

TECHNICAL PAPER

Technique for Generating DoDAF 2 OV-3 & SV-6 Matrix Reports Using Project Specific Measurements, Resources, & Resource Flows

by Chuck Faris
Senior Enterprise Architect
IBM



This technical paper was originally published by IBM®, and has been rebranded by UNICOM® Systems in April, 2018 after it acquired System Architect® and collateral material from IBM®. It has also been updated to include material on the UNICOM® license server and make any other necessary adjustments to the rebranded software, with consent from the author.

© Copyright UNICOM® Systems 15535 San Fernando Mission Blvd, Mission Hills, CA 91345; Produced in the United States of America. All rights reserved.

UNICOM® Systems and the UNICOM® Systems logo are trademarks of UNICOM® Global. All other brands, product names, trade names, trademarks and service marks used herein are the property of their respective owners.

Contents

Executive Summary.....	4
Why this Technique is Useful in DoDAF 2.....	4
OV-03 Operational Resource Flow Matrix and SV-06 System Resource Flow Matrix .	5
Structure Used in OV-3 and SV-6 Report Generation.....	6
Workflow – Measurement Creation to Report Generation.....	7
Creating a Measurement Instance	8
Using Measurement Instances in a Resource in a Resource Flow	9
Generating High-Level Views from Low-Level Views	10
Measurement Instance – Technical View.....	12
Measurement Instance Use in Other Frameworks.....	13
Resources	14
Learn.....	14
Discuss.....	14

Executive Summary

This paper describes a technique for creating and using measurement instances in DoDAF 2 to describe resources of all types in the resource flows throughout the architecture as well as a means of reporting on those the resource flows, their resources, and the properties using the OV-3 Operational Resource Flow Matrix and SV-6 System Resource Flow Matrix reports. In short, using this technique in System Architect version 11.4.3.2 or later supports the creation and reuse of custom measurements, assignment of those measurements to resources and OV-3/SV-6 reporting of the resource flows.

Why this Technique is Useful in DoDAF 2

This technique is useful because:

- It provides a flexible workflow supporting the creation and use of custom measurements and attributes needed to describe the Resources in Resource Flows reported on in OV-3 Operational Resource Flow Matrix, SV-6 System Resource Flow Matrix and other reports
- It supports user creation of OV-05b/SV-04 and auto-generation of Exchanges (OV-06c/SV-10c), Need Lines (OV-02) and System Resource Flows (SV-01 and SV-02)
- It contains a mechanism for creation and maintenance of needed measurements and attributes without metamodel changes
- It supports the creation and reuse of DoDAF2 Resources and subtypes (data, information, materiel, system, etc.) containing the Measurement Instances and Attributes
- It supports the creation and assignment of Resources to Resource Flows (ActivityResourceOverlap, System Data Flow, Service Data Flow)
 - Other definition types containing a Resource property can have Resources assigned but they will not be included in OV-03 and SV-06 reports
- It generates OV-03 and SV-06 reports based on your Measurements, Resources and Resource Flows in prescribed structure
- It involves the following Measurement Types : Boolean, Date, Fixed-List, Numeric or Text and their Unit Of Measure
 - Fixed-List collects the list elements and provides a drop-down list of elements
- It is also used in FEA2 and can be used in other frameworks.

OV-03 Operational Resource Flow Matrix and SV-06 System Resource Flow Matrix

It is useful to include measurements in the OV-3 and SV-6 reports that you can automatically generate from System Architect. Figure 1 shows what such an OV-3 looks like. The same would be true for an SV-6.

OV-03 Need Line Top Down to Resource Collection to Measures HTML

HTML Need Line-Operational Exchange-ActivityResourceOverlap-Resource Collection-Measures

