CAMEO
ENTERPRISE ARCHITECTURE™
UPDM / DoDAF / MODAF / SysML / BPMN / SoaML

DoDAF to UPDM

migration manual

No Magic, Inc.
2013

CONTENTS

DODAF MIGRATION TO UPDM 1.x

1. Basic Concepts and Instructions

The Unified Profile for DoDAF/MODAF (UPDM) is the standard of an Object Management Group (OMG) initiative to develop a modeling standard that supports both the US Department of Defense Architecture Framework (DoDAF) and UK Ministry of Defense Architecture Framework (MODAF). As UPDM supports both frameworks, it is particularly important for those used to DoDAF 1.5 or any earlier version to clear some misunderstandings.

The purpose of DoDAF migration to UPDM 1.x is to convert DoDAF 1.5 metamodel projects into UPDM 1.x metamodel projects without loosing data.

Migration to UPDM 1.x is not backward compatible, so before proceeding with migration we highly recommend that you back-up your DoDAF projects.

2. How to Use this Manual

The purpose of this manual is to help you understand the differences between UPDM 1.x and DoDAF 1.5 or any earlier versions, guide you along data mapping and explain how you as a user can have a say in the mapping process.

This manual is divided into two major sections:

> "3. DoDAF Projects Migration" on page 4 explains data mapping, addresses possible mapping issues and offers solutions.
>
> "4. Migrating Server Projects" on page 15 explains how to migrate Teamwork projects.
>
> "5. Customizing Migration" on page 16 explains how you can change the way the data is mapped by default.

It is highly recommended that you read the first part of the manual before migrating DoDAF projects to UPDM 1.x.

3. DoDAF Projects Migration

This section includes three sub-sections: "3.1 Products Mapping" on page 5; "3.2 Data Mapping" on page 7; and "3.3 Data Mapping Problems and Solutions" on page 14.

Most of the data mapping problems you may encounter come from the fact that certain concepts and model names in DoDAF 1.5 standard differ in UPDM 1.x standard. If you are a DoDAF user and are not familiar with UPDM concepts, we strongly recommend that you first read this document and UPDM specifications before proceeding with migration.

DoDAF 1.5 models in MagicDraw are represented in the forms of diagrams, matrixes or tables. As explained in sub-section 3.1 below (Product Mapping), while representation formats have not changed, separate product representation has.

Please also note that all reports differ in DoDAF and UPDM. If you are familiar with report templates and velocity language, you can make DoDAF reports compatible with UPDM ones manually by changing the names of the stereotypes used to gather project data. However, if you are not, we recommend that you use predefined UPDM reports.

3.1 Products Mapping

Products mapping converts DoDAF products into UPDM ones. It changes a product's name and representation form whenever required by the UPDM 1.1 plugin. DoDAF product migration is described in details in Table 1 on page 6 below.

