Supporting Interoperability Using the Discrete-event Modeling Ontology (DeMO)

GREGORY SILVER
KUSHEL RIA BELLPADY
JOHN MILLER
KRYS KOCHUT
WILLIAM YORK

Ontologies

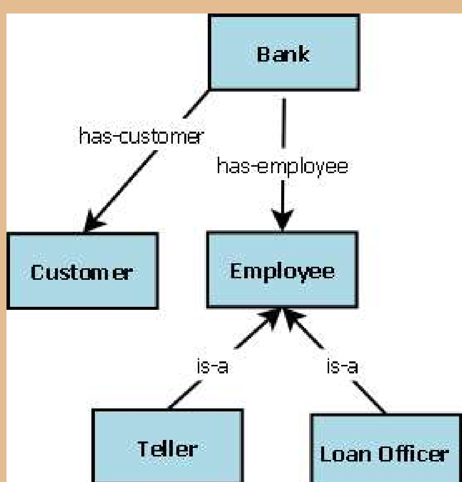
- **Ontology**: a description of concepts and relationships for a particular domain
  - Knowledge sharing and reuse
- **Ontological Commitment**: an agreement to use a vocabulary in a consistent way
  - Agents commit to ontologies in order to share knowledge
- **Domain Ontologies**: developed collaboratively
  - Biomedical
  - Geographical
  - Business
- **Web Ontology Language (OWL)**

Representations of Reality

Ontology

Represents reality by describing concepts and relationships

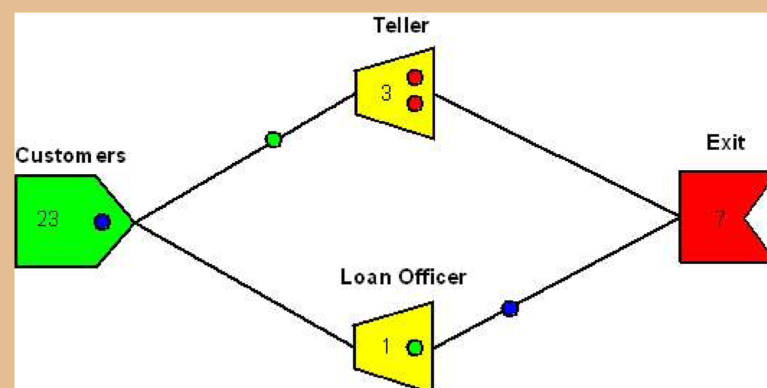
Focuses primarily on the structure of a domain