Need Line Name	Sending Performer	Receiving Performer	Sending Role	Receiving Role	Sending Activity	Receiving Activity	Operational Resource Flow Name	Operational Resource Flow Description	Resource	Type	Measurement	Value	BOM		
Forecaster Internal	Weather Forecaster	Weather Forecaster	Forecaster - Construct Forecast	Forecaster - Publish Forecast	Construct Forecast	Publish Forecast	Compiled Weather Data		Conditions for Friday Next Week	Data (DAC)	City	Boston			
											State	MA			
											Weather Condition	Clouds			
													Weather Condition	80	percent
													Probability		
													Humidity for Friday Next Week	90 - 100	percent
										City	Boston				
										State	MA				
											Temperature for Friday Next Week	85 - 90	degrees F		
										City	Boston				
										State	MA				
Forecaster - Forecast Conditions	Forecaster - Construct Forecast	Forecast Conditions	Construct Forecast	Conditions				Conditions for Friday Next Week	Data (DAC)	City	Boston				
										State	MA				
										Weather Condition	Clouds				
										Weather Condition	80	percent			
										Probability					
Forecaster to News Organization	Weather Forecaster	News Organization	Forecaster - Publish Forecast	Weather Reader - Receive Forecast	Publish Forecast	Broadcast Weather	Published Forecast		Weather Forecast for Friday Next Week	Data (DAC)	Temperature	85 - 90	degrees F		
											Humidity	90 - 100	percent		
											City	Boston			
										State	MA				
											Weather Condition	Clouds			
											Weather Condition	80	percent		
											Probability				
Weather Station to Forecaster	Remote Weather Station	Weather Forecaster	Boston - Humidity	Forecaster - Construct Forecast	Capture Humidity	Construct Forecast	Humidity		Humidity for Friday Next Week	Data (DAC)	Humidity	90 - 100	percent		
											City	Boston			
											State	MA			
											Temperature for Friday Next Week	85 - 90	degrees F		
											City	Boston			
											State	MA			

Figure 1. Automatically generated System Architect OV-3 Operational Resource Flow Matrix based on custom Measurements used in Resources in Resource Flows and generated Need Lines

Structure Used in OV-3 and SV-6 Report Generation

The structure used in OV-3 and SV-6 Report Generation is shown in Figure 2 and described below.

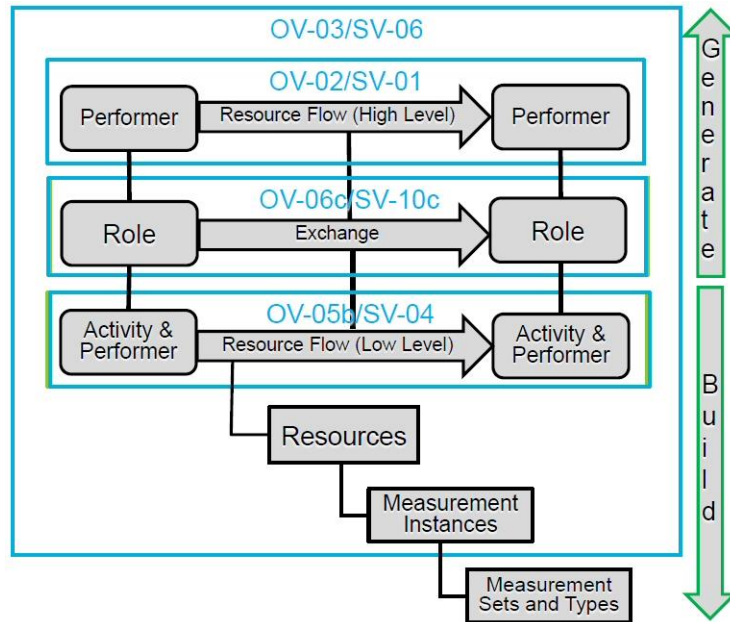


Figure 2. Structure used in OV-3 and SV-6 report generation.

At the highest level – which is the Resource Flow-level -- you have, depending on the domain:

- Needlines (Operational View)
- System Resource Flows (System View)

Below that highest-level you have Exchanges. Depending on the domain it will have:

- Operational Exchanges, or
- System Exchanges

At the lowest level, you have the Resource Flow view. Depending on the domain it will have:

- ActivityResourceOverlaps (Operational View)
- System Data Flows (System View)
- Service Data Flows (Services View)

Resources are applied to the lowest level – the Resource Flow level. The DoDAF 2 structure for Resource is as follows:

- Resource
 - Information
 - Data
 - Architectural Description
 - Domain Information
- Materiel
- Performer
 - System
 - Service
 - Service Interface
 - Person
 - Organization
 - Interface (Port)

Activities (of different types) transform resources from one form to another. The DoDAF 2 Activity subtype structure is as follows:

- Activity
 - System Function
 - Service Function

Workflow – Measurement Creation to Report Generation

The workflow for applying measurement instances to Resources and generating applicable reports is as follows:

1. Create the needed Measurement Sets, Types, and Values.
2. Create the Resources that will flow in the Resource Flows (use any DoDAF 2 Resource subtype: Data, Information, and Materiel are a few of the subtypes).
3. Assign Resources to low-level Resource Flows.
 - Low-Level Resource Flows are: ActivityResourceOverlaps, System Data Flows, and Service Data Flows.
4. Create and assign Measurement Instances to the Resources in Low-Level Resource Flows so they are included in the OV-3 and SV-6 reports. You can do this in two ways:
 - Top-down by working in a Resource and creating Measurement Instances as needed, or
 - Bottom-up by creating the Measurement Instances, then using them in Resources.

