

A Standard-based approach to interoperability: the Process Specification Language (PSL)

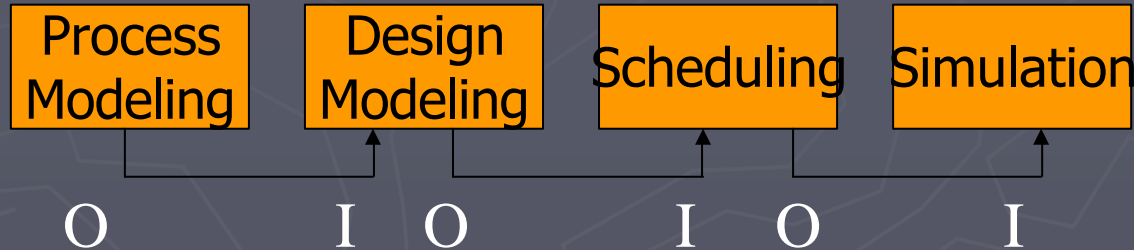
Line Pouchard
Computer Science and Mathematics Division
Oak Ridge National Laboratory
Department of Energy (US)

Sponsor: National Institute of Standards and Technology
(NIST)

The need for interoperability

- ? The ability to **share technical and business information throughout an extended enterprise** (supply chain) implies:
- The ability of software on multiple machines from multiple vendors to communicate, in particular,
 - The ability to exchange data, processes and information

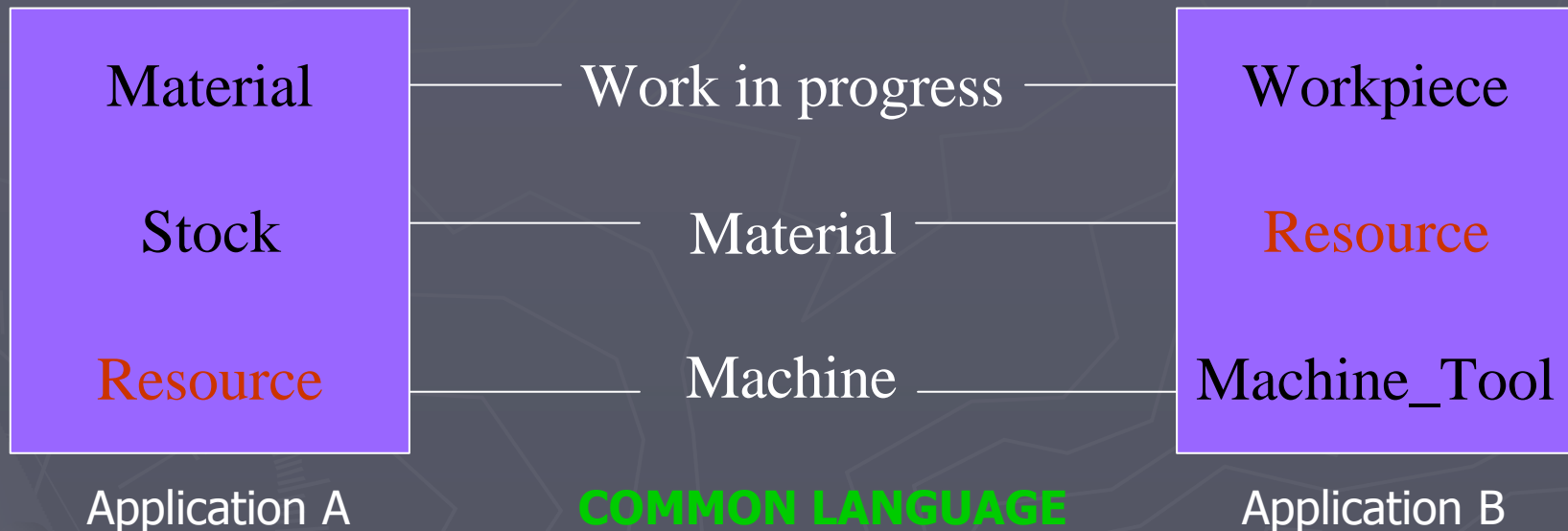
Example: the design life cycle



- ? The cost of a lack of interoperability:
- The US automotive sector expended **\$ 1 billion per year in 1999 to resolve interoperability issues**
 - 50% of this cost attributed to data file exchange.
 - Study commissioned by the National Institute of Standards and Technology

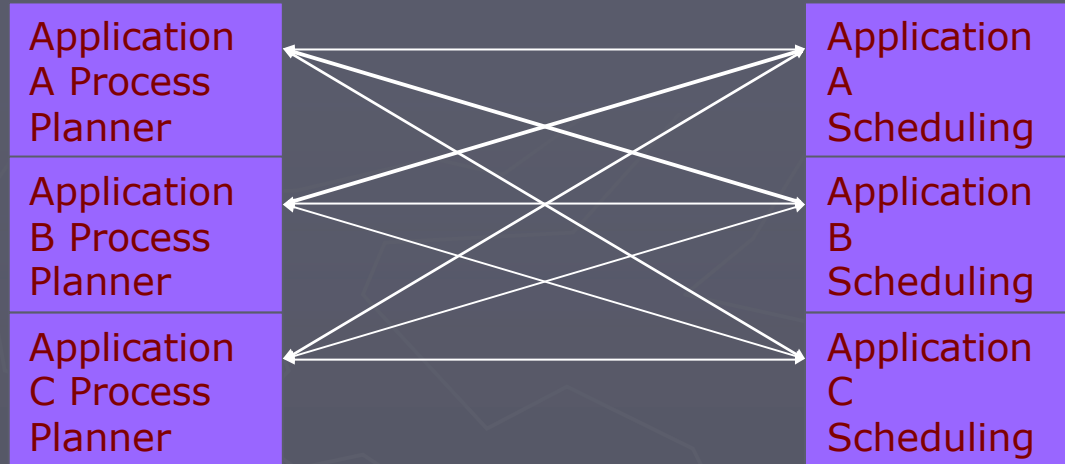
Challenges to interoperability

- ? Creation and support of virtual enterprises hindered by the lack of a common understanding of their business processes



- ? A common language for processes enables the integration of the business practices of partners within the virtual enterprise

When should we use the Process Specification Language?