TABLE 1. **DoDAF Migration to UPDM 1.x**

DoDAF 1.5		UPDM 1.x	
Product	**Representation form**	**Product**	**Representation Form**
OV-1 High-Level Operational Concept Graphic	UML Class Diagram	OV-1 High-Level Operational Concept Graphic	UML Composite Structure Diagram
OV-2 Operational Node Connectivity Description	UML Class Diagram	OV-2 Operational Node Relationship Description	UML Class Diagram
OV-3 Operational Information Exchange Matrix	Report	OV-3 Operational Information Exchange Matrix	Table
OV-4 Organizational Relationships Chart	UML Class Diagram	OV-4 Organizational Relationships Chart	UML Class Diagram
OV-5 Operational Activity Model	UML Class Diagram UML Activity Diagram	OV-5 Operational Activity Model	UML Class Diagram UML Activity Diagram
OV-6a Operational Rules Model	Report	OV-6a Operational Rules Model	Table
OV-6b Operational State Transition Description	UML State Machine Diagram	OV-6b Operational State Transition Description	UML State Machine Diagram
OV-6c Operational Event-Trace Description	UML Sequence Diagram	OV-6c Operational Event-Trace Description	UML Sequence Diagram
OV-7 Logical Data Model	UML Class Diagram	OV-7 Information Model	UML Class Diagram
SV-1 Systems Interface Description	UML Class Diagram UML Composite Structure Diagram	SV-1 Resource Interaction specification	UML Class Diagram UML Composite Structure Diagram
SV-2 Systems Communications Description	UML Class Diagram UML Composite Structure Diagram	SV-2 Resource Communications description	UML Class Diagram UML Composite Structure Diagram
SV-3 Systems-Systems Matrix	Matrix	SV-3 Resource Interaction Matrix	Matrix
SV-4 Systems Functionality Description	UML Class Diagram UML Activity Diagram	SV-4 Functionality Description	UML Class Diagram UML Activity Diagram
SV-5 Operational Activity to Systems Function Traceability Matrix	Matrix	SV-5 Function to Operational Activity Traceability Matrix	Editable Matrix
SV-6 Systems Data Exchange Matrix	Report	SV-6 Systems Data Exchange Matrix	Table
SV-7 Systems Performance Parameters Matrix	Table	SV-7 Resource Performance Parameters Matrix	Actual and Typical Tables

DoDAF 1.5		UPDM 1.x	
Product	**Representation form**	**Product**	**Representation Form**
SV-8 Systems Evolution Description	UML State Machine Diagram	SV-8 Capability Configuration Management	UML Class Diagram
SV-9 Systems Technology Forecast	Table	SV-9 Technology and Skills Forecast	Table
SV-10a Systems Rules Model	Report	SV-10a Resource Constraints Specification	Table
SV-10b Systems State Transition Description	UML State Machine Diagram	SV-10b Resource State Transition Description	UML State Machine Diagram
SV-10c Systems Event-Trace Description	UML Sequence Diagram	SV-10c Resource Event-Trace Description	UML Sequence Diagram
SV-11 Physical Schema	UML Class Diagram	SV-11 Physical Schema	UML Class Diagram
TV-1 Technical Standards Profile	Table	Does not map	-
TV-2 Technical Standards Forecast	Table	TV-2 Standards Forecast	Table

Do not be surprised, if you will not find any OV-3, OV-6a, SV-6 and SV-10a reports after migrating to UPDM 1.x. These reports do not exist in UPDM 1.x. There are tables instead. They can be easily filled with data, conveniently updated, and printed as any other MagicDraw diagrams.

These tables, however, will not be created automatically after migrating to UPDM 1.x. We suggest you to create them manually by selecting the migrated data you would like to see in each one.

3.2 Data Mapping

Data mapping mostly has to do with migrating elements and their relationships to UPDM. As mentioned in section 3 above, there are a lot of conceptual changes between DoDAF and UPDM projects. Table 2 below describes entity mapping and Table 3 relationships mapping.

TABLE 2. **Entity Mapping**

DoDAF 1.5	UPDM 1.x	Properties	Comments
Asset [Class]	ConceptRole [Property]	*actualImplementation* to *type* *representationType*. Data is stored in element's ToDo property. (See "3.3 Data Mapping Problems and Solutions" on page 14.)	Concept role is placed under the High Level Operational Concept element.
Capability [UseCase]	Capability [Class]	*mission*: Data is stored in element's ToDo property. *operationalThread* to *ownedBehavior.*	
CommunicationsNetwork [Class]	System [Class]	*securityClassification*: Data is stored in element's ToDo property.	
CommunicationsPath [Class]	System [Class]	*communicationLinks*: Data is stored in element's ToDo property.	
CommunicationsSystem [Class]	System [Class]		
Hardware/ SoftwareItem [Class]	System [Class]	*vendors/source*: Data is stored in element's ToDo property.	
InformationElement [Class]	InformationElement [Class]		*ImplementsOperational* relationship is created between *Information Element* and its implementer if any.
InformationExchange [InformationFlow]	InformationExchange [InformationFlow]	*consumedBy*: Data is stored in element's ToDo property. *producedBy*: Data is stored in element's ToDo property. *mission*: Data is stored in element's ToDo property.	
LAN [Class]	System [Class]		
MAN [Class]	System [Class]		
Milestone [State]	Configuration Deployed[InstanceSpecification]	*version*: Data is stored in element's ToDo property. *timePeriod* to *date.*	
Objective [UseCase]	EnterpriseGoal [Class]		