5. Using System Architect DoDAF 2 Utilities under the Tools menu, generate Operational Exchanges and Needlines, or System Exchanges and System Resource Flows, or Service Resource Flows from low-level Resource Flows.
6. Rename generated Resource Flows, Exchanges, and Roles if needed.
7. Using the DAF2.rpt report file of the System Architect Report Generator, generate an OV-3 Operational Resource Flow Matrix or an SV-6 System Resource Flow Matrix.

These steps can be done in many sequences – for example after generating an OV-3 report you can add other Resources to Resource Flows or other Measurement Instances to Resources and rerun the OV-3.

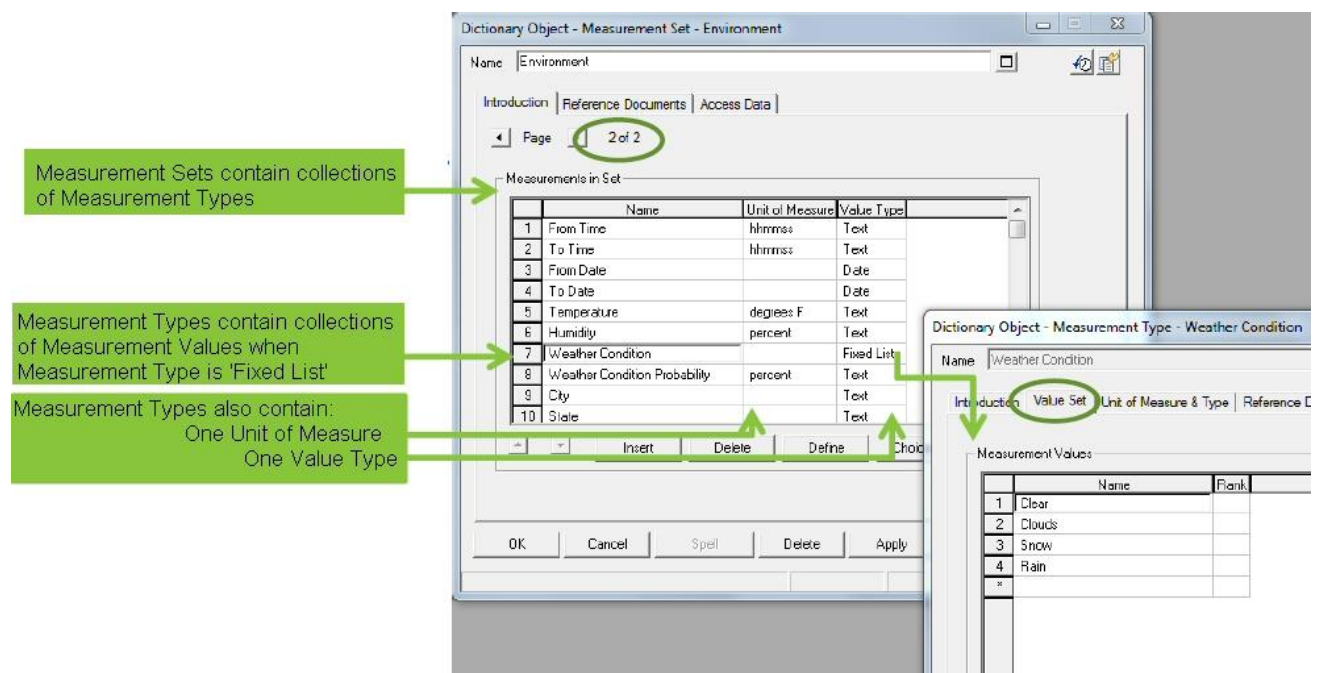


Figure 3. Select options for System Architect configuration

Creating a Measurement Instance

You create a Measurement Instance as follows (as shown in Figure 4):

1. Create a Measurement Instance definition.
2. Within it, select the Measurement Set.
3. Select the Measurement Type based on the restricted list presented.