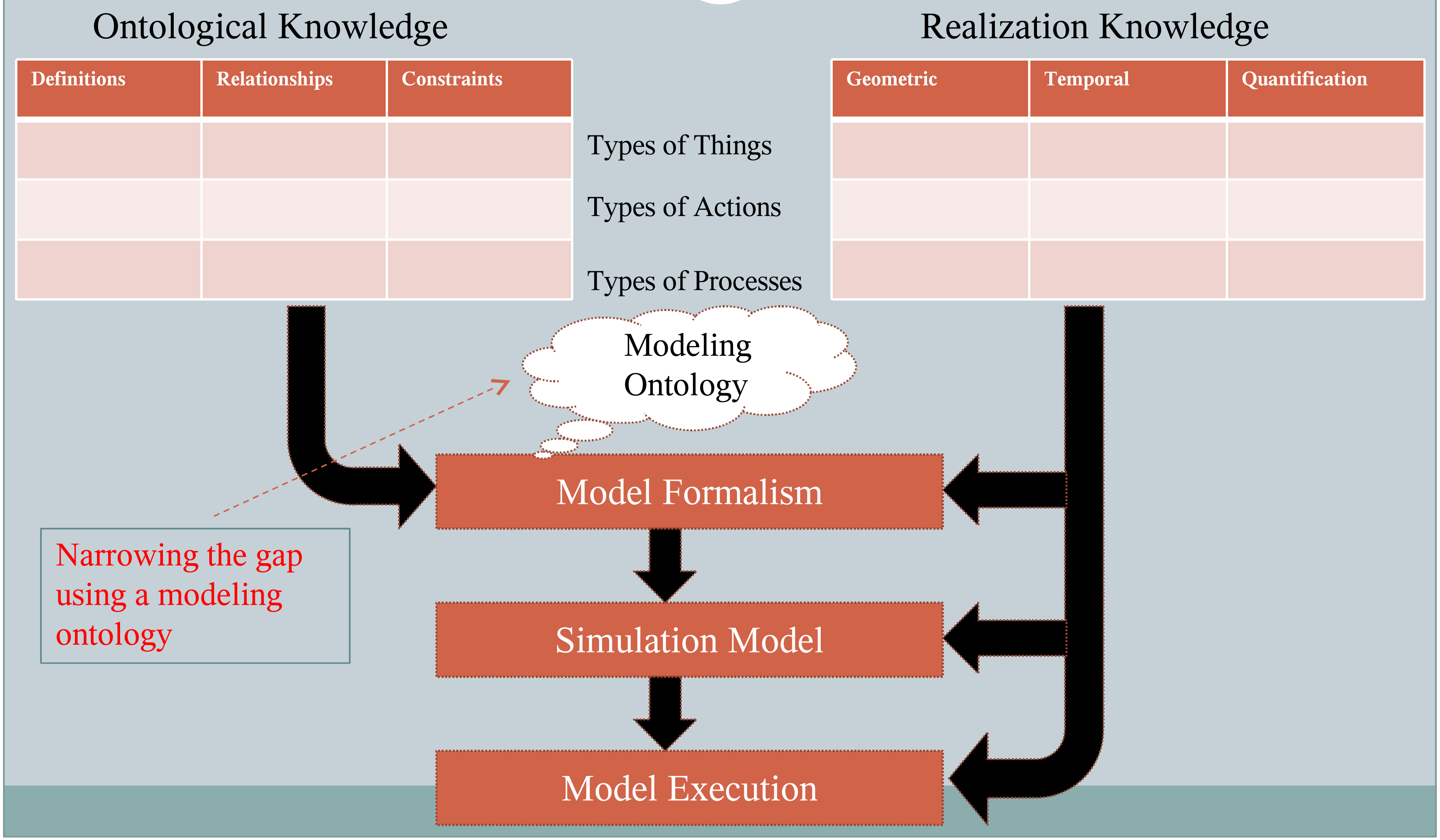
Simulation Model

Represents reality by modeling the functions as they occur over time and/or space.

Focuses primarily behavior of a system

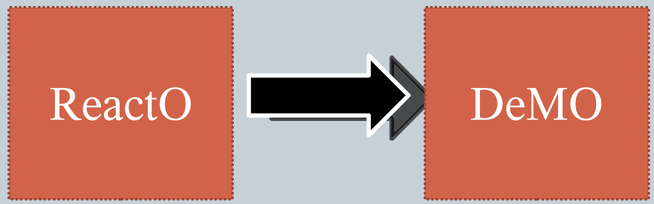


Exploiting Existing Knowledge

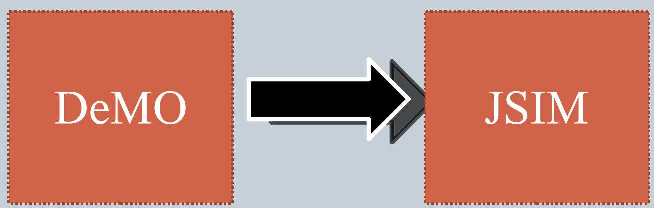


Three Areas of Interoperability

- Domain ontologies and modeling ontologies



- Modeling ontologies and discrete-even simulations



- Simulations using different worldviews

Using Domain Ontologies to Drive Simulation

- ReactO (Reaction Ontology)
  - Ontology instances used to model biochemical pathways
- DeMO (Discrete-event Modeling Ontology)
  - Ontotology instances represent simulation components
- Concept Mapping
  - ReactO concepts may be mapped to DeMO concepts
- Ontology Driven Simulation (ODS)

Phases of Ontology-driven Simulation

- Domain ontology logically linked to simulation software via modeling ontology
- Mapping Phase
  - Bridge for transferring knowledge from domain ontology to DeMO (syntactically compatible, semantically meaningful)
- Transformation Phase
  - Uses transferred knowledge to create DeMO-based model conforming to a particular formalism
- Generation Phase
  - Uses DeMO-based model to generate executable simulation

