Semantic Web Services-based Reasoning in the Design of Software Product Lines

J. Jeffrey Rusk and Dragan Gasevic

Athabasca University

Canada

Research Goal

To evaluate the suitability of the Web Service Modeling Ontology (WSMO) in the encoding of product configurations and related constraints from a software product line (SPL) in such a manner as to better enable reasoning approaches which facilitate higher automation of service discovery, composition, invocation, and monitoring in service oriented architectures (SOA).

Outline

- Background and Motivation
- Feature Models (FM)
- Web Service Modeling Ontology (WSMO)
- Model Transformations
  - FM to WSMO
  - Product Configuration to WSMO
- Orchestration in WSMO
- Reasoning
- Implementation, Conclusion and Future Work

Background Issues

- Impediments to successful implementation of SPL when considering SOA
- Challenges representing SOA as SPL
- Limits to the expressiveness of FM
- Limited reasoning capabilities
- Ontology-related technology exists to support

Deliverables

- Mappings between FM and WSMO
- Transformation implementation
- Reasoning framework

What do the deliverables make possible?

The ability to explore and evaluate:

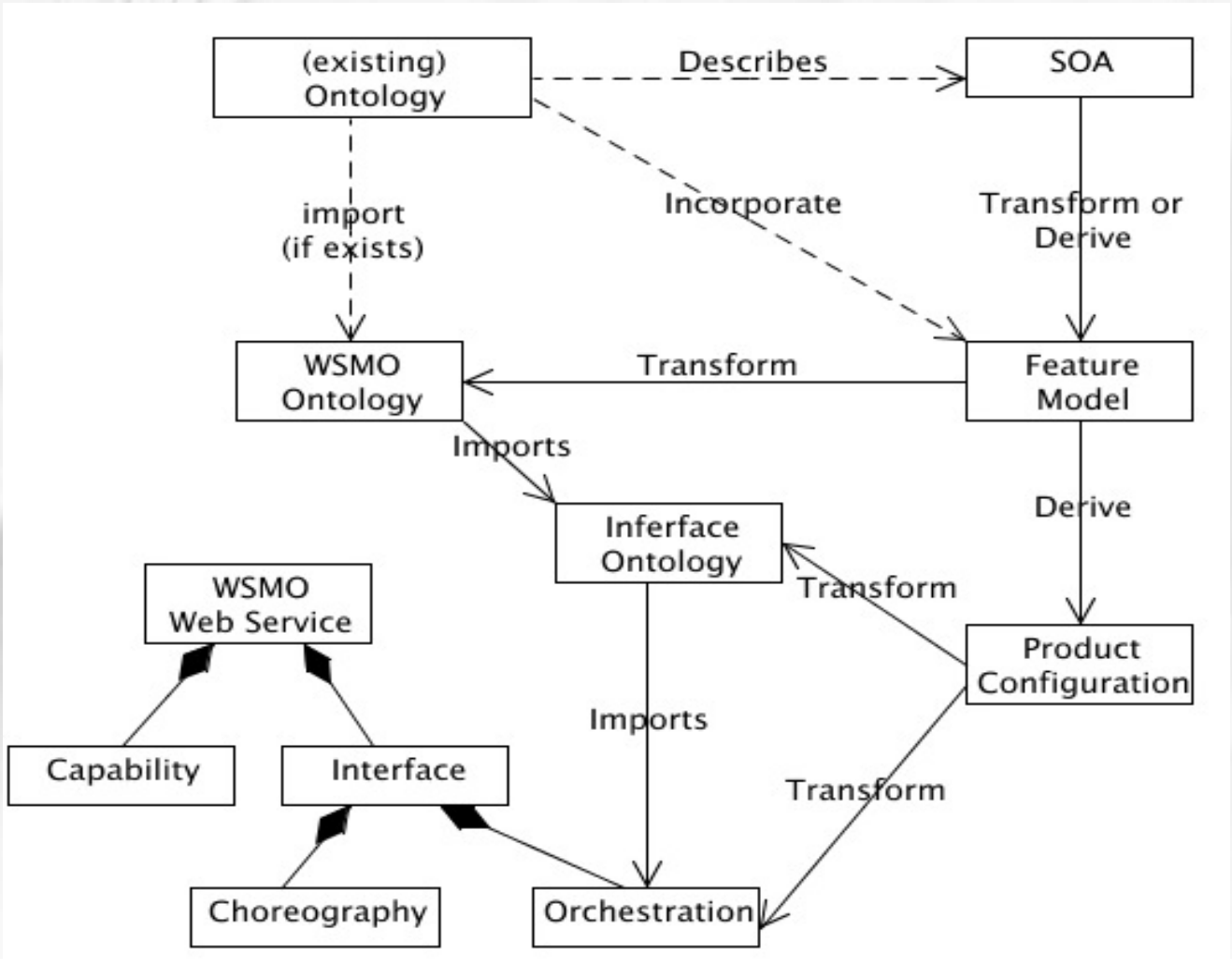
- accuracy of the mapping possible between the two formalisms.
- level of automation supported during transformation
- support or guidance that the ontology can provide to feature modeling.