4. Specify the Value Type.
5. Select a Value from the fixed list presented or fill in the Value depending on Value Type.
6. The Report Value will be populated automatically when you click the OK button.

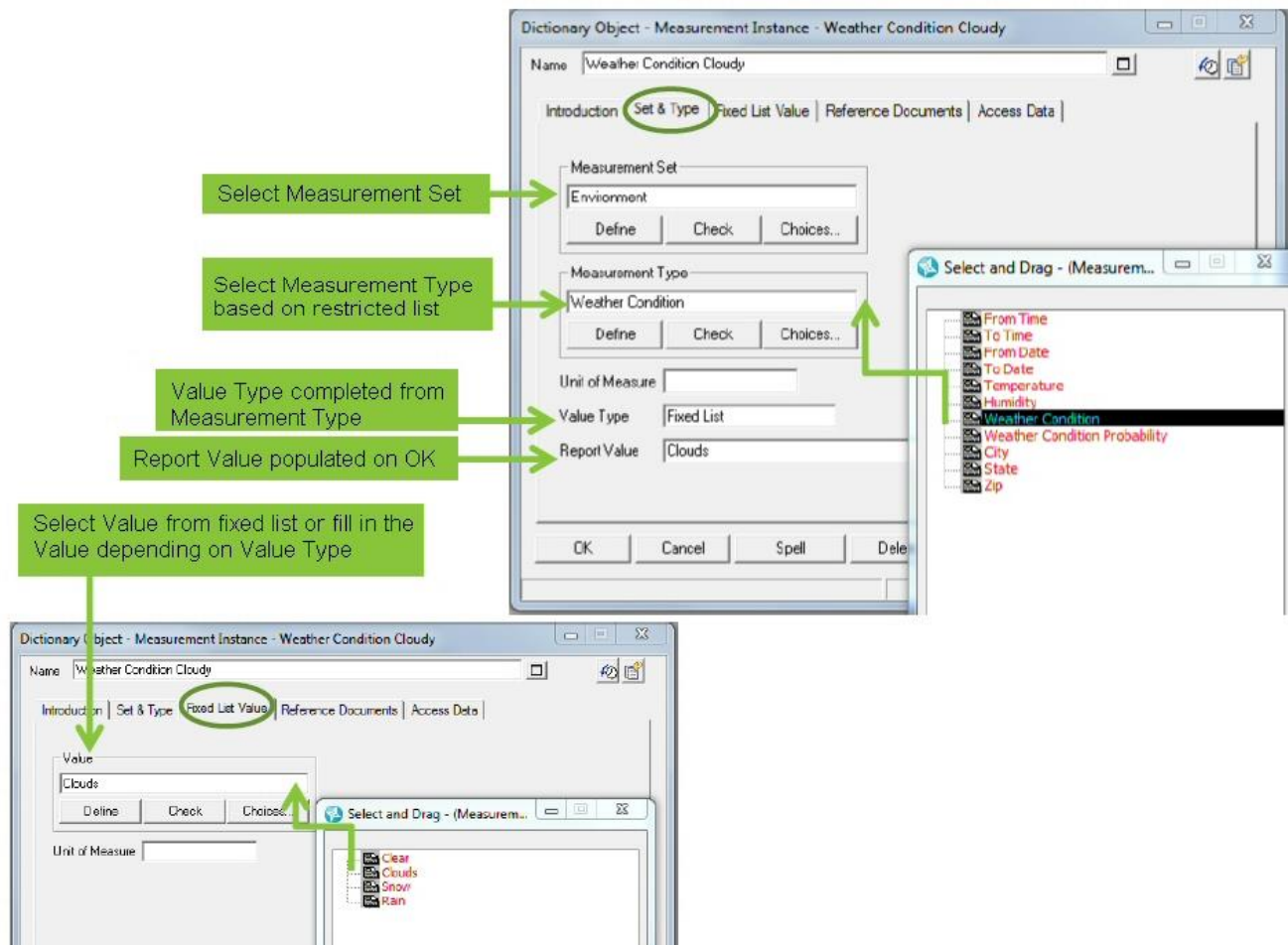


Figure 4. Select options for System Architect configuration

Using Measurement Instances in a Resource in a Resource Flow

DoDAF2 Resources, including all Resource subtypes (data, information, material, person, etc.), can contain Measurement Instances and can be assigned to Resource Flows. Figure 5 shows how this is done:

1. Open or create a Resource that will be assigned to a Resource Flow.
2. Add the Measurement Instances you need to the Resource. Measurement Instances can be created in their containers, top-down, or they can be reused.
3. Assign the Resource to an ActivityResourceOverlap System Data Flow, or Service Data Flow.

