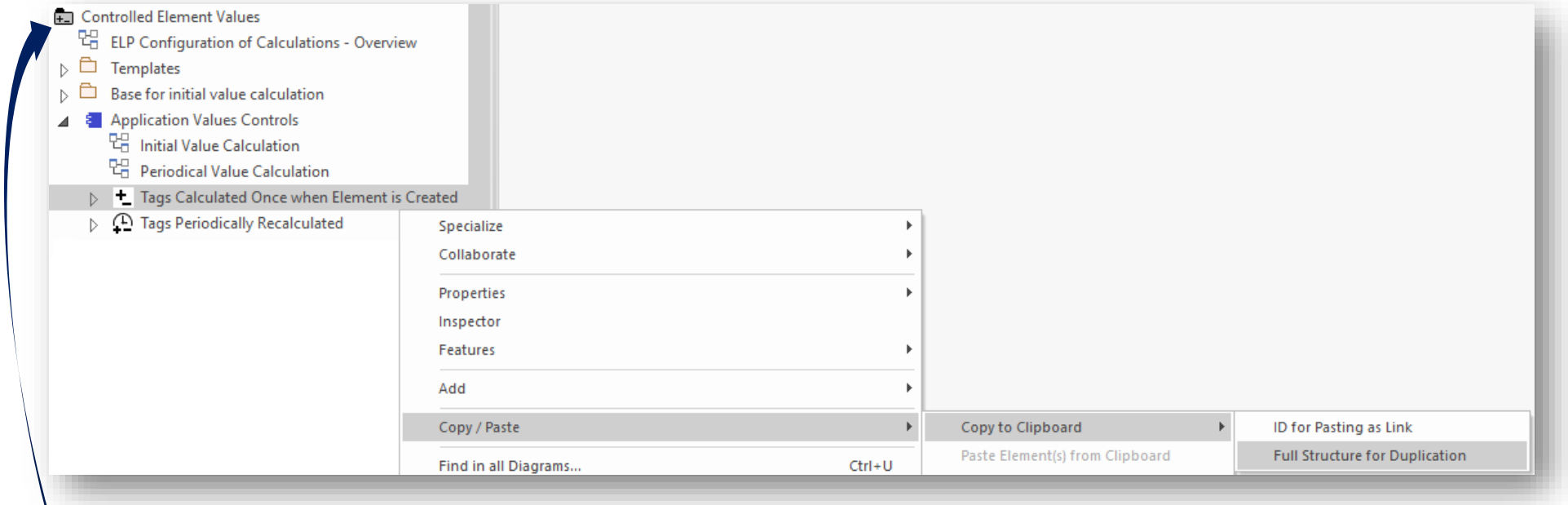
Add value controls following your needs and with some meaningful name(s)



Fast Track

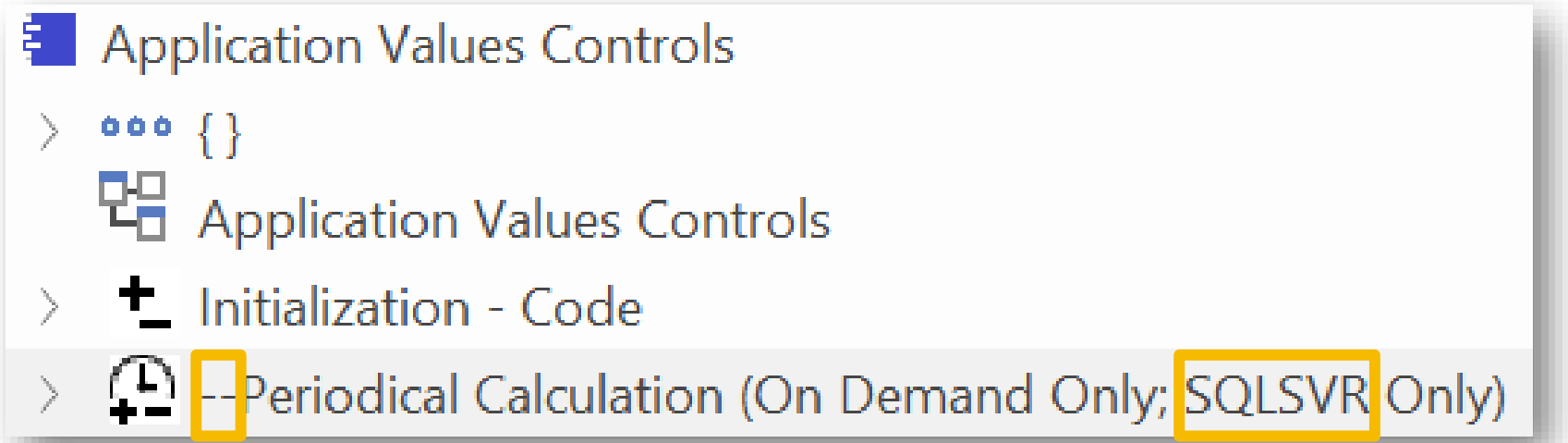
Copy and adapt an existing value control

- Copy an existing value control



- Paste to the “Controlled Element Values” Folder
- Rename the new value control
- Move it into the appropriate element prototype
- Add/remove the content you need. See following slides…

Value Control Behavior



If you want some periodical calculation template to be ignored by the Labnaf PowerShell, then add the prefix "**--**" to its name.

You will still be able to run it from the modeling environment.

If you build calculations that work only with SQL Server than include "**SQLSVR**" anywhere in the value control name.

They will be ignored in the case where the database is not SQL Server.
The Labnaf PowerShell will issue a warning.

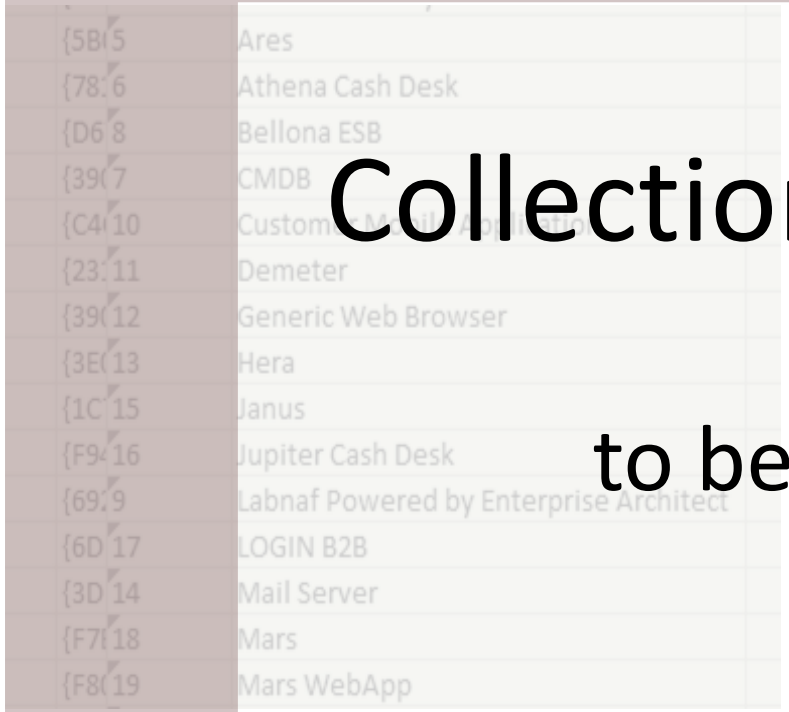
Completing/Adapting the New Value Control…

Value Calculation

1. Overview
2. Defining the type of element needing calculation
3. Scenarios for creating new value controls
4. Selecting the collection of elements to be calculated
5. Defining the tagged values to calculated
6. Making calculated tagged values read only
7. Value calculation summary
8. Triggering periodical value calculations

For PERIODICAL Value Calculations ONLY

Selecting the



Collection of Elements

to be Calculated

Defining a selection of elements (scope) requiring calculations

Options that can be defined in the template

1. Elements and packages present in a “scope” diagram that belongs to the template
2. Elements selected by some SQL select defined in the template
3. All elements in the catalog (default)

If multiple scopes have been defined, then the above sequence corresponds to priorities

