



D16.4v0.1. WSML F-Logic Syntax

WSMO Working Draft 22 March 2004

This version:

<http://www.wsmo.org/2004/d16/d16.4/v01/20040322/>

Latest version:

<http://www.wsmo.org/2004/d16/d16.4/v01/>

Editors:

Holger Lausen
Dumitru Roman

Authors:

Holger Lausen
Dumitru Roman

This document is also available in non-normative [PDF](#) version.

Copyright © 2004 [DERI](#)®, All Rights Reserved. [DERI](#) liability, trademark, document use, and software licensing rules apply.

Abstract

This document defines F-Logic Signatures for the Web Service Modelling Ontology.

Table of contents

- [1. Introduction](#)
- [2. F-Logic Syntax](#)
- [3. References](#)
- [Acknowledgements](#)

1. Introduction

This document provides F-Logic Syntax for the Modelling primitives outlined in the Web Service Modeling Language WSML. It is based on the BNF syntax provided by [\[Oren, 2004\]](#).

However since different formalisms serve different purposes and have different level of expressivity, there are some differences:

- The range value of the non functional properties is string and not identifier.
- Axiom definitions are not described (since we can't encode f-logic in itself).
- Within [\[Oren, 2004\]](#) more information about cardinality restrictions are given, i.e. in F-Logic signatures we can't express the difference between "zero or one" and "exactly one" and between "zero or more" and "one or more".
- The BNF syntax imposes an order on the syntactic constructs (i.e. the order of attribute definitions) which F-Logic does not.

2. F-Logic Syntax

The F-Logic Syntax (see [Appendix](#)) describes WSML as outlined earlier.

4. References

[Oren, 2004] E. Oren (editor) (2004). *BNF Syntax*. WSMO Working Draft v0.1, Digital Enterprise Research Institute (DERI), available from <http://www.wsmo.org/2004/d16/d16.1/>.

Appendix A. F-Logic Syntax

```
nonFunctionalProperty
[
  type=>string,
  rights=>string,
  subject=>string,
  format=>string,
  coverage=>string,
  creator=>string,
  relation=>string,
  contributor=>string,
  publisher=>string,
  source=>string,
  date=>string,
  language=>string,
  version=>string,
  title=>string,
  description=>string
].
```

nonFunctionalPropertyWS::nonFunctionalProperty

```
[
  identifier=>string,
  financial=>string,
  security=>string,
  networkRelatedQoS=>string,
  transactional=>string,
  performance=>string,
  scalability=>string,
  reliability=>string,
  robustness=>string,
  accuracy=>string,
  trust=>string
].
```

ontology

```
[
  identifier=>string,
  nonFunctionalProperties=>nonFunctionalProperty,
  usedMediators=>>ooMediator,
  conceptDefinitions=>>conceptDefinition,
  instances=>>instance,
  axioms=>>axiomDefinition,
  relationDefinitions=>>relationDefinition
].
```

webService

```
[
  identifier=>string,
  interfaces=>>interface,
  capability=>capability,
  nonFunctionalProperties=>nonFunctionalPropertyWS,
  usedMediators=>>ooMediator
].
```

goal

```
[
  identifier=>string,
  postconditions=>>axiomDefinition,
  nonFunctionalProperties=>nonFunctionalProperty,
  effects=>>axiomDefinition,
  usedMediators=>>oo_or_ggMediator
].
```

mediator

```
[
  identifier=>string,
  nonFunctionalProperties => nonFunctionalPropertiesWS,
  sourceComponent =>> <,
  targetComponent =>> goal_or_mediator_or_ontology_or_webService,
  mediationService => goal_or_wwMediator
].
```

ooMediator::mediator

```
[
```

```

    sourceComponent =>> ontology_or_ooMediator
  ].

ggMediator::mediator
[
  sourceComponent =>> goal_or_ggMediator,
  targetComponent =>> goal_or_ggMediator,
  usedMediators =>> ooMediator,
  reduction => axiomDefinition
].

wgMediator::mediator
[
  sourceComponent =>> webService_or_wgMediator,
  targetComponent =>> goal_or_wgMediator,
  usedMediators =>> ooMediator,
  reduction => axiomDefinition
].

wwMediator::mediator
[
  sourceComponent =>> webService_or_wwMediator,
  targetComponent =>> webService_or_wwMediator,
  usedMediators =>> ooMediator
].

axiomDefinition
[
  identifier=>string,
  definedBy=>logicalExpression,
  nonFunctionalProperties=>nonFunctionalProperty
].

instance::axiomDefintion
[
  instanceOf=>>conceptDefinition,
  attributeValues=>>attributeValueDefinition
].

instance::axiomDefintion
[
  instanceOf=>>conceptDefinition,
  attributeValues=>>attributeValueDefinition
].

attributeValueDefinition::axiomDefinition
[
  value => axiomDefinition
].

relationDefinition :: axiomDefinition
[
  parameters =>> parameterDefinition
].

parameterDefinition :: axiomDefinition

```

```

[
  domain => axiomDefinition
].

conceptDefinition :: axiomDefinition
[
  superConcepts =>> conceptDefinition,
  attributes =>> attributeDefinition,
  methods =>> methodDefintion
].

methodDefintion :: axiomDefinition
[
  range => axiomDefinition,
  parameters =>> parameterDefinition
].

capability
[
  identifier=>string,
  nonFunctionalProperties => nonFunctionalProperties,
  usedMediators =>> ooMediator_or_wgMediator,
  preconditions =>> axiomDefinition,
  postconditions =>> axiomDefinition,
  assumptions =>> axiomDefinition,
  effects =>> axiomDefinition
].

interface
[
  identifier=>string,
  nonFunctionalProperties => nonFunctionalProperties,
  usedMediators =>> ooMediator,
  choreography => instantiatedCMEP,
  orchestration => instantiatedPSP
].

% goal_or_mediator_or_ontology_or_webService
goal::goal_or_mediator_or_ontology_or_webService.
mediator::goal_or_mediator_or_ontology_or_webService.
ontology::goal_or_mediator_or_ontology_or_webService.
webService::goal_or_mediator_or_ontology_or_webService.
goal_or_mediator_or_ontology_or_webService.

%ontology_or_ooMediator
ontology::ontology_or_ooMediator.
ooMediator::ontology_or_ooMediator.
ontology_or_ooMediator.

%goal_or_wwMediator
goal::goal_or_wwMediator.
wwMediator::goal_or_wwMediator.
goal_or_wwMediator.

% ooMediator_or_wgMediator

```

ooMediator::oo_or_ggMediator.
ggMediator::oo_or_ggMediator.
oo_or_ggMediator::mediator.

%goal_or_ggMediator
goal::goal_or_ggMediator.
ggMediator::goal_or_ggMediator.
goal_or_ggMediator.

%webService_or_wwMediator
webService::webService_or_wwMediator.
wwMediator::webService_or_wwMediator.
webService_or_wwMediator.

%webService_or_wgMediator
webService::webService_or_wgMediator.
wgMediator::webService_or_wgMediator.
webService_or_wgMediator.

%goal_or_wgMediator
goal::goal_or_wgMediator.
wgMediator::goal_or_wgMediator.
goal_or_wgMediator.

Acknowledgements

The work is funded by the European Commission under the projects DIP, Knowledge Web, Ontoweb, SEKT, SWWS, Esperanto, COG and h-TechSight; by Science Foundation Ireland under the DERI-Lion project; and by the Vienna city government under the CoOperate programme.

The editors would like to thank to all the [members of the WSMO working group](#) for their advises and inputs to this document.

[webmaster](#)