Note: Measurements are also enabled in all DoDAF2 and FEAF2 definitions – as the DoDAF specification says, everything can be measured.

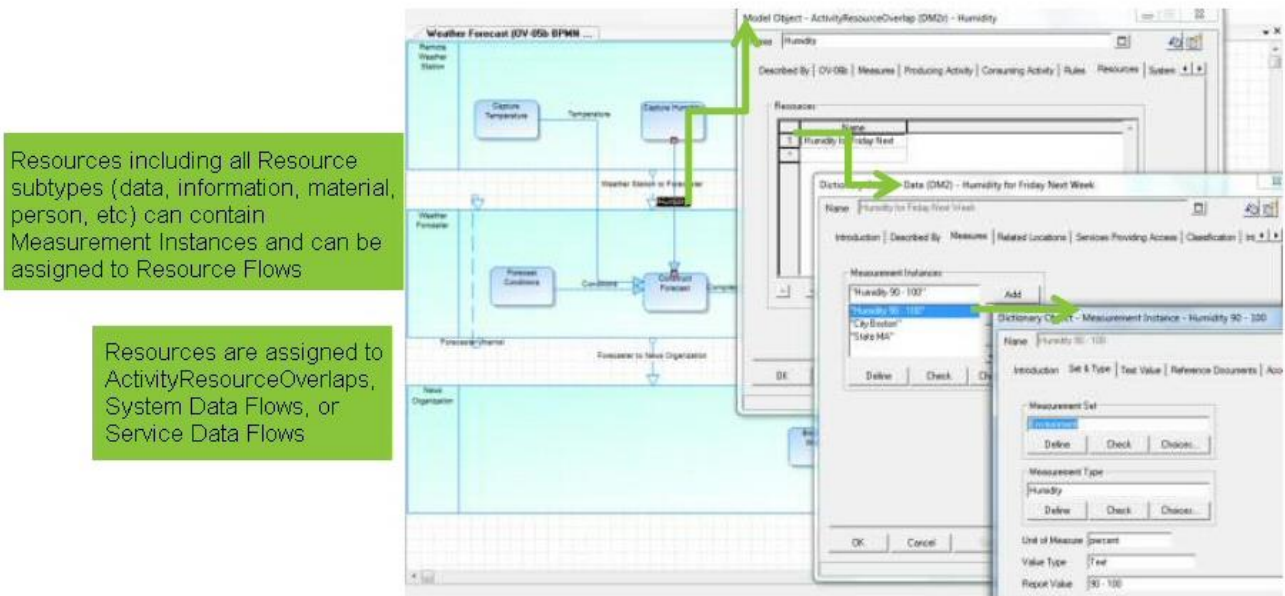


Figure 5. Select options for System Architect configuration

Generating High-Level Views from Low-Level Views

System Architect’s automatic generation from low-level views to high-level views (ActivityResourceOverlap to Needline for example) supports traceability and drill-down access to low-level Resource Flows, their Resources, and their Measurement Instances.

An example is shown in Figure 6. Steps to creating the architecture are as follows:

1. Create a low-level view such as an OV-5b BPMN Operational Activity Model populated with Sequence/Message Flows (DoDAF2 ActivityResourceOverlap)

2. Generate Operational Exchanges and Needlines using the DoDAF2 Utilities (Tools > DoDAF2 Utilities).
3. Refresh the diagram and the generated Need Lines become visible.
4. Open one of the Needlines and drill down through Operational Exchange, ActivityResourceOverlap, and Resource to see Measurements.
5. Generate the OV-3 Operational Resource Flow Matrix (available in the DAF2.rpt file in the Reports Generator (Reports > Report Generator)).