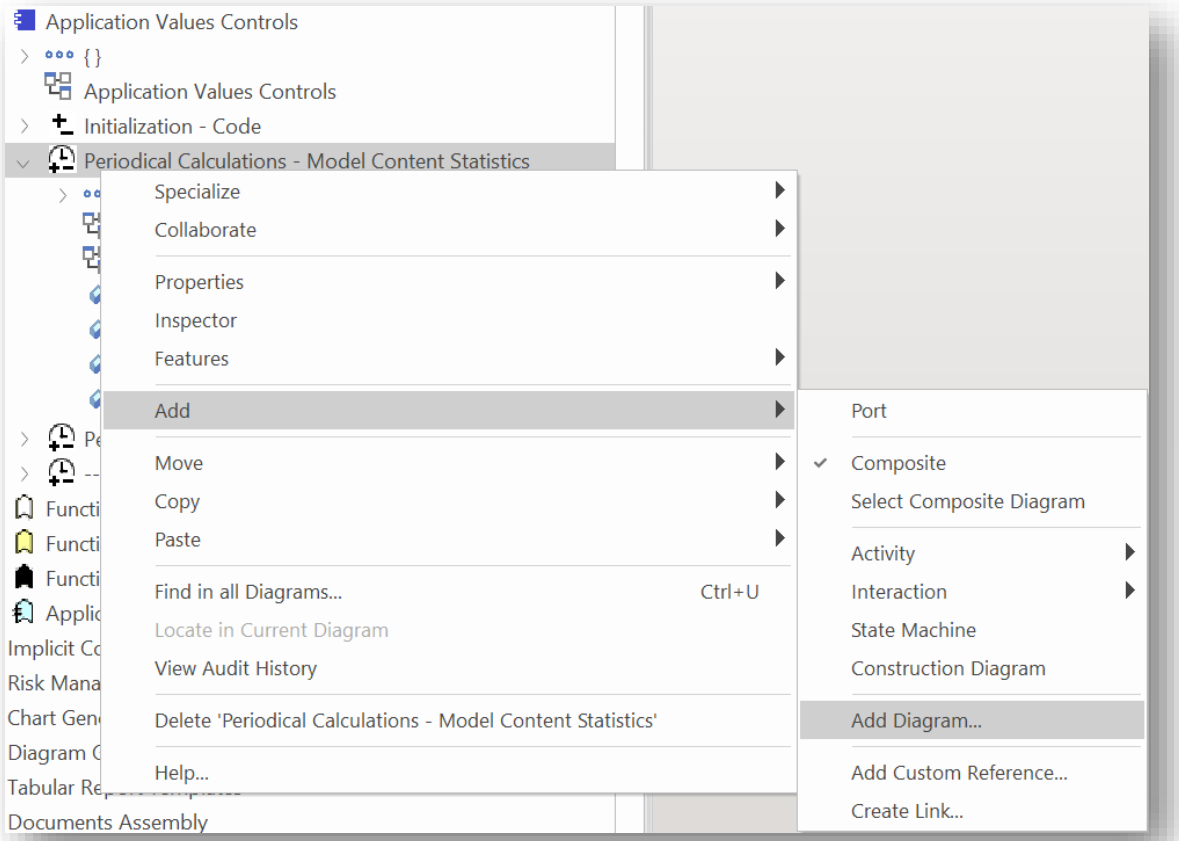
Additional option when the calculation is started from the user interface

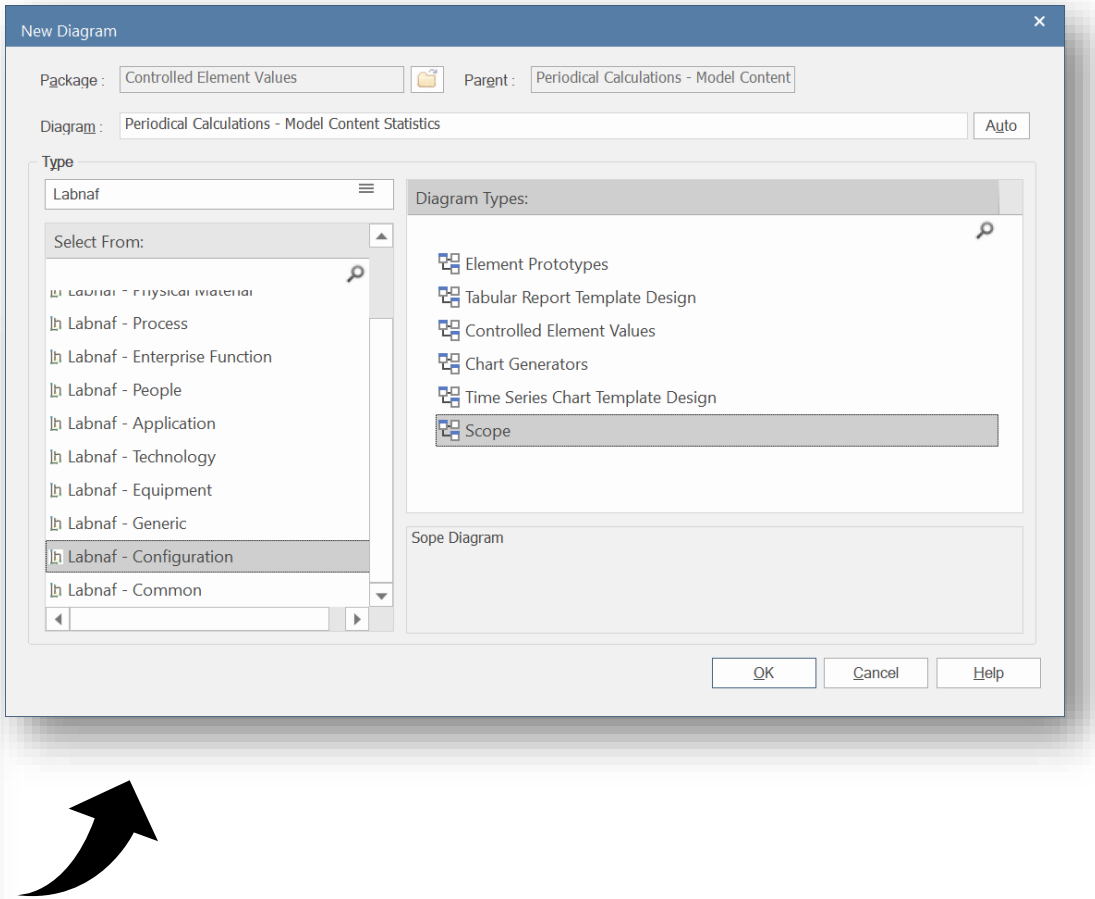
- Apply all calculations applicable to the element selected in the project browser or in the active diagram (typically used for calculation development and test purpose)

1. Scope Diagram

Creating a “scope” diagram defining the collection of element to be reported

- Select a periodical value calculation template element and add a « Scope » diagram



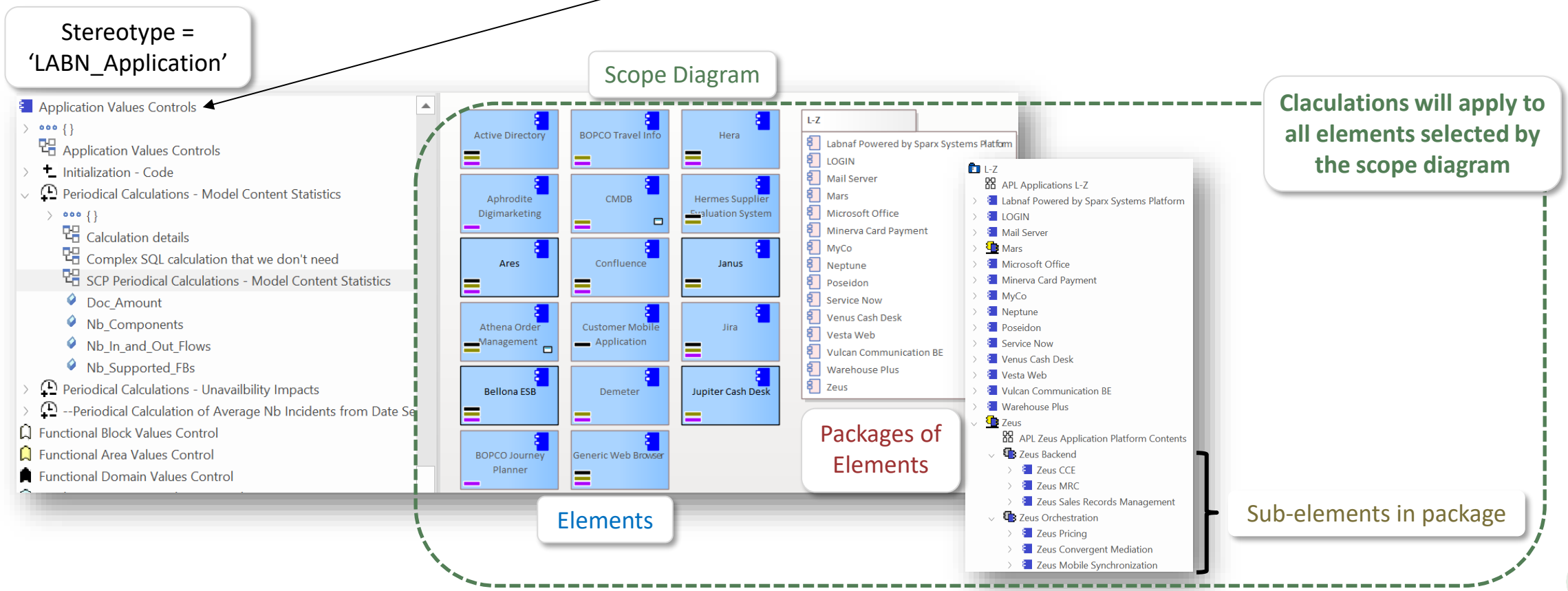


Populate the Scope diagram

- Add to the scope diagram the elements and packages of elements for which calculations are required

  ⇒ Collected elements =

  - Elements and packages of elements (including sub-elements) in the scope diagram
  - that have the same stereotype as the parent element prototype



2. Scope As SQL Select

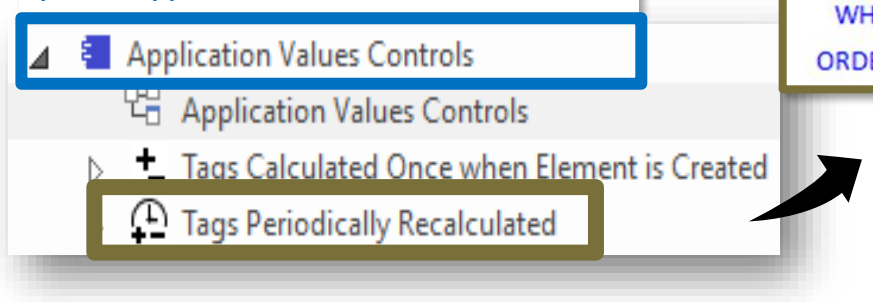
Custom **SQL** for selecting the **collection of elements** requiring calculations.