DoDAF 1.5	UPDM 1.x	Properties	Comments
OperationalActivity [Activity]	OperationalActivity [Activity]	*capability*: Data is stored in element's ToDo propertyconsumes does not map. *levelIdentifier*: Data is stored in element's ToDo propertyperformedAt does not map. *produces*: Data is stored in element's ToDo property.	ImplementsOperational relationship is created between OperationalActivity and its implementer if any.
OperationalActivityAction [CallBehaviorAction]	OperationalActivityAction [CallBehaviorAction]		
OperationalNode [Class]	OperationalNode [Class]	*levelIdentifier*: Data is stored in element's ToDo property*operationalRole* does not map. *performers*: Data is stored in element's ToDo property.	If *isExternal* is true *ExternalNode* should be created instead of *OperationalNode*.
OperationalNodeUsage [Property]	NodeRole [Property]	*operationalRole*: Data is stored in element's ToDo property.	
OperationalRule [Constraint]	OperationalRule [Constraint]	*type* to *kind.*	
Organization [Class]	ActualOrganization [InstanceSpecification]	*responsibility*: Data is stored in element's ToDo property.	
OrganizationalResourceUsage [Property]	SubOrganization [Property]		
OrganizationType [Class]	Organization [Class]		
PerformanceMeasurement [Slot]	ActualMeasurement [Slot]		
PerformanceMeasurementSet [InstanceSpecification]	ActualMeasurementSet [InstanceSpecification]	*measuredSystem* to *measuredElement.*	
PerformanceParameterSet [Class]	MeasurementSet [Class]	*measuredSystems* to *measuredElement.*	
PerformanceParameterType [Property]	Performance Parameter [Property]	*objectiveValue* to *propertyValue* *thresholdValue* to *maxValue.* *unitOfMeasure*: Data is stored in element's ToDo property.	

DoDAF 1.5	UPDM 1.x	Properties	Comments
Performer [Class]		performs: Performs relationship are created between performer and performed Elements.	
Person [Class]	ActualPerson [InstanceSpecification]	responsibilities: Data is stored in element's ToDo property. roles: Data is stored in element's ToDo property.	
ReferenceModel [Package]	UML Package	source: Data is stored in element's ToDo property.	
Responsibility [Class]	Competence [Class]	persons: Data is stored in element's ToDo property. roles: Data is stored in element's ToDo property.	
Role [Class]	PostRole [Property]	*persons*: Data is stored in element's ToDo property. *responsibilities*: *RequiresCompetence* relationships are created between *PostRole* and *Competences*.	
Service [Package]	Resource [Class]	*status*: Data is stored in element's ToDo property.	
ServiceArea [Package]	UML Package	*version/date*: Data is stored in element's ToDo property.	
ServiceSpecification [Class]	ServiceInterface[Interface]	*serviceDescription*: Comment is added.	
SoaService [Port]	ServicePoint [Port]		
Standard [Class]	Standard [Class]		
Standards/ Performance Subject [Element]	UPDMElement [Element]	*measurements* to *actualMeasurements.* *performanceParameterSet* to *measurementTypes.*	
StandardsForecastProfile [Package]	View (SysML) with conforms to TV-2	*basedOn*: Data is stored in element's ToDo property.	
StandardsProfile [Package]	View (SysML) with conforms to TV-1	*applicableDate*: Data is stored in element's ToDo property. *basedOn*: Data is stored in element's ToDo property.	
System [Class]	System [Class]	*participant*: Data is stored in element's ToDo property. *performedFunctions*: Performs relationship is created between function and System.	
SystemDataElement [Class]	DataElement[Class]		

