



# Model-Driven Software Engineering (MDSE)

## UML Profiles

**Bahman Zamani, Ph.D.**  
[bahmanzamani.com](http://bahmanzamani.com)

Department of Software Engineering

Faculty of Computer Engineering

University of Isfahan



## Why UML Profile? [zam 09]

- Different projects (and organizations) have different needs and use their own domain concepts
- Therefore, it is needed to customize UML for specific domains.
- Fortunately, from the first day, UML was designed to be extendable and customizable.
- New modeling extensions can be introduced into UML by defining a UML Profile.



## What is a UML Profile? [zam 09]

- UML profiles are extension mechanisms that allow you to tailor UML for specific areas such as Telecommunication.
- Profiles are sometimes referred to as the ‘lightweight’ built-in extension mechanisms of UML.
- A mechanism provided by UML that allows us to extend it for specific purposes, in other words
- To define a Domain-Specific Modeling Language (DSML)

# Defining a UML Profile [zam 09]

- Defining a UML Profile includes
  - Identifying a subset of UML metamodel
  - Specifying the well-formedness rules, preferably by OCL
  - Defining Stereotypes, Tagged Values, and Constraints



# A Systematic Approach for Defining UML Profiles [Sel07]

1. Defining the domain metamodel
  - domain model of the DSL that we are designing
    - This domain model is in fact the metamodel of our language.
2. Mapping the domain metamodel to UML metamodel
  - Map each concept in the domain model into one appropriate base class in UML metamodel.
  - define one stereotype for each concept

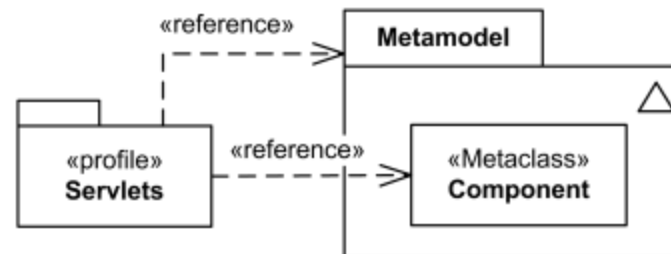
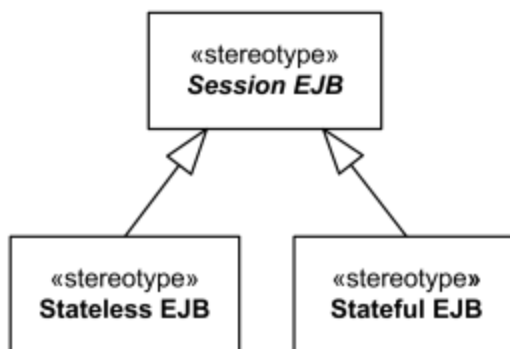
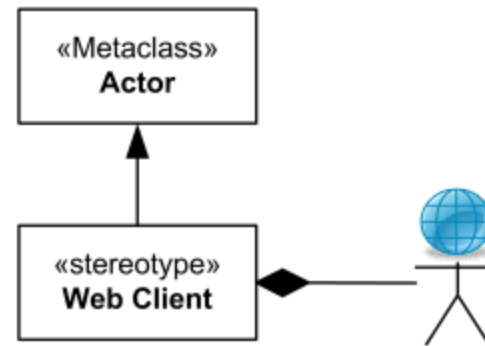
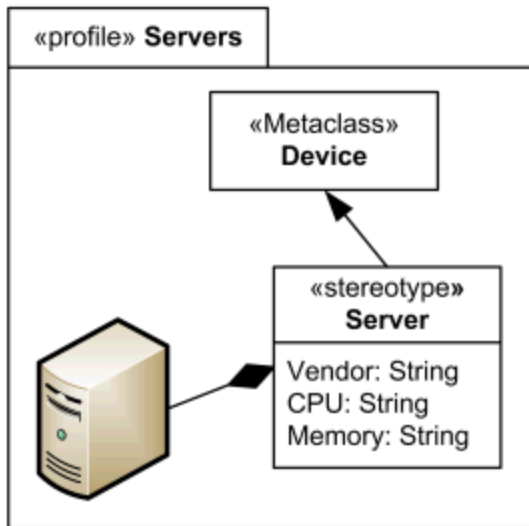
# UML Profile Diagrams