The "**Notes**" property of the value control element contains some SQL SELECT statement.
That SQL statements selects the elements that have values to be calculated.

Notes

```
/* Selection in a 2 levels hierarchy of packages using a variable to be defined as
an attribute in the "(((ModelingLanguageConfiguration)))" element */
SELECT * FROM t_object o
WHERE stereotype='LABN_Application'
AND o.Package_ID IN
(SELECT package1.package_ID FROM t_package package1
LEFT JOIN t_package package2 ON package2.package_id = package1.parent_id
  WHERE package2.ea_guid = '#MyParentApplicationPackageGUID#')
ORDER BY Name
```

By default, all elements with the same stereotype as the element prototype are selected.



With professional database engines, that SELECT statement can reach a level of sophistication that goes way beyond users' requirements.

File-based databases, on the other hand, have some limitations, but it is still usually sufficient to implement most use cases. File-based databases are anyway not designed for running on professional database servers.

Sample SQL Statements

```
/* Selection in a 2 levels hierarchy of packages */
SELECT * FROM t_object o
WHERE stereotype='LABN_Application'
AND o.Package_ID IN
(SELECT package1.package_ID FROM t_package package1
LEFT JOIN t_package package2 ON package2.package_id = package1.parent_id
  WHERE package2.ea_guid = '#uMyVariableContainingAPackageGUID#')
ORDER BY Name

/* selection in a set of packages
SELECT * FROM t_object WHERE stereotype ='LABN_Application' AND package_ID=550
UNION
SELECT * FROM t_object WHERE stereotype ='LABN_Application' AND package_ID=1126
*/

/* Ordered selection in a set of packages (Access only)
SELECT * FROM (
SELECT * FROM t_object WHERE stereotype ='LABN_Application' AND package_ID=550
UNION
SELECT * FROM t_object WHERE stereotype ='LABN_Application' AND package_ID=1126
)
ORDER BY NAME
*/
```

Statement can include comments

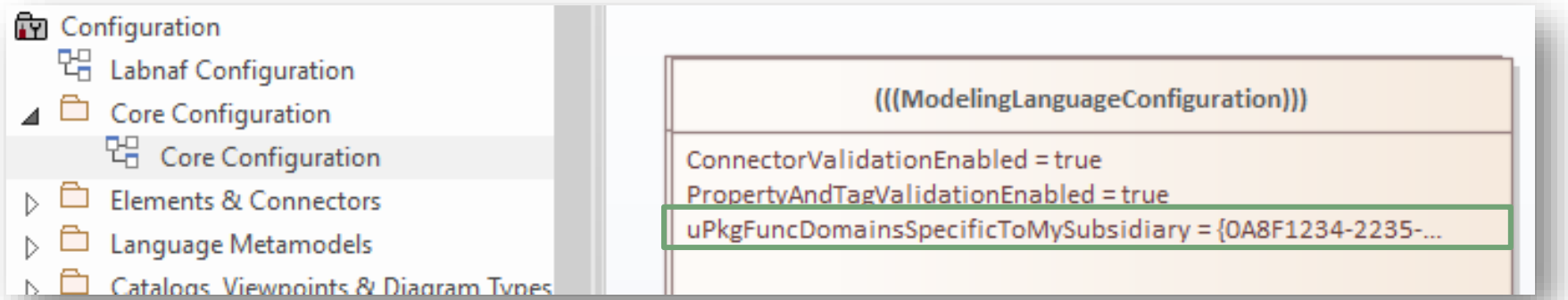
/* my comment */

These SQL statements can include variables

Variables defined in the core configuration can be used in SQL statements. You can add your own variables.

User-define variable must start with the letter **'u'**

Example: A user-defined variable containing the GUID of a package:



Usage of the user-defined variable in a SQL statement:

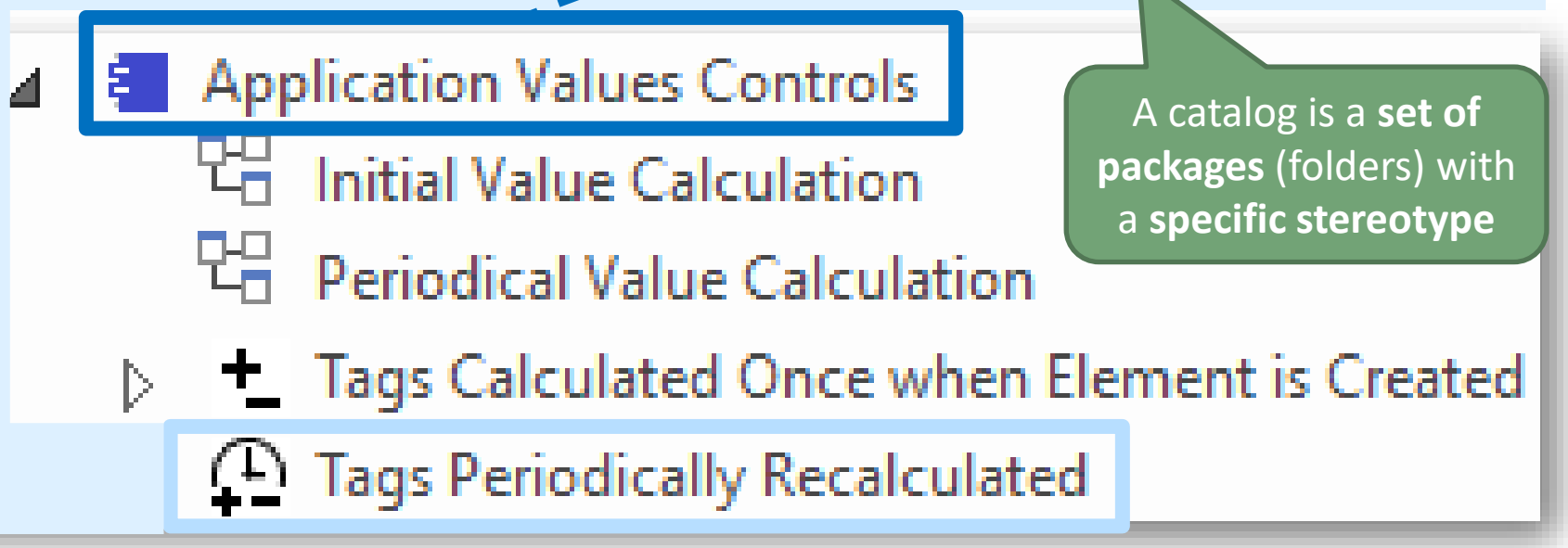
```
select * from t_object o inner
join t_package p on o.package_id = p.package_id
  where o.ParentID = 0 and o.stereotype like 'LABN_%'
  and p.ea_guid = '#uPkgFuncDomainsSpecificToMySubsidiary#'
ORDER BY o.Name
```

3. Scope As Catalog (Default)

Default Scope = All elements in the catalog

By default, all elements with the same stereotype as the element prototype are selected from the related catalog

```
SELECT * FROM t_object WHERE stereotype = 'LABN_Application'
AND Package_ID IN
(SELECT PDATA1 from t_object
 where Object_Type='Package' and Stereotype = 'LNCAT_Application')
order by Name
```