Ontology Modeling vs. Simulation Modeling

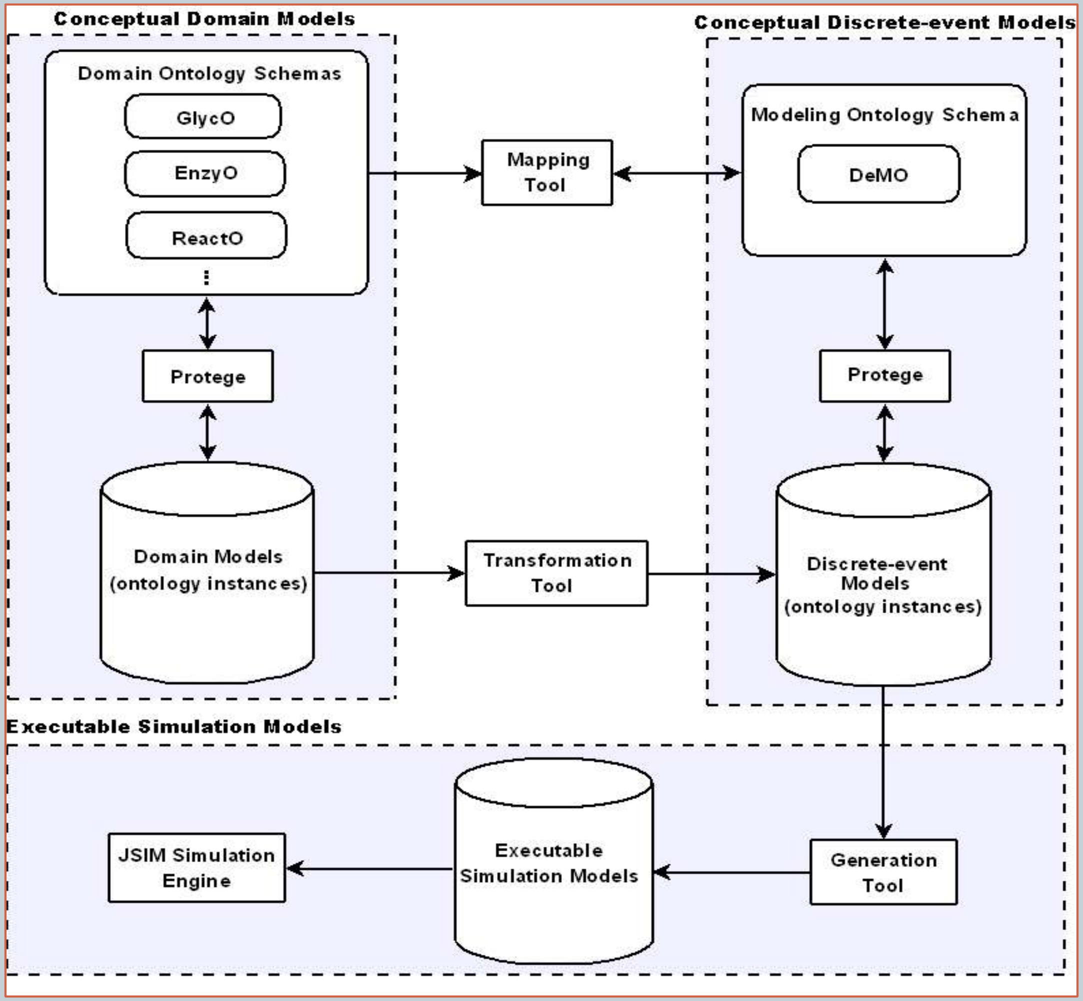
	Ontology Modeling	Simulation Modeling
Explanatory Power	Descriptive	Predictive
Randomness	Typically Deterministic	Often Stochastic
Time Dependency	Usually Time Independent	Often Time Dependent
Spatial Specificity	Usually Just Topological	Often Geometric
Causality	Secondary (if at all)	Central Issue
Breadth	High to Medium	Low
Focus	Structure (nouns) and Function (verbs)	Behavior

2. Together they provide a more complete view of reality
3. They can feed off of each other

Research Challenges

- Essential Challenges:
  - Transformation and supplementation of ontology models to form simulation models (ontology => simulation) *"our focus"*
  - Extraction of meta-data from simulation models to populate ontologies (simulation => ontology)
- Size and complexity of ontologies make mapping difficult
  - Typical ontology alignment techniques not applicable to mapping for ODS
  - Implementation of user assisted graphical techniques
- Reuse of mappings for similar scenarios or submodels
- Selecting preferred mappings amongst multiple possiblities