[<http://www.uml-diagrams.org/> ]

- **Profile diagram** is structure diagram which describes **lightweight extension mechanism** to the UML by defining custom stereotypes, tagged values, and constraints.
- Graphical nodes and edges drawn on profile diagrams are: profile, metaclass, stereotype, extension, reference, profile application.

# UML Profile Diagram Examples

[<http://www.uml-diagrams.org/>]



# Sample Profiles: SoaML

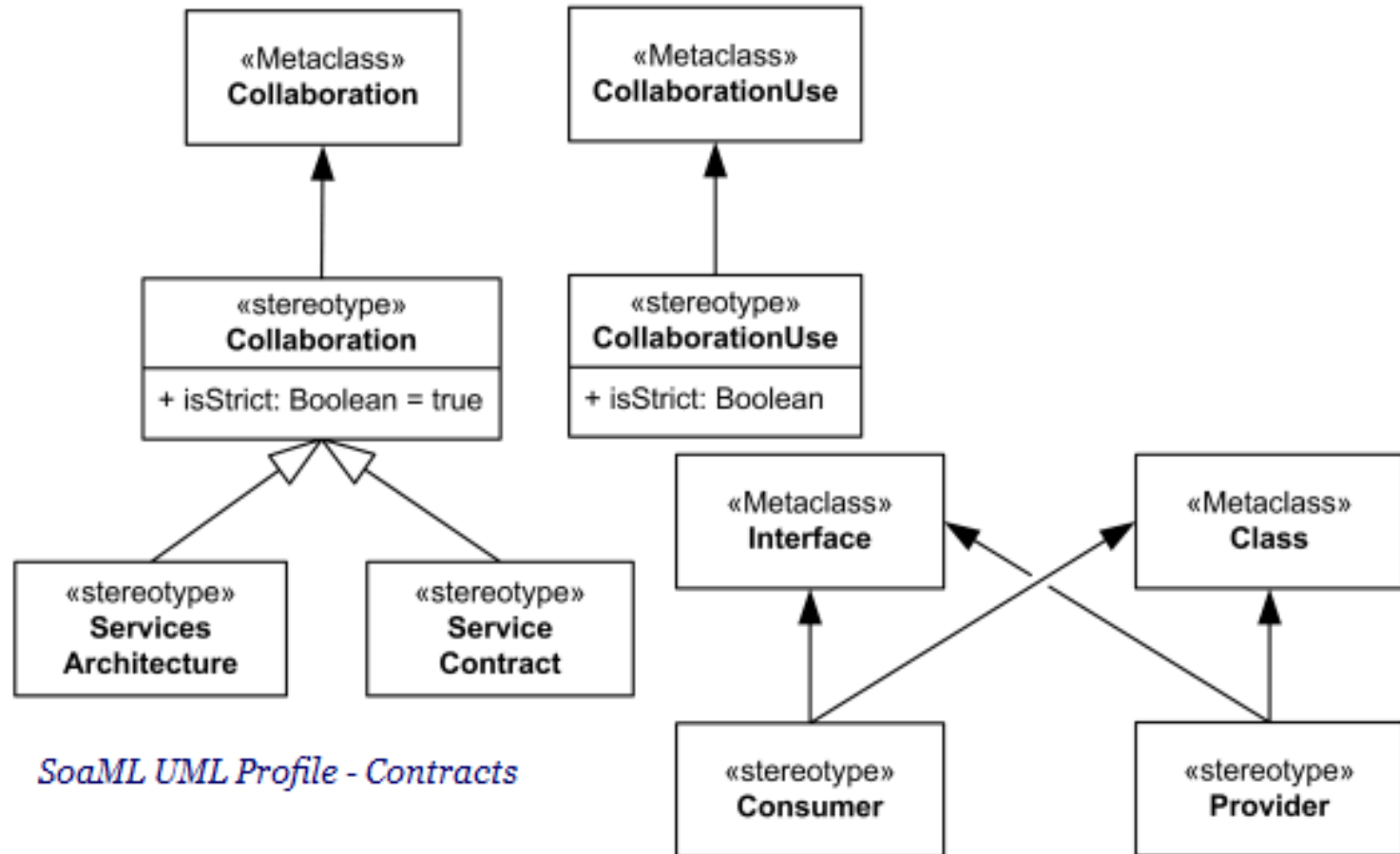
[<http://www.uml-diagrams.org/> ]