A catalog is a **set of packages** (folders) with a **specific stereotype**

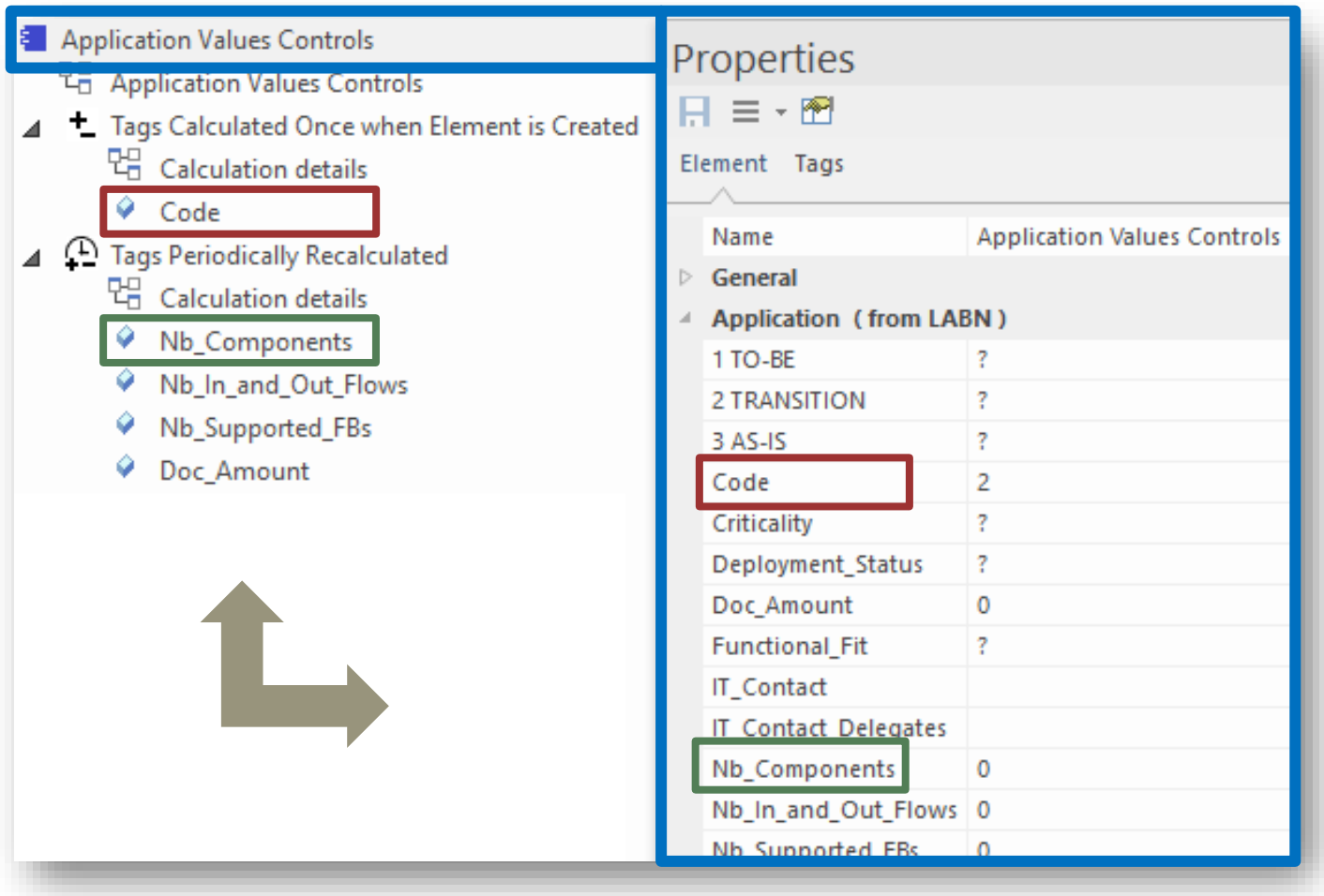
Value Calculation

1. Overview
2. Defining the type of element needing calculation
3. Scenarios for creating new value controls
4. Selecting the collection of elements to be calculated
5. Defining the tagged values to calculated
6. Making calculated tagged values read only
7. Value calculation summary
8. Triggering periodical value calculations

Tagged values to be calculated

'**Initial value calculation**' and '**periodical value calculation**' elements contain attributes.

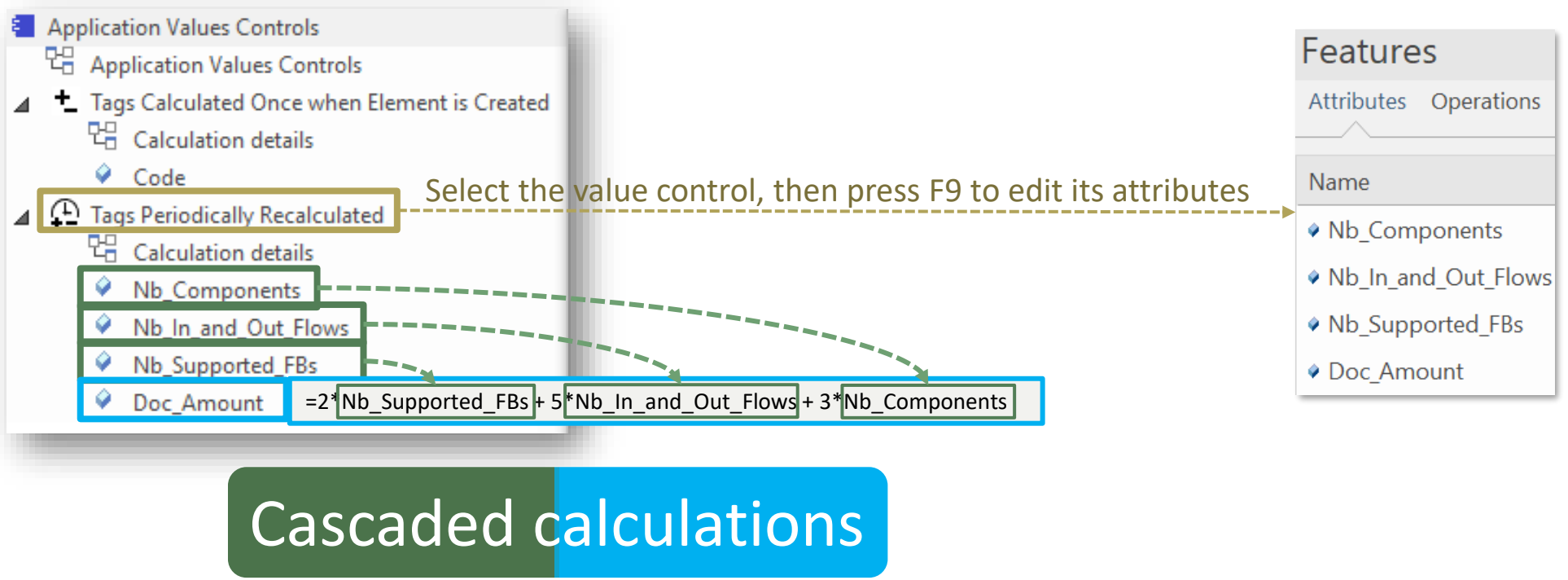
The name of the **attributes** correspond to the **tagged values** to be calculated.



The Sequence of (Cascaded) Calculations

The sequence of calculation follows the order of elements and attributes in the project browser

Use the **Ctrl-Up and Ctrl-Down keys** to change the order of elements and attributes



Cascaded calculations

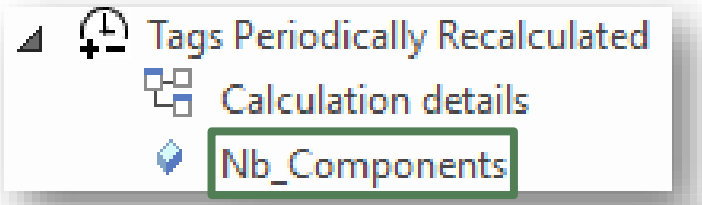
Calculation Expressions

Calculations expressions are stored in the attribute's **Notes** property, e.g. **NB_Components.Notes**

The calculation can be expressed in three different formats, depending on your needs.

Formats =>	SQL SELECT statement	Date Series Consolidation Function + SQL SELECT statement	Mathematical expression
Syntax	**SQL SELECT statement** that returns **one single row** and a **single column called** '**CALCULATED_VALUE**'.	A **function name** followed by a **SQL SELECT statement** that returns **multiple rows** and a **single column called** '**CALCULATED_VALUE**'. Function is either **SumValuesInMatchingDateSeries** or **AverageValuesInMatchingDateSeries**	={math expression}
Example	Nb_Components.Notes SELECT COUNT**(*)** AS **CALCULATED_VALUE** from **t_object** oChild where oChild**.**ParentID = **(**SELECT Object_ID from **t_object** oParent where oChild**.**Stereotype in **(**'LABN_DataStore'**,**'LABN_ApplicationComponent'**)** AND oParent**.**ea_guid **= '#CurrentElementGUID#')**	MTBF.Notes **AverageValuesInMatchingDateSeries** SELECT oApp.Name as AppName, prop.value AS **CALCULATED_VALUE** FROM t_object oApp LEFT JOIN t_objectproperties prop ON oApp.object_id = prop.object_id WHERE oApp.Stereotype = 'LABN_Application' AND oApp.Object_ID IN (SELECT Start_Object_ID FROM t_connector WHERE t_Connector.Stereotype = 'LABN_Realization' AND t_connector.End_Object_ID = **#CurrentElementID#)** AND prop.Property='MTBF'	Doc_Amount.Notes =2*Nb_Supported_FBs + 5*Nb_In_and_Out_Flows + 3*Nb_Components
Input data	Any combination of repository content including catalogs, elements, connectors, diagrams, properties/tagged values, authors…	Any combination of repository content including catalogs, elements, connectors, diagrams, properties/tagged values, authors…	Properties/tagged values that belong to a same element

SQL SELECT Statements



SQL SELECT statement that returns **one single row** and a **single column called 'CALCULATED_VALUE'.**

Notes

```
/* The number of application components and data stores in the application */

SELECT COUNT(*) AS CALCULATED_VALUE from t_object oChild where oChild.ParentID =
 (SELECT Object_ID from t_object oParent where oChild.Stereotype in ('LABN_DataStore','LABN_ApplicationComponent')
  AND oParent.ea_guid = '#CurrentElementGUID#')
```

/* Insert your comments between delimiters */

Contextual variables can be used in the SQL calculation statements:

- #CurrentElementGUID#
- #CurrentElementID#

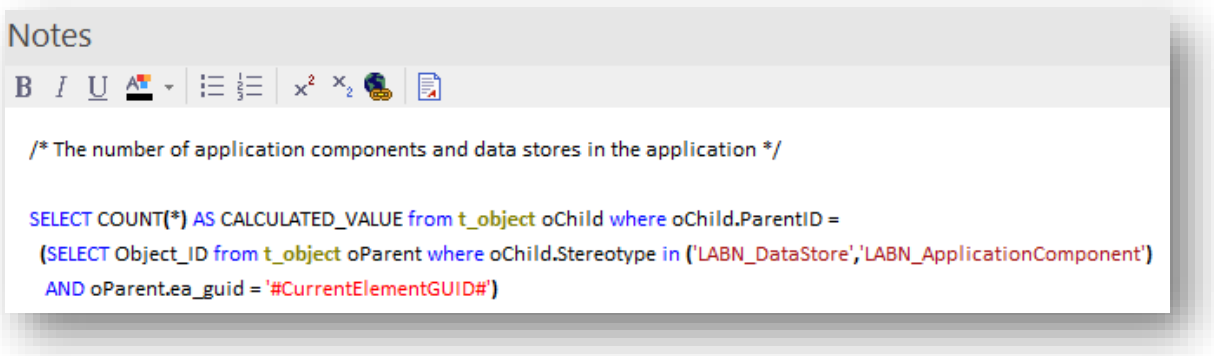
SQL SELECT Statements

Database Engine: Use SQL Server

- SQL syntax is different across database engines
- **SQL Server is very powerful including for queries**

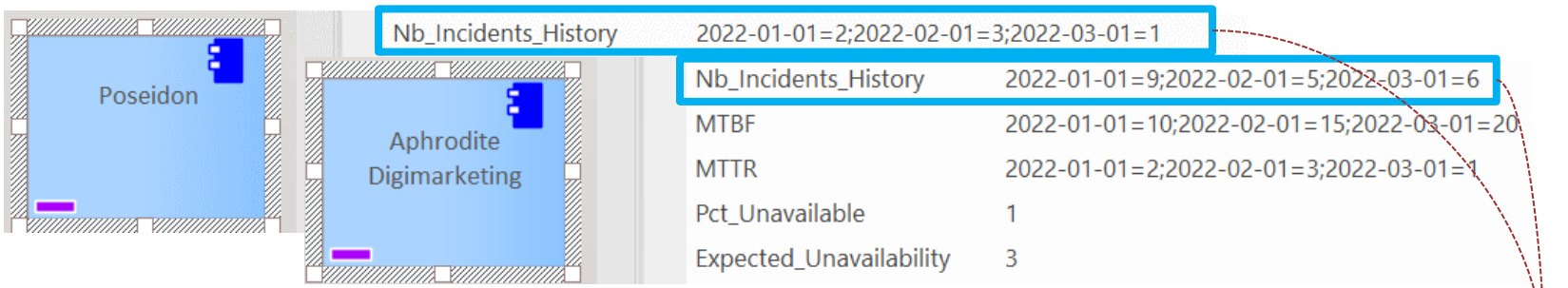
  The possibilities and combinations are almost infinite
- **SQLite is OK for queries that are not too complex**
- Access databases are **limited**

  Trying to make SQL-based value calculations in Access databases is mostly a waste of time



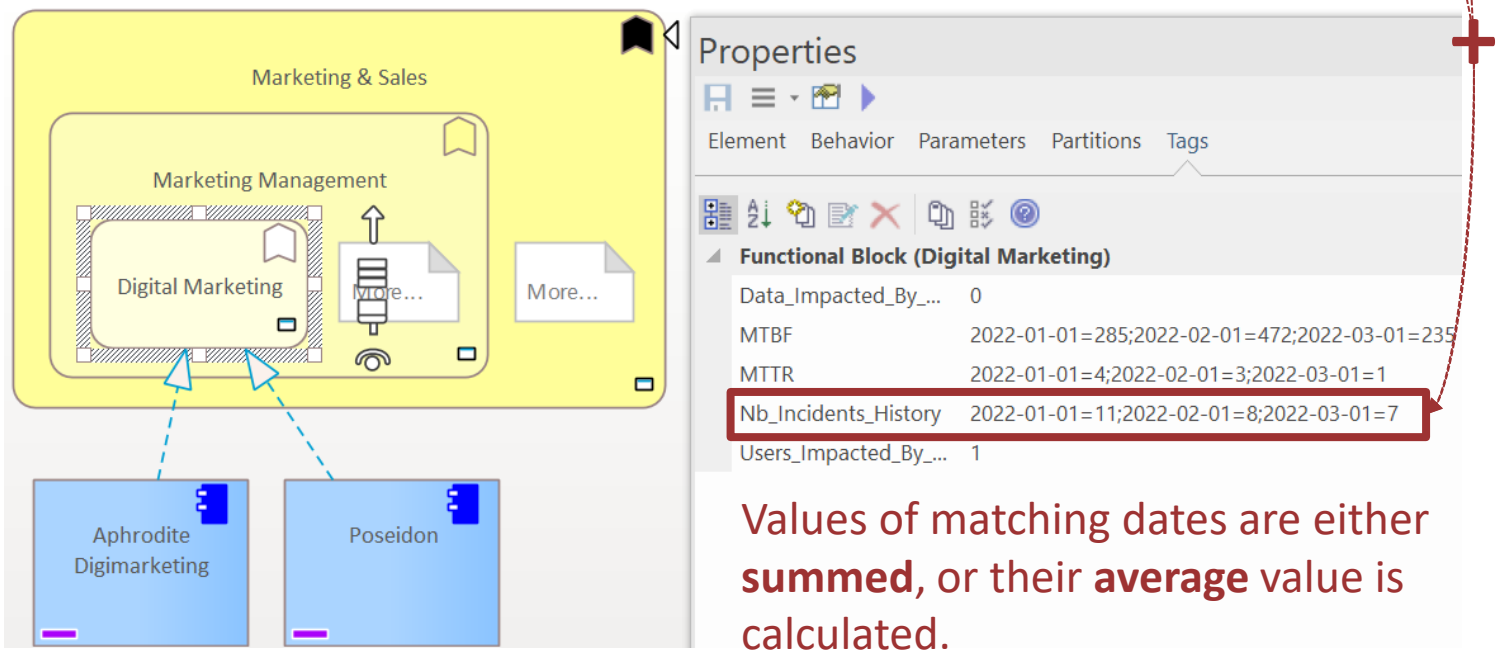
Date Series Consolidation Functions

- **Definition**: A date series is a series of values of a quantity obtained at successive dates.
- **Format:** Labnaf Date Series tagged value format is: “YYYY-MM-DD=NumericValue; YYYY-MM-DD=NumericValue; …”
- **Sample Application Date Series:**



- **Multiple date series can be consolidated.**

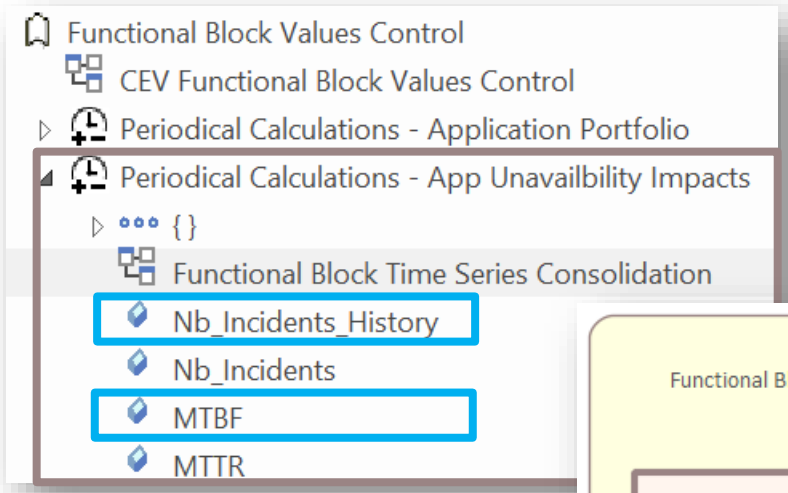
For example, the property **Nb_Incidents_History** from multiple applications realizing a same functional block can be consolidated into a single **Nb_Incidents_History** for that functional block.



Values of matching dates are either **summed**, or their **average** value is calculated.

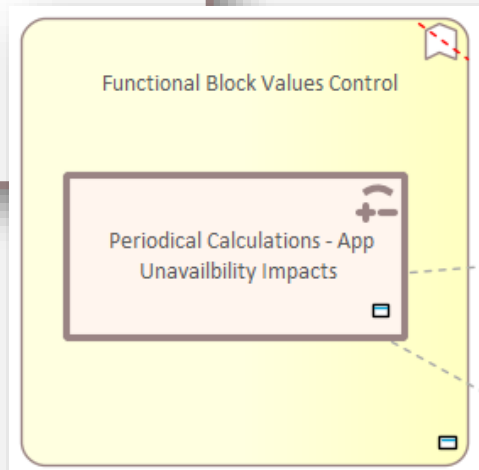
SumValuesInMatchingDateSeries and AverageValuesInMatchingDateSeries

Sample Date Series consolidation for the properties **Nb_Incident_History** and **MTBF**



Each select statement selects multiple application date series properties to be consolidated..

For **NB_Incidents_History**, values are summed for the matching dates.
For **MTBF**, the average values are calculated for the matching dates.