DoDAF 1.5	UPDM 1.x	Properties	Comments
SystemDataExchange [InformationFlow]	ResourceInteraction[InformationFlow]	consumedBy: Data is stored in element's ToDo property. producedBy: Data is stored in element's ToDo property.	
SystemDataRepository [DataStoreNode]	Does not map		
SystemFunction [Activity]	SystemFunction [Activity]	*allocatedAt*: Allociation relationship to Systems Node should be created. *consumes*: Data is stored in element's ToDo property. *parent*: Aggregation relationship is created between System Function and parent System Function. *performedBy*: Performs relationship is created between System Function and System. *produces*. Data is stored in element's ToDo property. *subfunction*: Aggregation relationship is created between System Function and child System Function.	ImplementsOperational relationship is created between Systems Function and its implemented elements if any.
SystemFunctionAction [CallBehaviorAction]	SystemFunctionAction [CallBehaviorAction]		
SystemRule [Constraint]	ResourceConstraint [Constraint]	*type* to *kind*	
SystemsNode [Class]	SystemsNode [Class]	*allocatedFunctions*: Allociation relationship from System Function should be created.	ImplementsOperational relationship is created between Systems Node and its implemented elements if any.
SystemsNodeUsage [Property]	UsedConfiguration[Property]		
SystemUsage [Property]	Subsystem[Property]	*participant*: Data is stored in element's ToDo property.	
TargetArea [Class]	Location [DataType]		
Technology [Class]	Artifact[Class]		

DoDAF 1.5	UPDM 1.x	Properties	Comments
TechnologyForecastProfile [Package]	UML Package	*basedOn*: Data is stored in element's ToDo property. *timePeriod*: Data is stored in element's ToDo property.	
TimePeriod [TimeConstraint]	ISO8601DateTime [LiteralString]		
TimePeriodList [Class]	Timeline [Package]	*timePeriods* to *ownedElements.*	
WAN [Class]	System[Class]		

TABLE 3. **Relationship Mapping**

DoDAF 1.5	UPDM 1.x	Properties	Comments
Backup [Usage]	ResourceInteraction [InformationFlow]		Name is set to "Backup"
CommunicationsLink [Association, Connector]	SystemConnector[Association, Connector]	communicationsPath: Data is stored in element's ToDo property.	
Contributing [Usage]	ResourceInteraction [InformationFlow]		Name is set to "Contributing"
Coordination [Usage]	ResourceInteraction [InformationFlow]		Name is set to "Coordination"
Direct [Usage]	Commands [InformationFlow]		
Indirect [Usage]	ResourceInteraction [InformationFlow]		Name is set to "Indirect"
InformationFlow [ControlFlow, ObjectFlow]	OperationalActivityEdge [ActivityEdge]		
Interface [Association, Connector	SystemConnector[Association, Connector]		ImplementsOperational relationship is created between System Connector and its implemented elements if any.
Line [Association]	ArbitraryRelationshipConnector [Connector]	representationType: Data is stored in element's ToDo property	
Needline [Association, Connector]	Needline [Association, Connector]		ImplementsOperational relationship is created between Information Elemented and its implementations if any.
OrganizationalRelationship [Usage]	ResourceInteraction [InformationFlow]		Name is set to "OrganizationalRelationship".
SituationDependent [Usage]	ResourceInteraction [InformationFlow]		Name is set to "SituationDependent".

DoDAF 1.5	UPDM 1.x	Properties	Comments
TimedStandardsForecast [Usage]	Forecast [Dependency]	*discussion*: Data is stored in element's ToDo property. *requires*: Data is stored in element's ToDo property. *standardStatus*: Data is stored in element's ToDo property. *timePeriod* to *startDate*.	
TimedTechnologyForecast [Usage	Forecast [Dependency]	*discussion*: Data is stored in element's ToDo property. *requiredBy*: Data is stored in element's ToDo property. *retiredStandard*: Data is stored in element's ToDo property. *timePeriod* to *startDate*.	