DeMOForge ODS Tool



DeMOForge Mapping Support

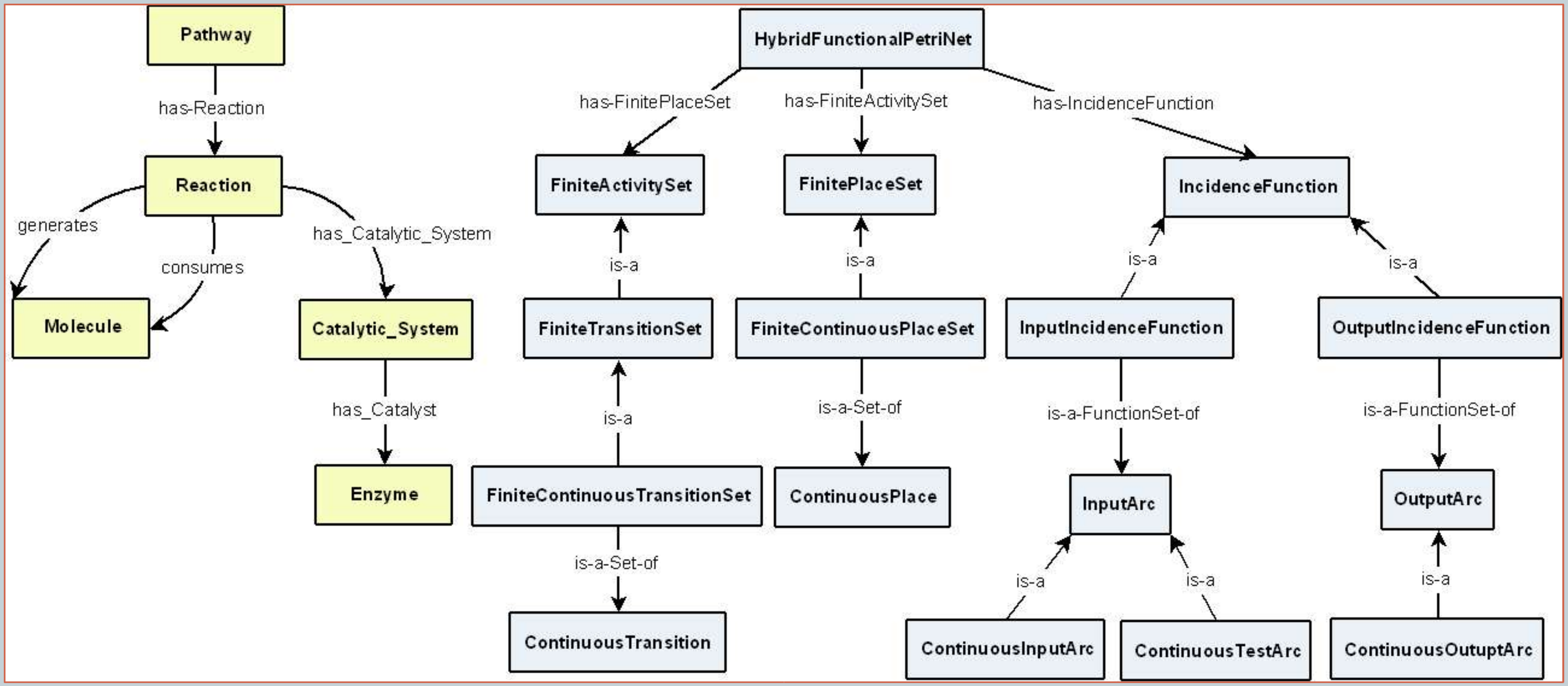
- Traditional ontology mapping is inadequate
  - Equivalency or subsumption *(automobile = car)*
- ODS ontology mapping is fundamentally different
  - Concepts connected as analogs, not synonyms

Biochemical Pathway Example Using ReactO

ReactO Ontology class	DeMO Ontology class
Pathway	HybridFunctionalPetriNet
Reaction	ContinuousTransition
Molecule	ContinuousPlace
Enzyme	ContinuousPlace