- UML Profile for Service Oriented Architecture described in **Service Oriented Architecture Modeling Language (SoaML)** specification.
- The SoaML UML profile supports modeling of service-oriented architectures, including specification of systems of services, specification of individual service interfaces, and specification of service implementations.



# SoaML UML Profile – Contracts

[<http://www.uml-diagrams.org/>]



*SoaML UML Profile - Contracts*

# Sample profiles from OMG

[<http://www.omg.org/spec/>]

## UML Profile Specifications

SPECIFICATION	acronym	topical area / domain	Document #
<u>UML Profile for Advanced and Integrated Telecommunication Services</u>	TelcoML	telecommunications	formal/2013-08-02
<u>SES Management TelcoML Extension</u>	TelcoML-SES	telecommunications	formal/2013-08-03
<u>UML Profile for BPMN Processes</u>	BPMNProfile	domain, modeling	dtc/2013-04-01
<u>UML Profile for CORBA</u>	CORP	middleware	formal/2002-04-01
<u>UML Profile for CCM</u>	CCMP	middleware	formal/2005-07-06
<u>UML Profile for CORBA and CCM</u>	CCCMP	middleware	formal/2008-04-07
<u>UML Profile for Enterprise Application Integration</u>	EAI	domain, modeling	formal/2004-03-26
<u>UML Profile for Enterprise Distributed Object Computing</u>	EDOC	domain, modeling	formal/2004-02-01
<u>UML Profile for MARTE: Modeling and Analysis of Real-time and Embedded Systems</u>	MARTE	real-time, middleware	formal/2011-06-02
<u>UML Profile for QoS and Fault Tolerance</u>	QFTP	real-time, middleware, modeling	formal/2008-04-05
<u>UML Profile for Schedulability, Performance and Time</u>	SPTP	real-time, middleware, modeling	formal/2005-01-02
<u>UML Profile for System on a Chip</u>	SoCP	real-time, middleware, modeling	formal/2006-08-01
<u>UML Profile for Software Radio (aka PIM &amp; PSM for Software Radio Components)</u>	SDRP	software-based communications	formal/2007-03-01
<u>UML Profile for Voice</u>	VOICP	telecommunications	formal/2008-04-06
<u>UML Testing Profile</u>	UTP	modeling	formal/2013-04-03