Entity and relationship mapping raises further issues that requires specific solutions, all of which are described in sub-section 3.3.

3.3 Data Mapping Problems and Solutions

If you went through the previous sections of the manual, you probably noticed that a large number of element properties were either marked "Data is stored in element's ToDo property" or replaced by relationships in UPDM.

"Data is stored in element's ToDo property" indicates that element properties which can not be mapped will be placed in element's **ToDo** property. For example, if you have *Capability* in a DoDAF project that has *mission* property defined, the *migration mission* value will be placed in the ToDo property of *Capability* in the following text form: "**DoDAF mission = UseCase [Element Name]**"..

Value of the element property that does not map is placed in owner's ToDo property.

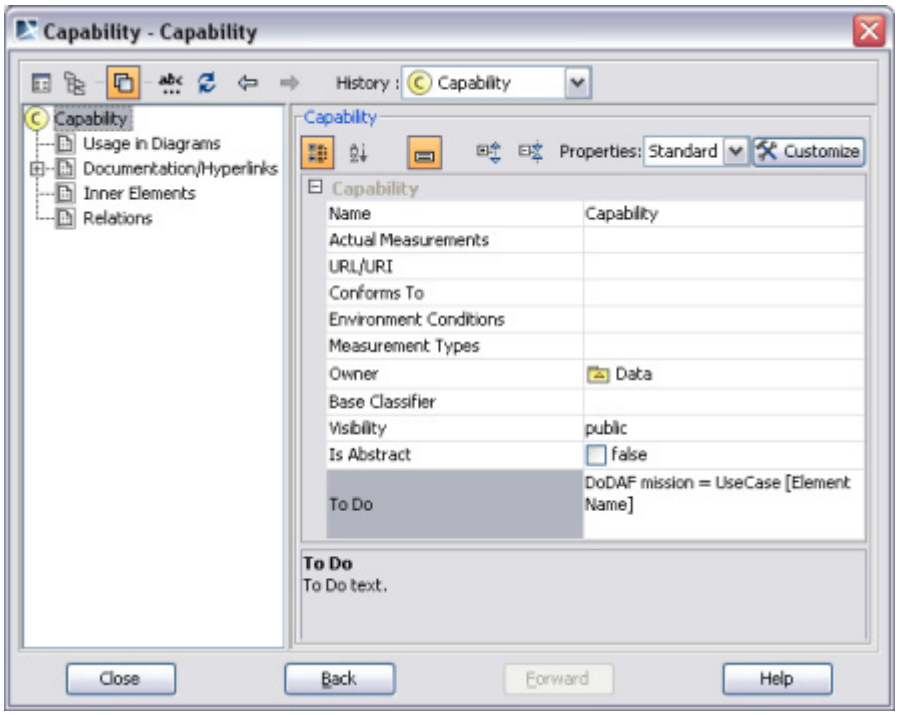

Figure 1 -- ToDo property in migrated element's specification

Use the **Find ToDo** feature to search for elements containing **ToDo** information. From the **Edit** main menu, choose **Find ToDo**.

The following migration issues require properties to map relationships:

1. *Implements* and *implementedBy* properties of DoDAF elements are mapped to *implementsOperational* relationship. Relationship source is *implements* property owner and target is *implementedBy* property owner. As you will notice, these properties also exist in UPDM. However, they are derived ones and as such can not be edited, which means they are calculated from *implementsOperational* relationships.
2. *Performs* and *performedBy* properties of DoDAF elements are mapped to *performs* relationship. Relationship source is a *performs* property owner and target is a *performedBy* property owner.

Problems can also arise when UML or SysML relationships are used instead of UPDM ones. All these are explained in Table 2 on page 8.