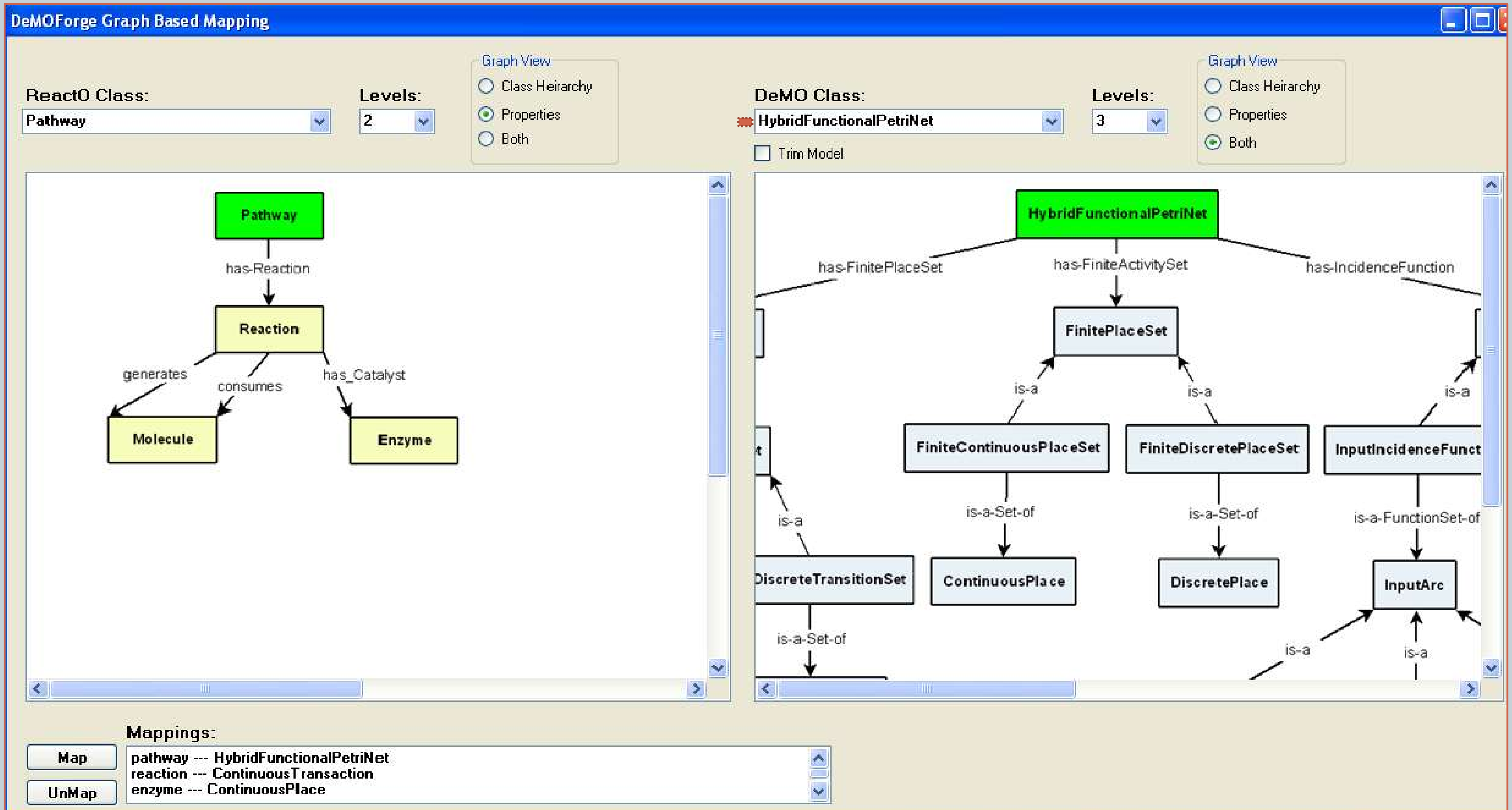
ReactO Ontology property	DeMO Ontology class
Reaction **consumes** Molecule	ContinuousInputArc
Reaction **generates** Molecule	ContinuousOutputArc
Reaction **… has-Catalyst** Enzyme	ContinuousTestArc