# UML Profile and Metamodel for Voice-based Applications (VOICP) [<http://www.omg.org/spec/VOICP/1.0/>]

- The Voice Metamodel

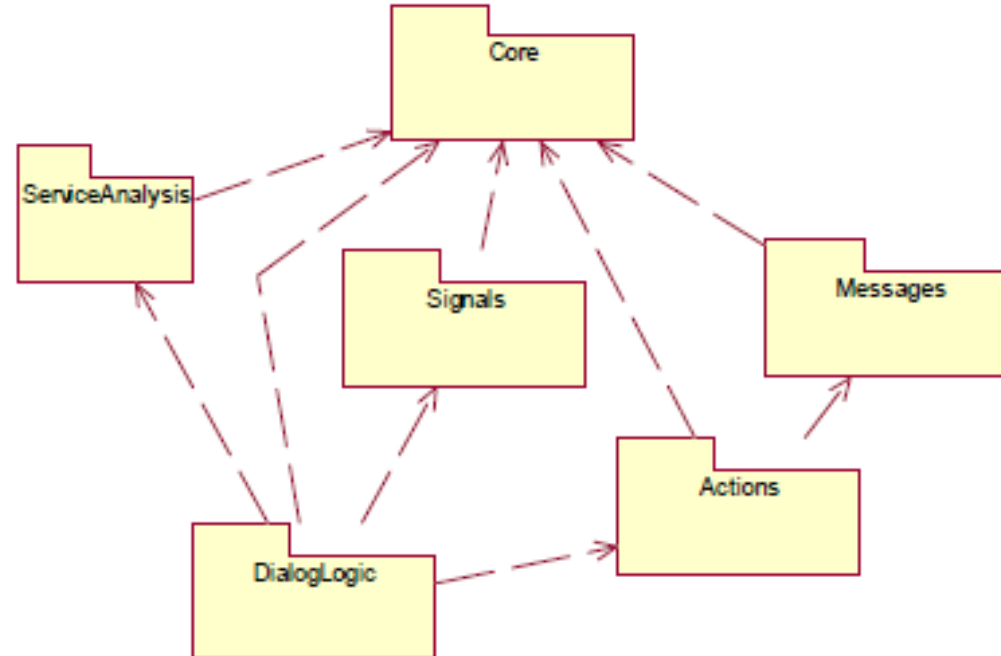

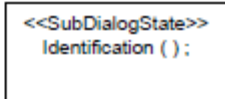

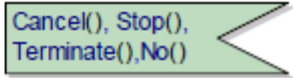


Figure 8.1 - Structure of the Voice metamodel

# Voice Metamodel to UML Correspondences

[<http://www.omg.org/spec/VOICP/1.0/> ]

Voice Metamodel Concept	UML 2.0 Concept	Notation
<b>VOICE DIALOGS</b>		
Dialog	State machine stereotyped <<Dialog>>	One or more state-transition diagrams
WaitState	State stereotyped <<WaitState>>	
SubDialog-State	Action stereotyped <<SubDialogState>>	
Transition	Transition.	Transition arrow. The trigger and the actions of the "whole" transition are explicitly drawn as nodes linked by transitions.
Trigger	Trigger	<p>A unique trigger symbol</p>  <p>Multiple triggers</p> 