4. Migrating Server Projects

When you are working with server projects, part of the elements might be locked by other users. You cannot migrate locked elements. The following procedure explains how to migrate a server project from DoDAF to UPDM 1.x.

To migrate a server project

1. In the Model Browser, open the **Locked Elements** tab to check if there are no elements locked by other users.

NOTE	If there are elements locked by other users, each user must unlock elements locked by him/her. Migration will be aborted if at least one element cannot be locked.

2. Perform the project migration. See the chapter "3. DoDAF Projects Migration" on page 4.
3. Commit the project.

5. Customizing Migration

You can customize DoDAF migration to UPDM. However, to do so, you need to have some basic knowledge of UML, especially stereotypes. For this reason, we highly recommend that you use predefined migration configurations.

Customizing DoDAF migration to UPDM requires a basic understanding of how to perform migration. Basic rules and principles of data migration to UPDM can be found in <install.root>\profiles\DoDAFtoUPDMMapping.mdzip project. This project consists of DoDAF, UPMD profiles and predefined relationships defining mapping between DoDAF and UPDM stereotypes.

DoDAF to UPDM mapping rules can be found in **<install.root>\profiles\DoDAFtoUPDMMapping.mdzip** project.

Data Migration is based on three types of stereotyped relationships:

1. Replace Tagged Value,
2. Replace Stereotype,
3. Replace Type.

These relationships are the basics of migration.

1. Replace tagged value relationship is drawn between tag definitions of two different stereotypes. Relationship direction shows the tag definition that is to be replaced during migration.
2. Replace stereotype relationship is drawn between two different stereotypes. Relationship direction shows the stereotype that is to be replaced during migration.
3. Replace type relationship is drawn between concrete elements. It replace the type of element dependant on the direction of relationship.

Figure 2 on page 17 below shows data mapping rules for Performance Parameter Type element.