Themes of this Workshop

- Variability and variability mechanisms
- Product composition

How does this work relate to these themes?

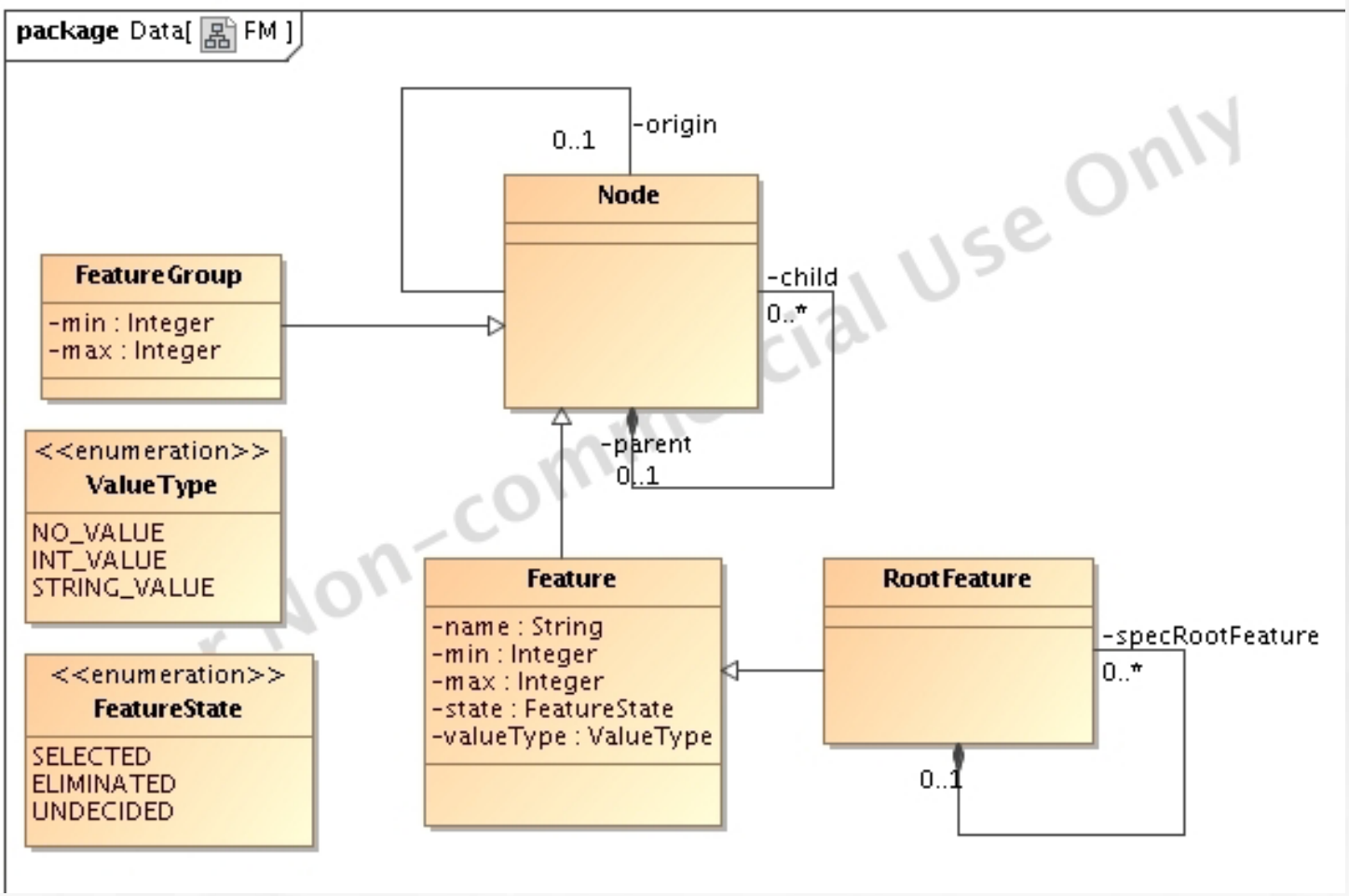
Overall Flow of Information



Feature Models

- SPL implementations typically feature-based
- FM ideal representation for SOA
- Using Czarnecki et al. notation and rendering
- Metamodel of FM and product configurations
- Tool support