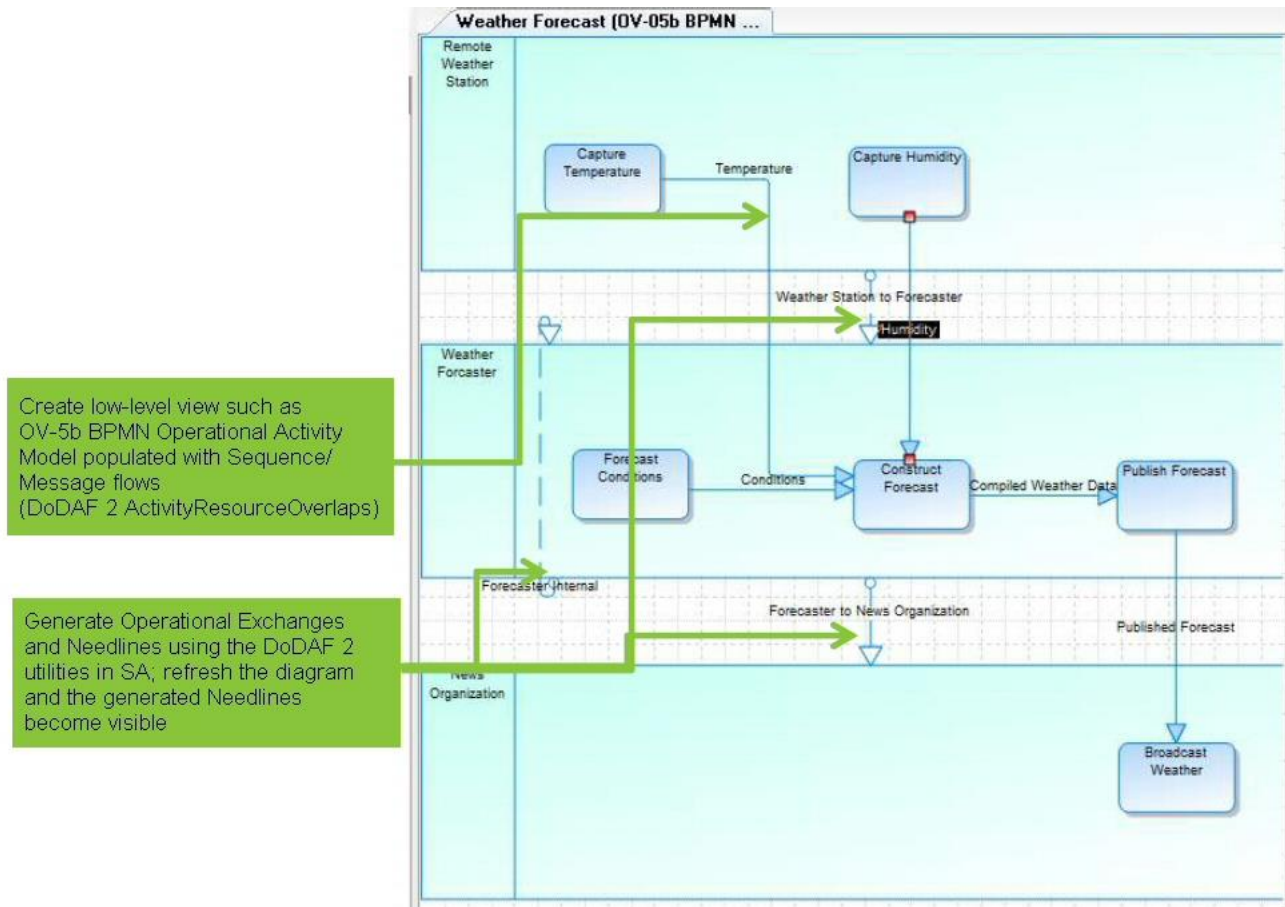


Figure 6. Select options for System Architect configuration

Measurement Instance – Technical View

The metamodel of the DoDAF 2 Measurement structure is as follows, and pictured in Figure 7:

Measurement Instance contains:

- One Measurement Set
- One Measurement Type restricted based on selected Measurement Set
- One Value Type and One Unit of Measure (UOM) based on Measurement Type
 - For “Fixed List” Value Type, Value is restricted based on selected Measurement Type
 - For Value Types Boolean, Date, Numeric or Text Value Type, the Value is not read only and is user entered