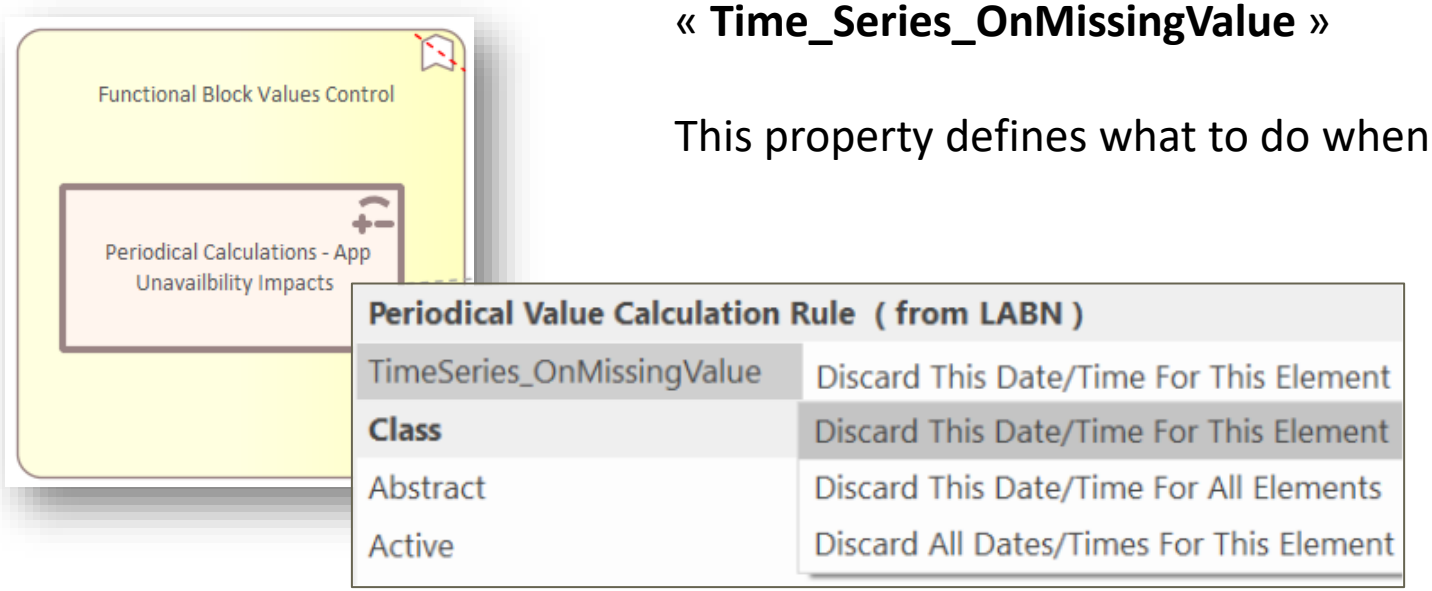
The attribute's Notes that contain the calculation

```
Nb_Incidents_History
SumValuesInMatchingDateSeries
SELECT oApp.Name as AppName, prop.value AS CALCULATED_VALUE /* Direct realization of any type of EF by applications */
FROM t_object oApp LEFT JOIN t_objectproperties prop ON oApp.object_id = prop.object_id
WHERE oApp.Stereotype = 'LABN_Application'
AND  oApp.Object_ID IN
            (SELECT Start_Object_ID FROM t_connector
             WHERE t_Connector.Stereotype = 'LABN_Realization'
             AND  t_connector.End_Object_ID = #CurrentElementID#)
AND prop.Property='Nb_Incidents_History'
```

```
MTBF
AverageValuesInMatchingDateSeries
SELECT oApp.Name as AppName, prop.value AS CALCULATED_VALUE /* Direct realization of any type of EF by applications */
FROM t_object oApp LEFT JOIN t_objectproperties prop ON oApp.object_id = prop.object_id
WHERE oApp.Stereotype = 'LABN_Application'
AND  oApp.Object_ID IN
            (SELECT Start_Object_ID FROM t_connector
             WHERE t_Connector.Stereotype = 'LABN_Realization'
             AND  t_connector.End_Object_ID = #CurrentElementID#)
AND prop.Property='MTBF'
```

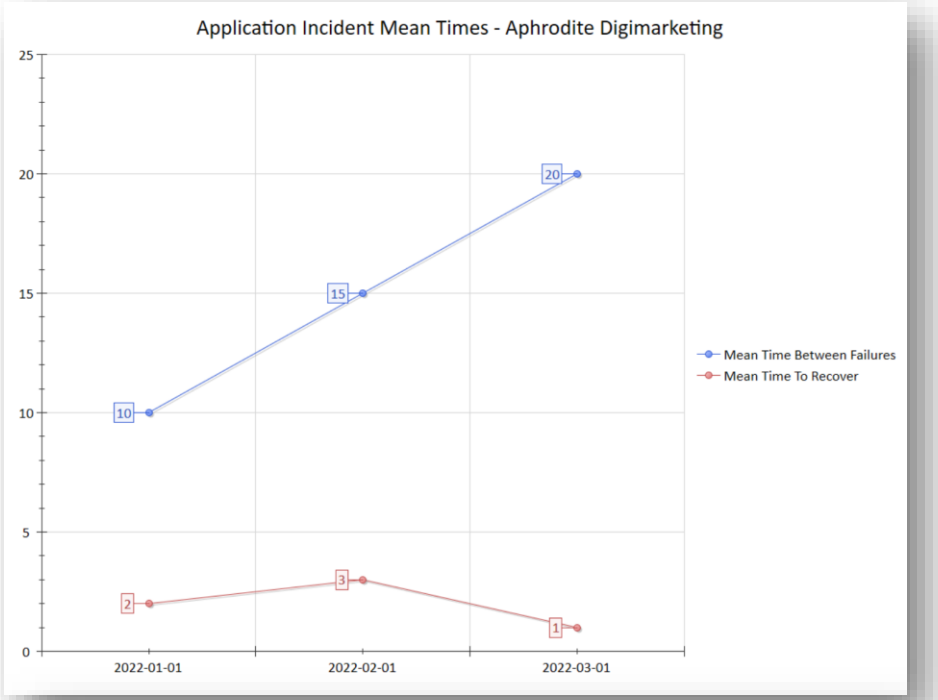
SumValuesInMatchingDateSeries and AverageValuesInMatchingDateSeries

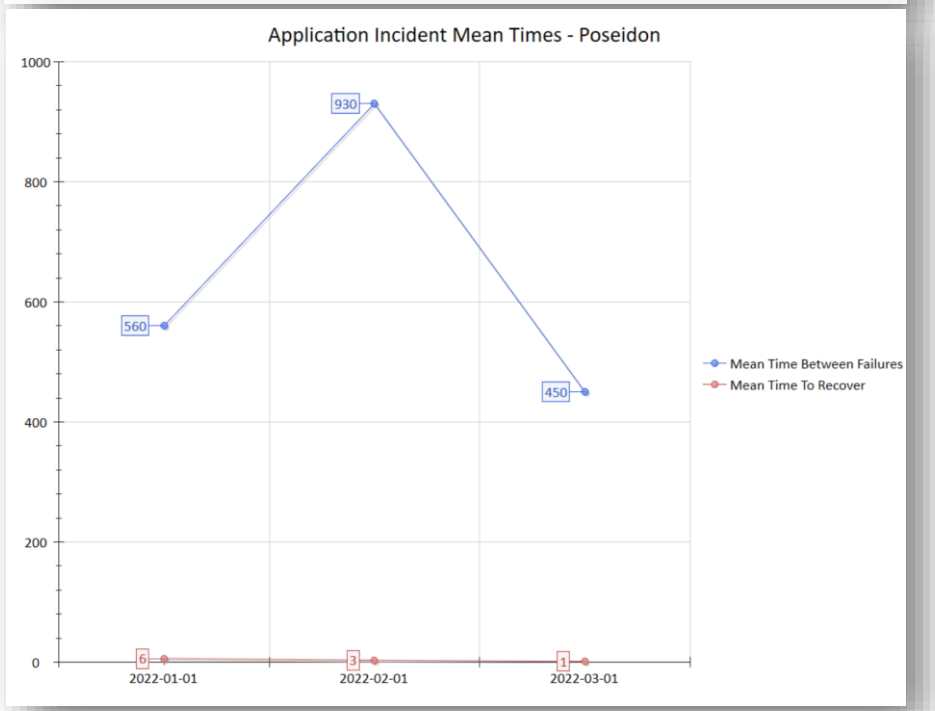
Consolidating Inconsistent Sets of Date Series