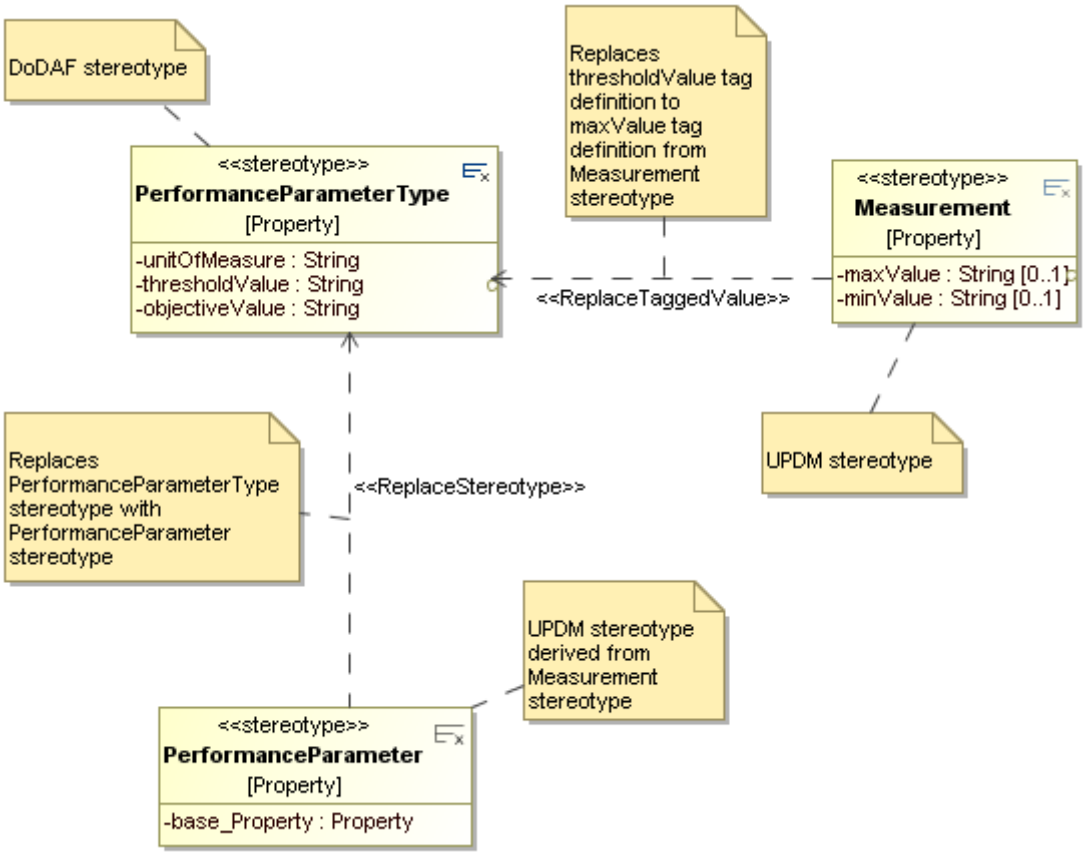

Figure 2 -- Migration basics

As shown in Figure 2 on page 17, *PerformanceParameterType [Property]* stereotype from DoDAF profile is marked to be replaced with *PerformanceParameter [Property]* from UPDM profile. Also *threshold* tag definition owned by *PerformanceParameterType* should be changed to *maxValue* tag definition owned by *Measurement* element from UPDM profile.

As shown in Figure 3 on page 18, another exceptional case is property value mapping to the **ToDo** property of the element. *TODO_Owner [Element]* stereotype from UML Standard profile is marked to replace

unitOfMeasure tag definition owned by *PerformanceParameterType*. This case results to "*DoDAF unitOfMeasure = [value]*" in *PerformanceParameter* element **ToDo** property.

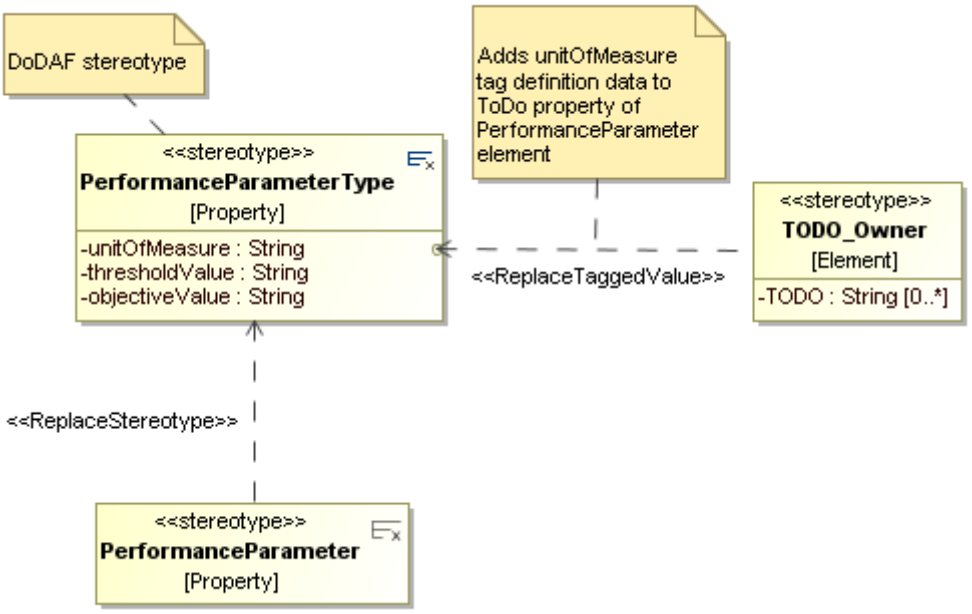

Figure 3 -- Storing derived tag data to ToDo property

For each DoDAF stereotype migration there are Diagrams created in migration profile. We suggest that you study these diagrams carefully before customizing migration to meet your specific needs.

For each DoDAF stereotype there are Diagrams created in migration profile.