Feature Model Metamodel

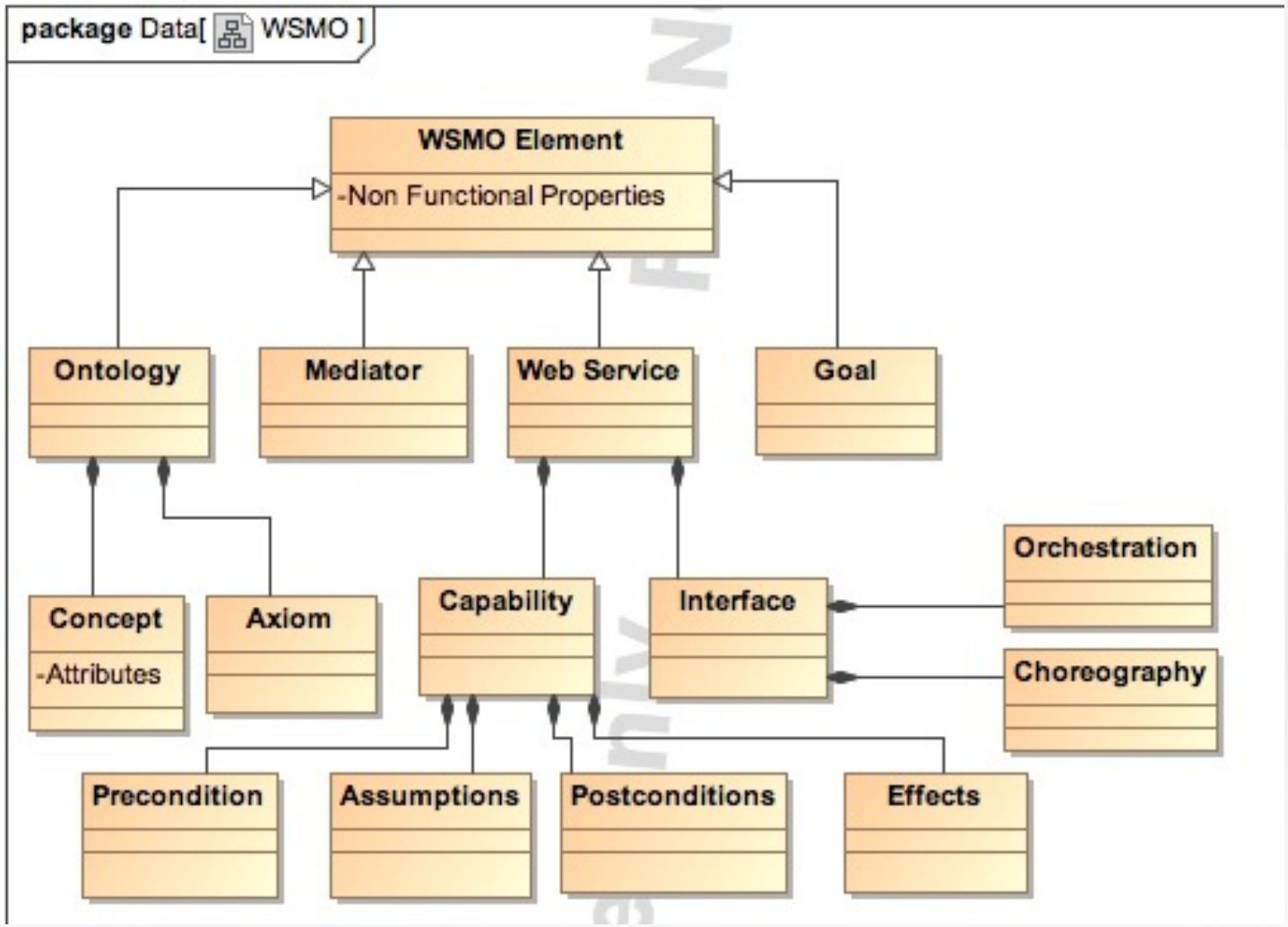


Adapted from: C.H.P. Kim, K. Czarnecki. Synchronizing cardinality-based feature models and their specializations. In *Model Driven Architecture – Foundations and Applications. 331-348. 2005.*