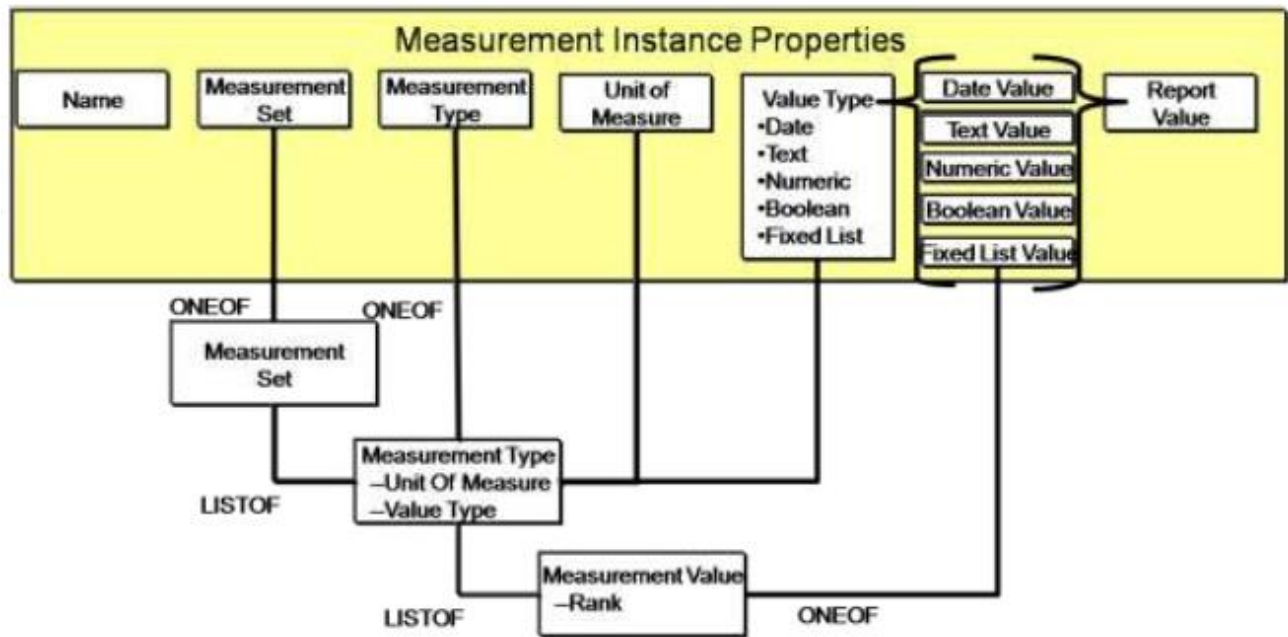


Figure 7. Select options for System Architect configuration

Measurement Instance Use in Other Frameworks

Measurement Instance use is preset in DoDAF2 and FEAF2. You can add Measurement Instances to a framework other than DoDAF 2 or FEAF 2 (such as TOGAF or ArchiMate). To do so, perform the following steps:

1. Add the following to any definition type in the USRPROPS.TXT file:

```
PROPERTY "Measurement Instances"{ EDIT ListOf "Measurement Instance" .....}
```

Note that:

- The property name must be “Measurement Instances” to populate the Report Value property from the type specific values.
 - The referenced definition type must be “Measurement Instance” to access the correct definition type and its supporting definition types.
2. Enable Measurement Sets on the advanced page of the System Architect Configuration dialog.
 3. Use the Report Value property for reports since it will contain each Measurement Instance’s value regardless of its Value Type.

Resources

Note that a powerpoint version of this paper is provided on Slideshare here:
<https://www.slideshare.net/chuckfaris/measurements-in-dm2-resource-flows-v2>

Learn

- Learn more about UNICOM® System Architect®:
 - Browse the product page UNICOM® System Architect® at <http://unicomsi.com/products/system-architect/>
 - Watch videos of UNICOM® System Architect® on its YouTube channel, [SystemArchitectTrain](#).
 - Explore the [UNICOM Customer Support Portal](#) for System Architect technical resources and best practices.

Discuss

- Participate in the [System Architect LinkedIn forum](#), where you can share information about methods, frameworks, and tool implementations. Discussions include tool-specific technical discussions about System Architect.
- Follow UNICOM® System Architect® on:
 - [Facebook.com\SystemArchitect](#),
 - [Twitter](#) (@SystemArchitect), and
 - [YouTube \(SystemArchitectTrain\)](#),

and add your comments and requests.