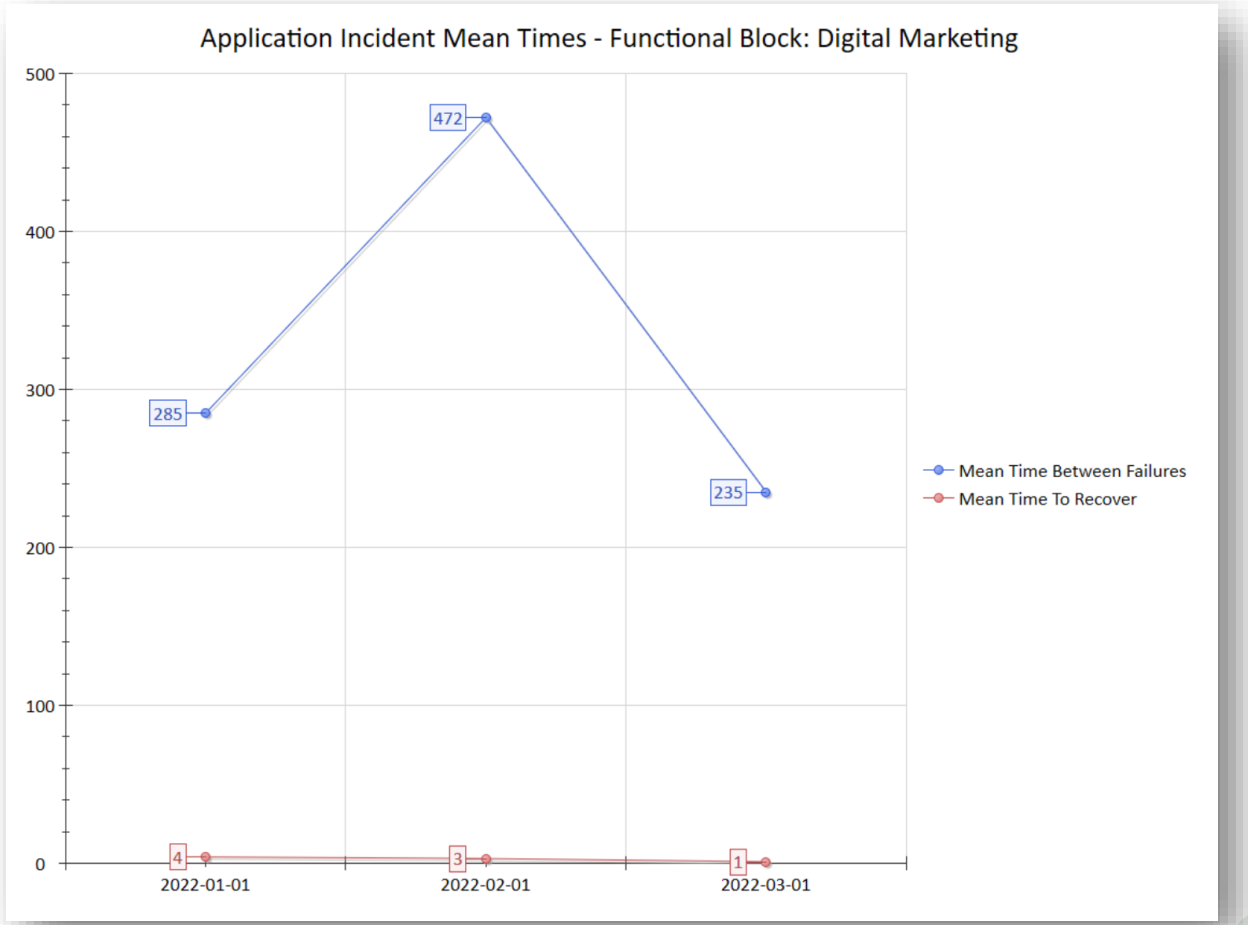
Periodical Value Calculation Rule elements contain a property called « **Time_Series_OnMissingValue** »

This property defines what to do when a date/value is missing in a time series.





Typical usage is consolidated time series chart generation



Mathematical Expressions

The attribute's **Notes** field contains the expression as follows:

"**= {Calculation Expression}**"

where {Calculation Expression} can include any of the items on the right =>

- **Property/Tagged Value names** that exist in the current element
- **Operators**:

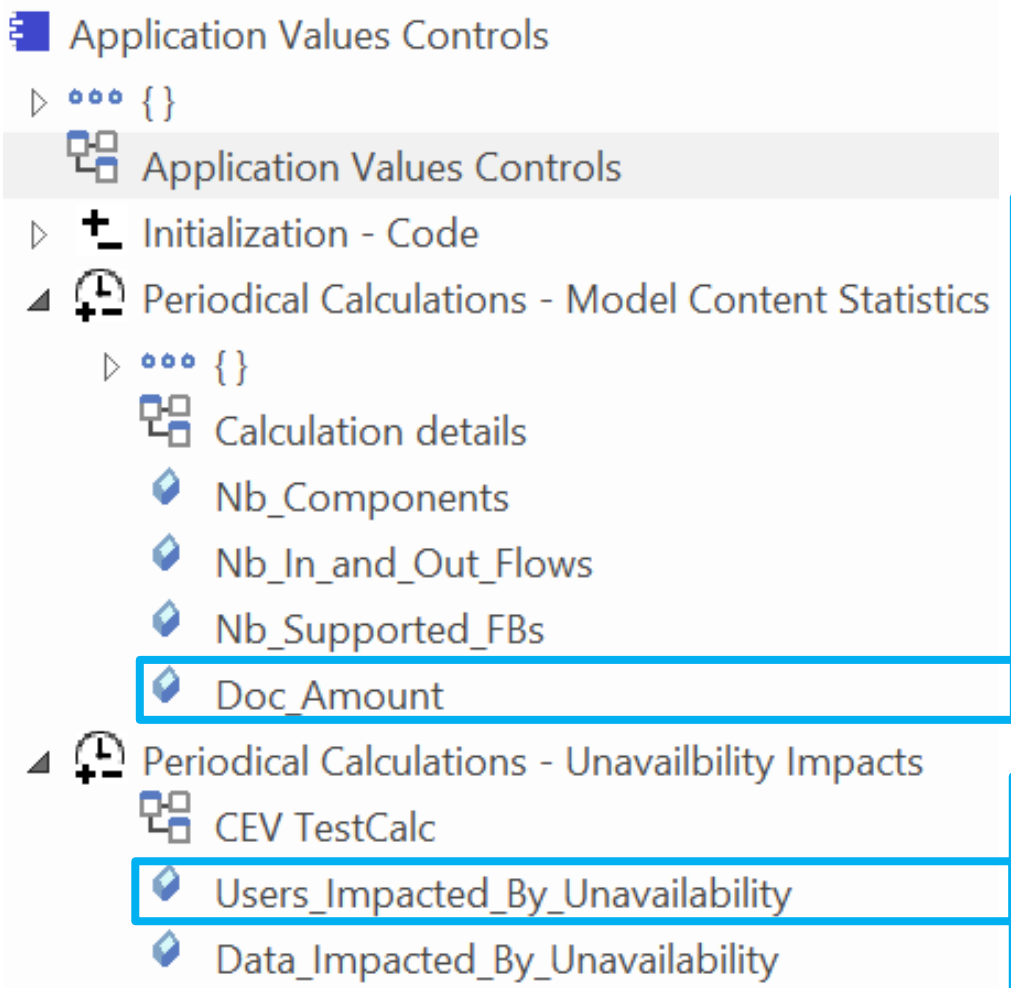
  *** / + -**

Power: x^y	Example: 2 ^ 3 = 8
Modulo: x%y	Example: 5 % 2 = 1
Factorial: x!	Example: 4! = 24
Max: x@y	Example: 5 @ 7 @ 2 = 7
Min: x\y	Example: 5 \ 7 \ 2 = 2

- **Functions**: **Round**(x), **RoundDown**(x), **Abs**(x), **Sqrt**(x)
- **Parenthesis**, which can be embedded: **(())**

Mathematical Expressions

Examples



Notes

```
/* Cascaded Calculation (Calculation based on previous calculations)
   Number of FBs: Weight=20%
   Number of flows: Weight=50%
   Number of application components and data stores: Weight=30% */

=2*Nb_Supported_FBs + 5*Nb_In_and_Out_Flows + 3*Nb_Components
```

Notes

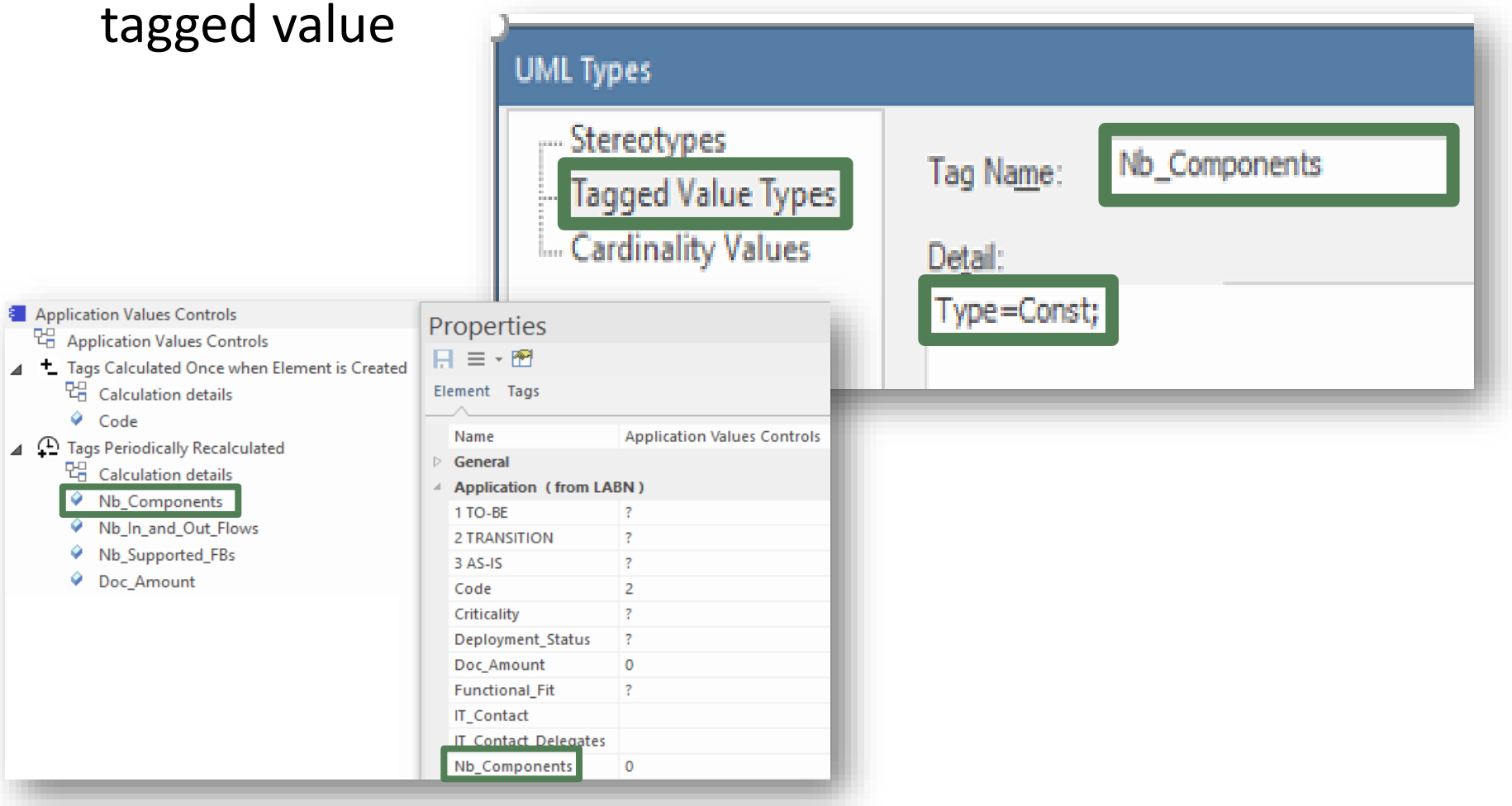
```
=Round(Expected_Unavailability*User_Community_Size/5)
```