Web Service Modeling Ontology (WSMO)

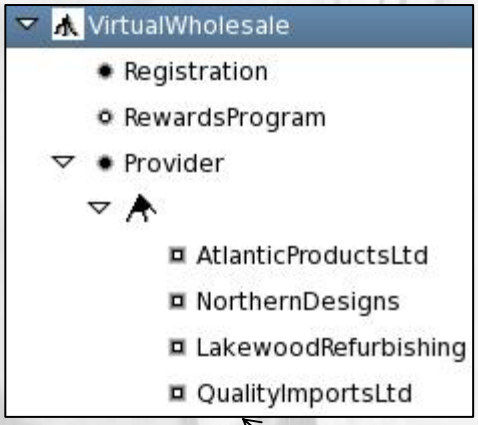
- Semantic describes all aspects of SWS
- Relatively new framework
- Tool support
- Four core elements
  - Ontologies
  - Web Services
  - Goals
  - Mediators

WSMO Metamodel



Model Transformation

Feature Model



WSMO

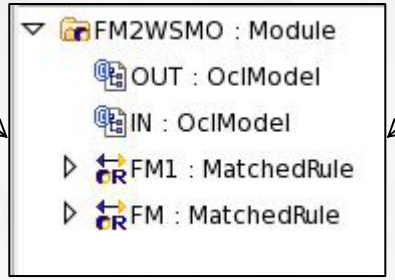
```
wsmlVariant _"http://www.wsmo.org/wsml/wsml-syntax/wsml-flight"
namespace { _"http://www.jeffrusk.org#"
,
     wsmostudio _"http://www.wsmostudio.org#" }

ontology VirtualWholesale
     nonFunctionalProperties
          wsmostudio#version hasValue "0.7.2"
     endNonFunctionalProperties

concept VirtualWholesale
     Registration impliesType  (1 1) Registration
     RewardsProgram impliesType  (0 1) RewardsProgram
     Payment impliesType  (1 1) Payment
     Provider impliesType  (1 4) Provider
     Shipping impliesType  (1 4) Shipping

concept Registration
```

ATL



XML Formats

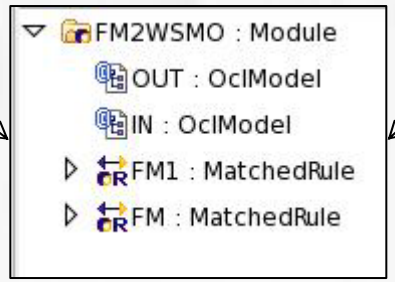
Feature Plugin XML Export

```
<feature min="1" max="1" name="VirtualWholesale" type=
  <feature min="1" max="1" name="Registration" type="I
  </feature>
  <feature min="0" max="1" name="RewardsProgram" type=
  </feature>
  <feature min="1" max="1" name="Provider" type="NONE"
    <featureGroup min="1" max="4" id="group">
      <feature min="0" max="1" name="AtlanticProductsL
      </feature>
      <feature min="0" max="1" name="NorthernDesigns"
      </feature>
      <feature min="0" max="1" name="LakewoodRefurbish
      </feature>
      <feature min="0" max="1" name="QualityImportsLtd
      </feature>
    </featureGroup>
  </feature>
```