ReactO Pathway and DeMO HFPN Classes

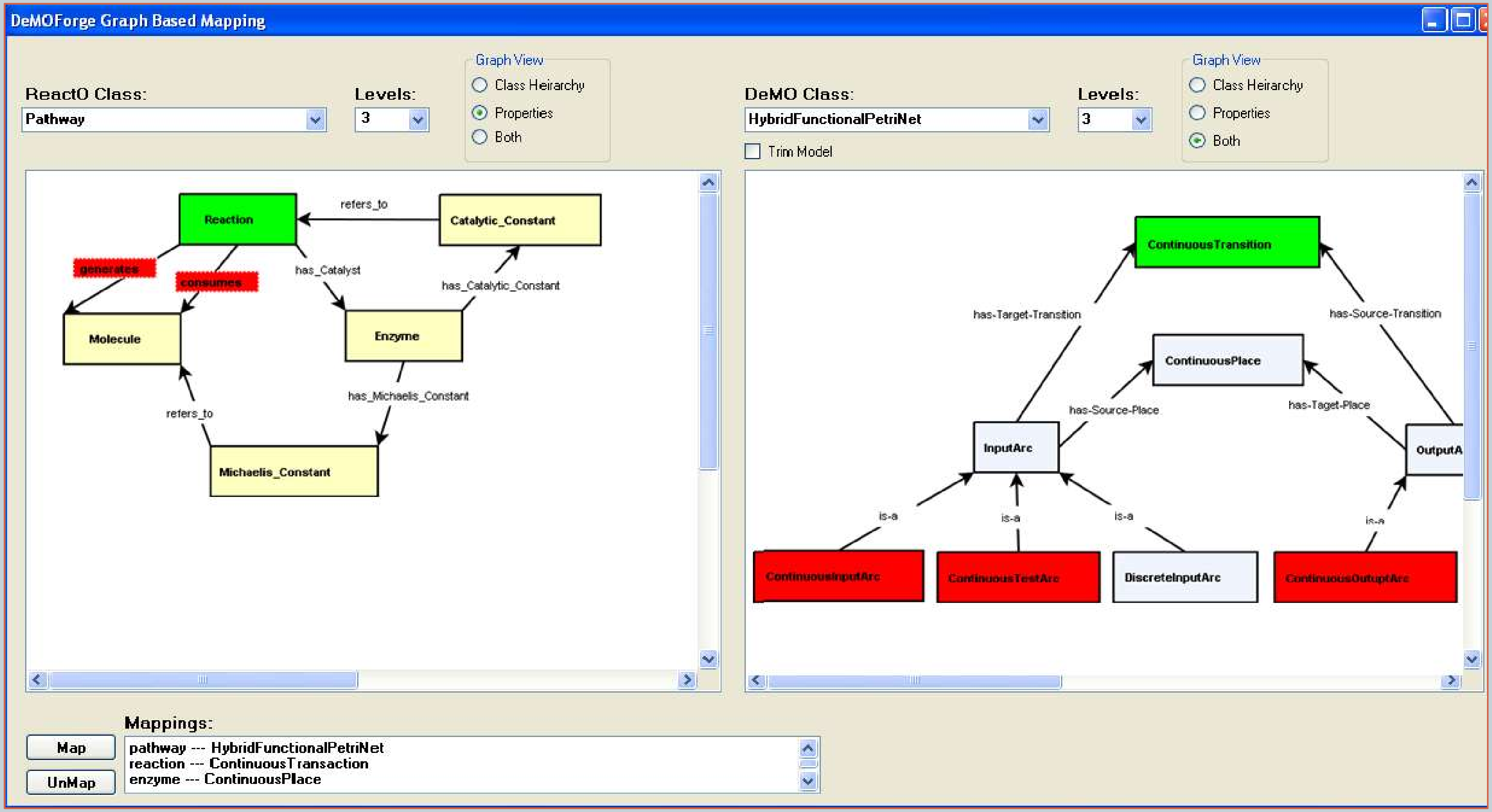


Slideware to show mapping

DemoForge Mapping Support



DemoForge Mapping Support

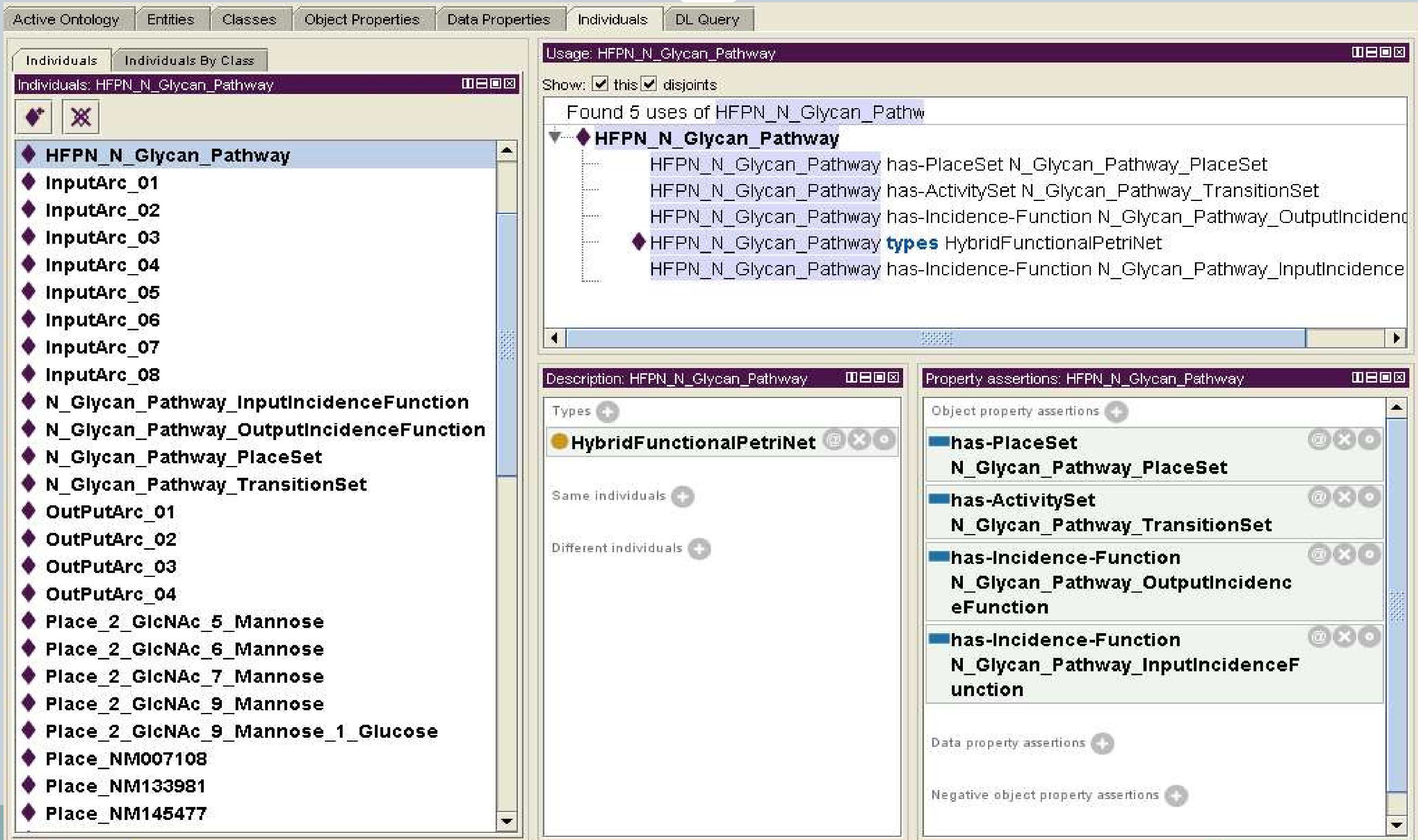


DemoForge Transformation Phase Support

1. SWRL* queries generated to retrieve ReactO instances representing pathway model
2. Mappings used to create DeMO instances based on ReactO instances.
3. SWRL rules generated based on the DeMO HFPN* class structure
4. Rules from step 3 used to complete the creation of the DeMO-based pathway model