Value Calculation

1. Overview
2. Defining the type of element needing calculation
3. Scenarios for creating new value controls
4. Selecting the collection of elements to be calculated
5. Defining the tagged values to calculated
6. Making calculated tagged values read only
7. Value calculation summary
8. Triggering periodical value calculations

Making calculated values Read Only

- ‘**Configure** (Ribbon) > **UML Types** > **Tagged Value Types**
- Add a new tag name that correspond to the name of your tagged value

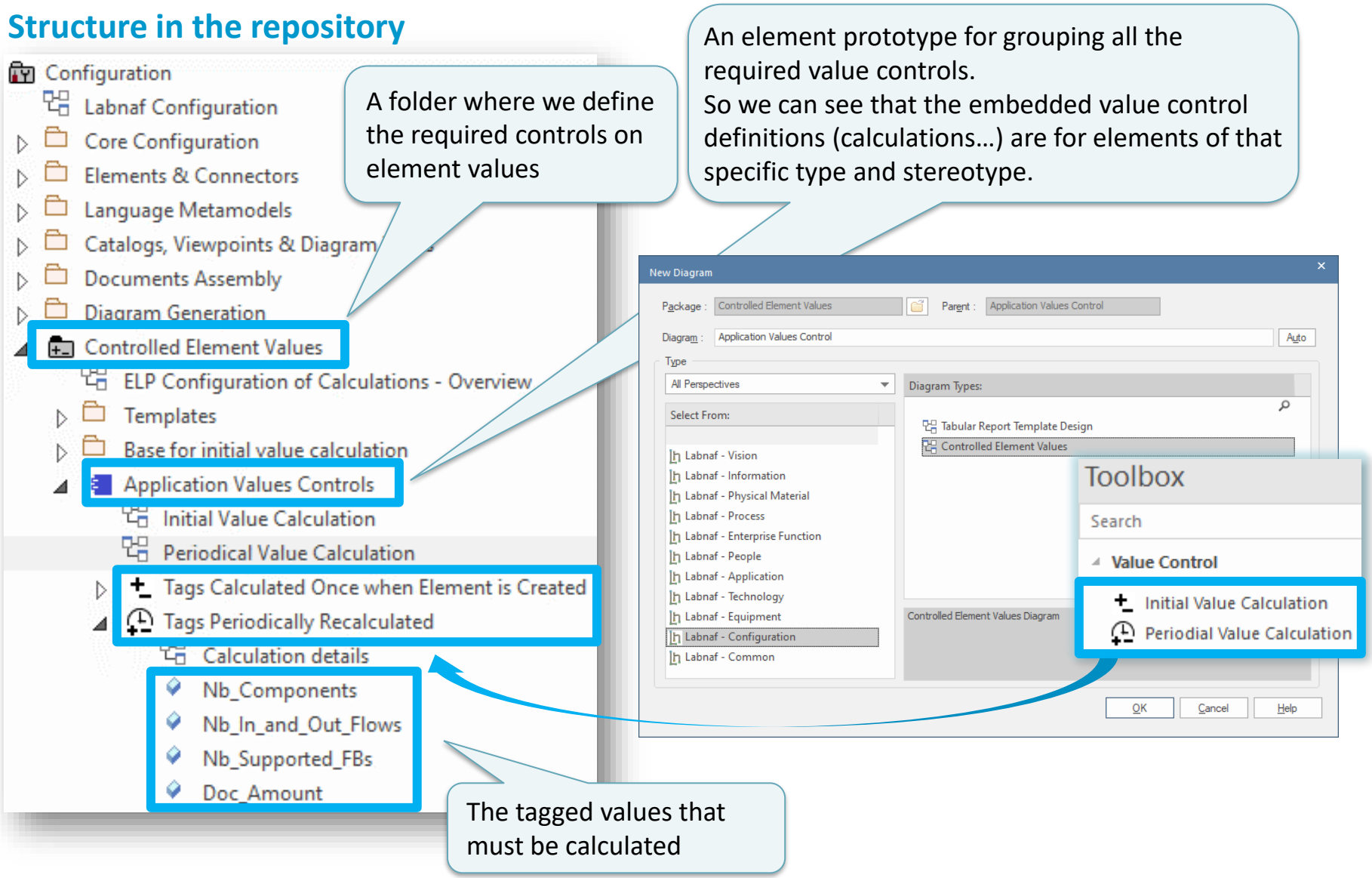


Value Calculation

1. Overview
2. Defining the type of element needing calculation
3. Scenarios for creating new value controls
4. Selecting the collection of elements to be calculated
5. Defining the tagged values to calculated
6. Making calculated tagged values read only
7. Value calculation summary
8. Triggering periodical value calculations

Value Calculation - Overview

Structure in the repository

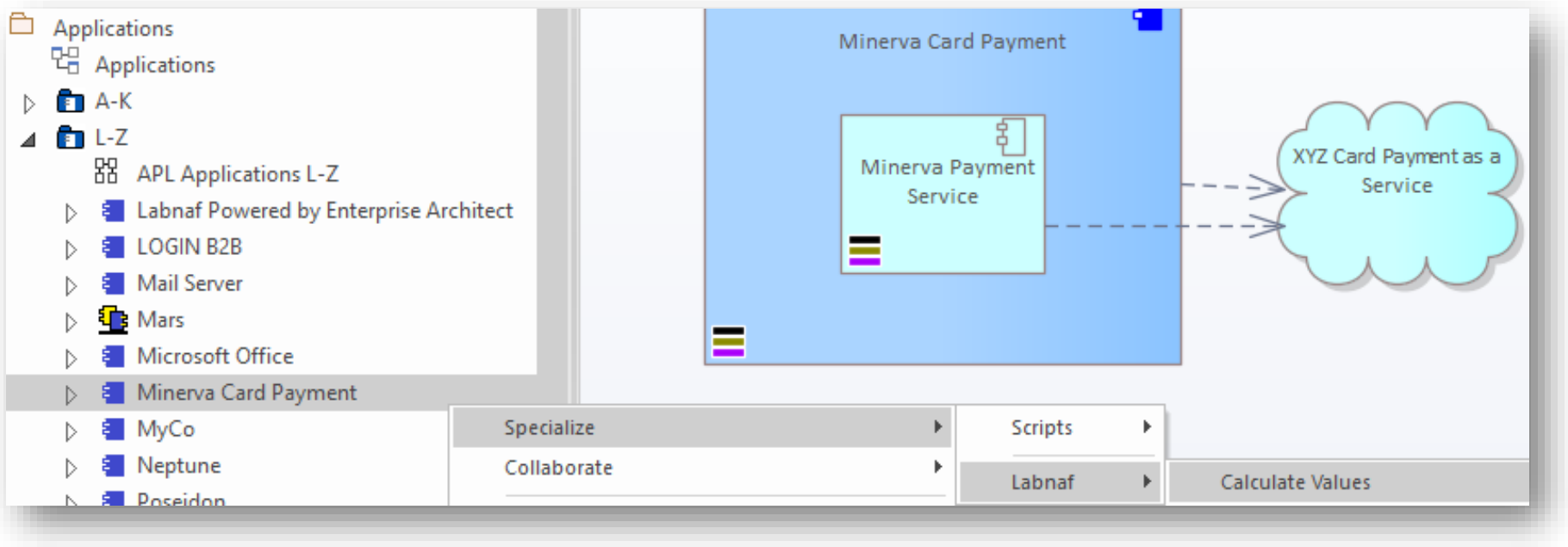


Value Calculation

1. Overview
2. Defining the type of element needing calculation
3. Scenarios for creating new value controls
4. Selecting the collection of elements to be calculated
5. Defining the tagged values to calculated
6. Making calculated tagged values read only
7. Value calculation summary
8. Triggering periodical value calculations

Testing your Periodical Value Calculation

- Right-click on any element that has a periodical value calculation associated to its type (either in the browser window or in a diagram)
- Select **Specialize > Labnaf > Calculate Values**



Scheduling Periodical Value Calculation

See the
Labnaf PowerShell
documentation