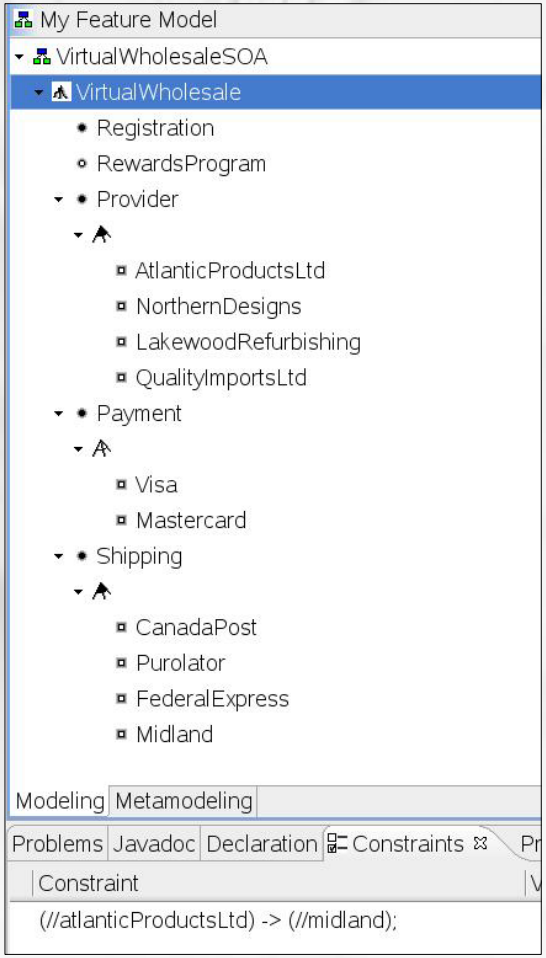
WSML-in-XML

```
<?xml version="1.0" encoding="UTF-8"?>
<wsml>
<ontology name="http://www.example.org#Virtual
<nonFunctionalProperties>
<attributeValue name="http://www.wsmostudio.or
<value type="http://www.wsmo.org/wsml/wsml-syn
</attributeValue>
</nonFunctionalProperties>
<concept name="http://www.example.org#Virtual
<attribute name="http://www.example.org#Regist
<range>http://www.example.org#Registration</ra
<minCardinality>1</minCardinality>
<maxCardinality>1</maxCardinality>
</attribute>
<attribute name="http://www.example.org#Reward
<range>http://www.example.org#RewardsProgram</
<maxCardinality>1</maxCardinality>
```

ATL



Feature Model to WSMO




```
wsmlVariant _"http://www.wsmo.org/wsml/wsml-syntax/wsml-flight"

ontology VirtualWholesale

concept VirtualWholesale
    Registration ofType  (1 1) Registration
    RewardsProgram ofType  (0 1) RewardsProgram
    Payment ofType  (1 1) Payment
    Provider ofType  (1 4) Provider
    Shipping ofType  (1 4) Shipping

concept Registration
concept Payment
concept Provider
concept Shipping
concept RewardsProgram
concept Visa subConceptOf Payment
concept Mastercard subConceptOf Payment
concept AtlanticProductsLtd subConceptOf Provider
concept NorthernDesigns subConceptOf Provider
concept LakewoodRefurbishing subConceptOf Provider
concept QualityImportsLtd subConceptOf Provider
concept CanadaPost subConceptOf Shipping
concept Purolator subConceptOf Shipping
concept FederalExpress subConceptOf Shipping
concept Midland subConceptOf Shipping

axiom DisjointPayment
    definedBy
        !- ?x memberOf Visa
  and ?x memberOf Mastercard.
```

Product Configuration to WSMO

- Most accurately represented as orchestration
- Overall executable business process that can be defined through interaction between Web services
- Choreography may be a factor as well

Orchestration in WSMO

- Unlike choreography, orchestration in WSMO is still under development
- Both based on abstract state machine
- Composed of state and set of guarded transitions
- State in form of ontology providing
  - Vocabulary for transition rules
  - Set of instances that change state.

Reasoning

- Effects of:
  - Product configuration choices
  - Adding, moving, deleting features
  - Assigning values to attributes
- Guidance for:
  - Constraints in ontology not present in FM
  - FM relationships not represented in ontology
  - Orchestration dependencies
  - Orchestrating services required

Implementation

- Feature Model Plugin
- WSMO Studio and KAON2 Reasoner
- ATL
- Eclipse

Conclusion

- Mappings between FM and WSMO
  - Accuracy of mappings
  - Level of automation attainable
  - Precision of feature discovery
  - Guidance provided by ontology
- Suitability of WSMO
  - Expressiveness
  - Related work

Future Work

- Refine mappings
- Improve transformation
- Explore further the available reasoning
- Integrate the various utilities into comprehensive plugin working in Eclipse environment

Thank you!
Questions?

Jeff Rusk
Director, Technical Services
Nunavut Impact Review Board
Cambridge Bay, Nunavut, Canada
(also MSc Candidate at Athabasca University)

jrusk@nirb.ca

Acknowledgements

The authors acknowledge the support of Athabasca University through the Graduate Student Research Fund.