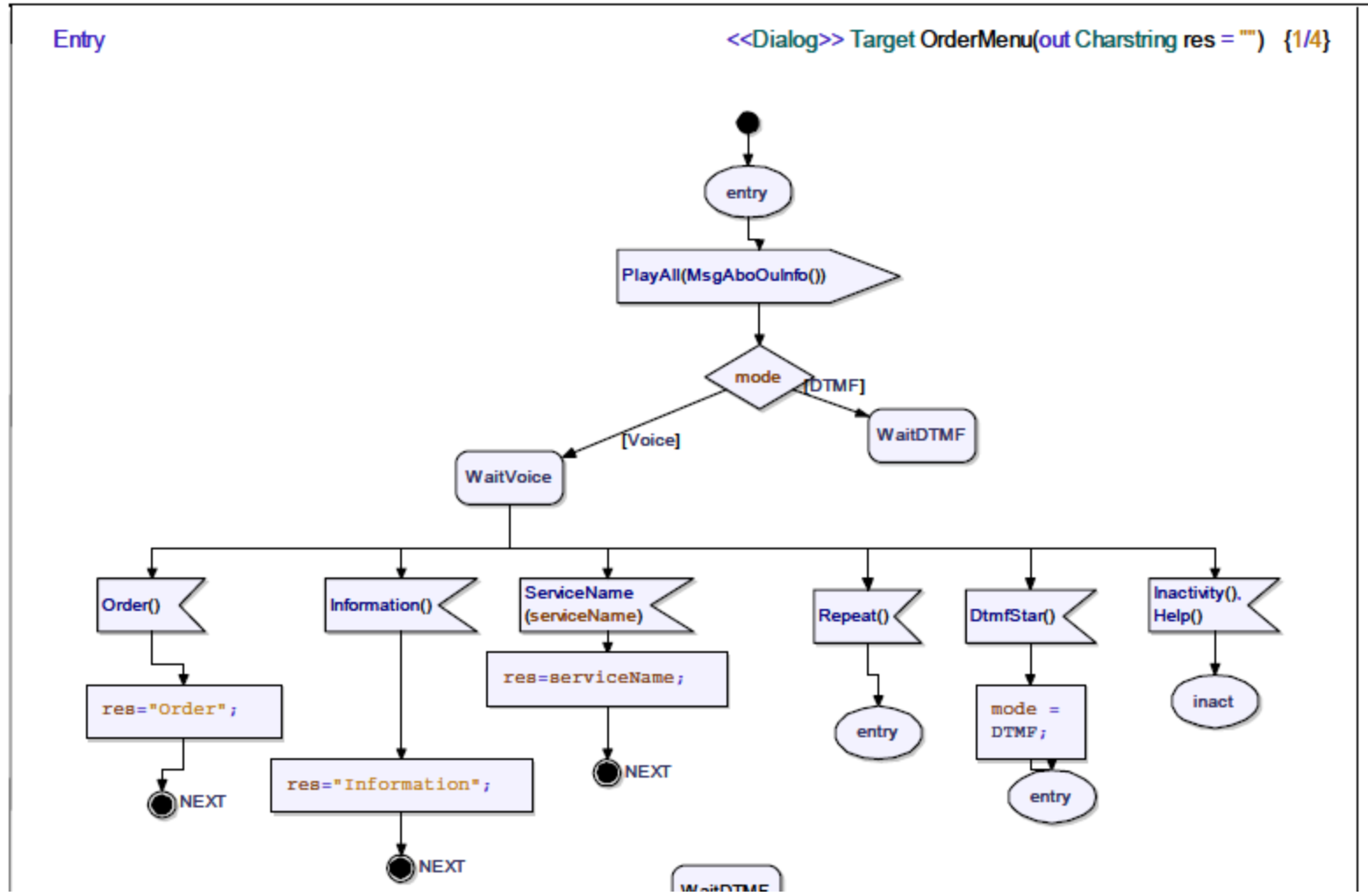
# Stereotypes of the UML Voice Profile

[<http://www.omg.org/spec/VOICP/1.0/> ]

Stereotype	UML 2.0 Base class	Voice MM concept	Tagged Values
<<Dialog>>	StateMachine	Dialog	
<< WaitState >>	State	WaitState	
<<SubDialogState>>	Action	SubDialogState	
<<Diversion>>	FinalState	DiversionNode	
<<Accept>>	Trigger	Trigger	
<<Concept>>	Signal	Concept	
<<DTMF>>	Signal	DTMF	
<<ExternalEvent>>	Signal	ExternalEvent	
<<MessageContainer>>	Class	Ownership of Message	
<<Inputcontainer>>	Class	Ownership of InputEvent	
<<Message>>	Operation	Message	
<<Silence>>	Operation	Silence	
<<FixPart>>	Operation	FixPart	format : String
<<VariablePart>>	Operation	Variable	format : String
<<Service>>	Package	Service Root of the definitions for a given VoiceService	

# Example: A Menu Dialog

[<http://www.omg.org/spec/VOICP/1.0/>] ]





# Tool Support for UML Profile

- Rational Software Architect
- Enterprise Architect
- Altova Umodel
- ArgoUML
- Visual Paradigm for UML
- Papyrus



# References

- [Sel07] Bran Selic. A systematic approach to domain-specific language design using UML., In *Proc. ISORC'07*, pages 2–9, 2007.
- [Zam 09] Bahman Zamani. *On Verifying the Use of a Pattern Language in Model Driven Design*. PhD thesis, Concordia University, Canada, 2009.