•Semantic Web Rule Language
•Hybrid Functional Petri Net

Transformation Output

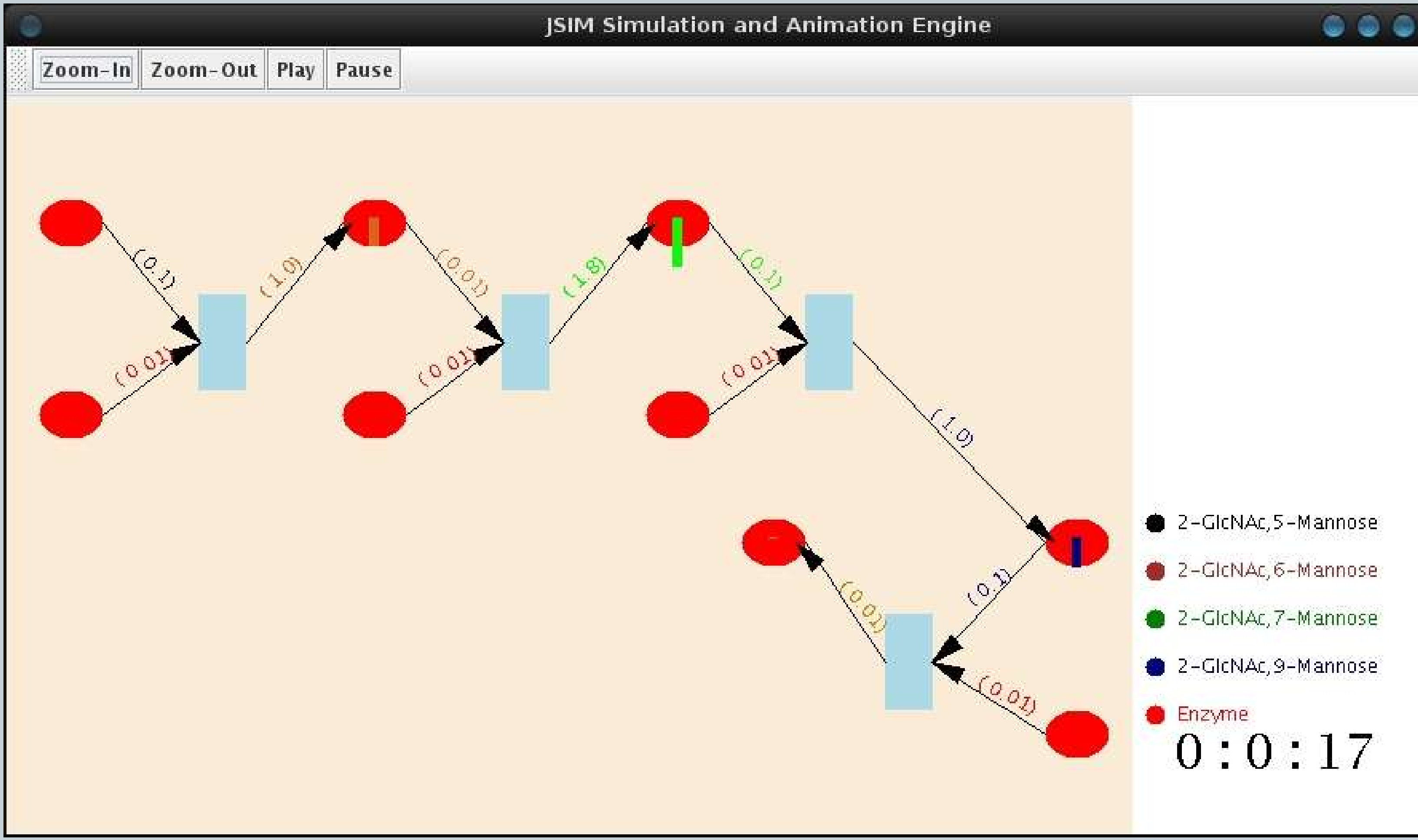


DemoForge Generation Phase Support

- Code generator translates DeMO instances into executable simulation models

3. The OWL API from University of Manchester used to read DeMO instances
4. JSIM models are generated using a category specific code generator that includes a layout manager
5. Scenario management module retrieves additional information needed for simulation run

A HFPN Model of a Biochemical Pathway



Recapitulation: What Was Accomplished?

- Incorporation of ontological knowledge and realization knowledge in the creation of simulation models
- DeMO supports multiple modeling techniques
- Mapping of domain concepts to modeling concepts
  - Size and complexity mandates graphical user assisted techniques
  - Versioning for reuse
- Transformation of domain knowledge into discrete-event models
- Generation of executable simulations using DeMO-based models
- Documentation of the modeling process

Related Work

- Creating Modeling Ontologies
  - Process Interaction Ontology for Discrete Event Simulation (PIMODES) (Lacy 2006)
  - Component Simulation and Modeling Ontology (COSMO) (Teo and Szabo 2008)
- Ontology => Simulation
  - CODES (Teo and Szabo 2007)
- Simulation => Ontology
  - Investigating the role of ontologies in simulation (Taylor 2009)
  - PIMODES (Lacy 2006)
- Interoperability
  - Use of domain ontologies in agent-supported interoperability of simulations (Yilmaz and Paspuleti 2005)
  - Levels of conceptual interoperability (Tolk 2003)

Conclusions

- First complete end-to-end ODS methodology
  - Ontologies utilized to drive key phases of model development
  - Expedites the development of simulation models
  - Facilitates the reuse of application knowledge
  - Utilized Three Distinct Phases
    - Ontology Schema Mapping
    - Domain ontology instances => DeMO instances
    - DeMO instances => Executable simulation model
- Ontology knowledge and realization knowledge incorporated to drive the development simulation models
- Mapping information provides a basis for documenting the model development process.

Future Work

- Enhance the DeMO ontology to support hierarchical modeling
- Add functionality which allows DeMOForge to transform DeMO-based models from one formalism to another
- Enhance the ODS methodology to more formally support the documentation of model development
- Evaluation of the effectiveness of the ODS methodology