Scenario 1

- n planners,
- m schedulers,
- $m * n$ translators

Scenario 2: use of PSL

- n planners,
- m schedulers,
- $m + n$ translators



Who is the target audience for PSL?

? End users

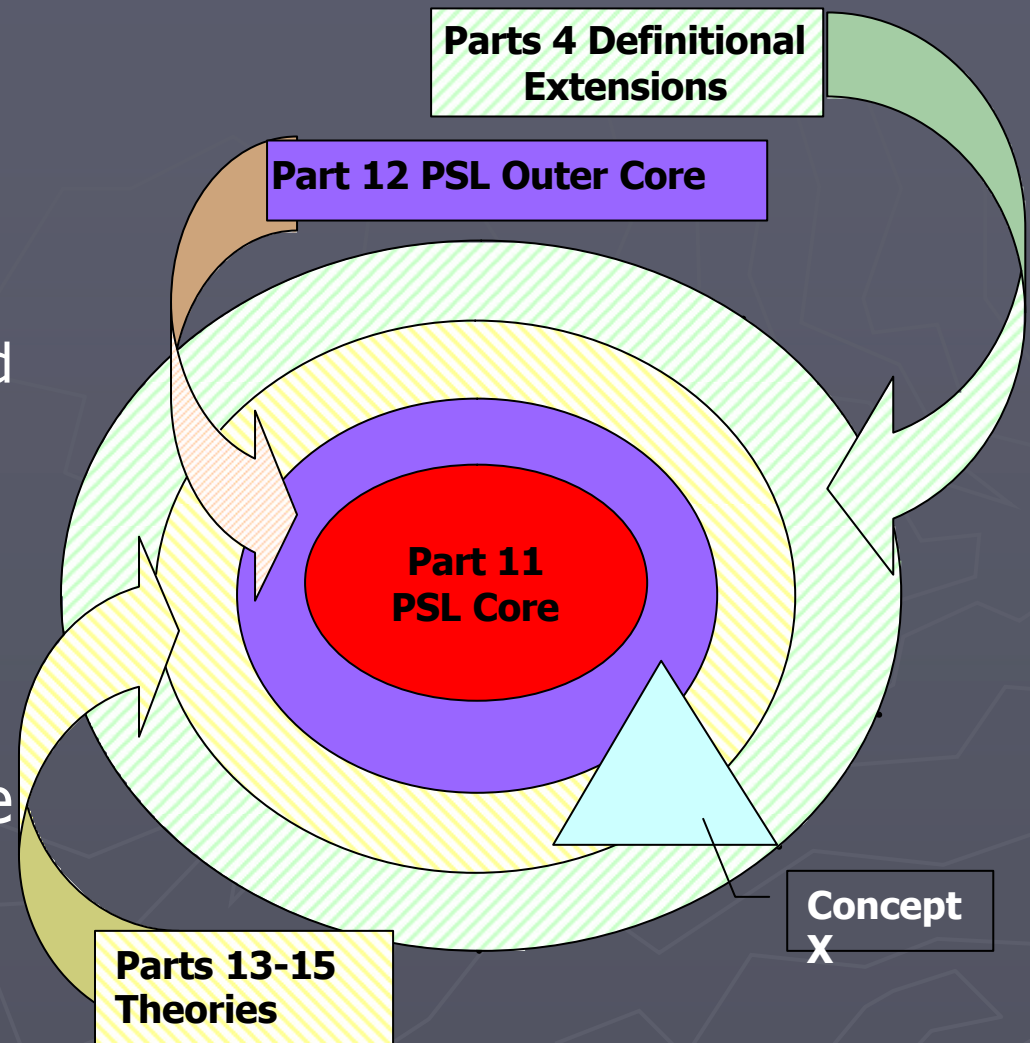
- Engineers, staff, anyone who needs to exchange process information among applications within their company
- Collaborators from partnering companies
- Suppliers and customers in vertical integration

? Prescribers

- Software developers and vendors who produce the design tools, such as CAD, project planning softwares, etc...need to incorporate PSL translators into their tools
- Make design tools PSL compliant.

The Architecture of PSL

- ? A computer-readable language
 - specifies entities such as concepts, terminology, and relationships for manufacturing processes
 - includes axioms constraining the use of entities and relationships between them
- ? Structure of the language
 - A core and outer core
 - Theories
 - Definitional extensions



Standardization: ISO 18629, SC4, TC 184

? Part 10 Series: Core Theories

- Part 11 : PSL-Core
- Part 12 : Outer Core
- Part 13 : Duration and Ordering Theories
- Part 14 : Resource Theories
- Part 15 : Actor and Agent Theories

? Part 2X: External Mappings

- EXPRESS, XML, UML

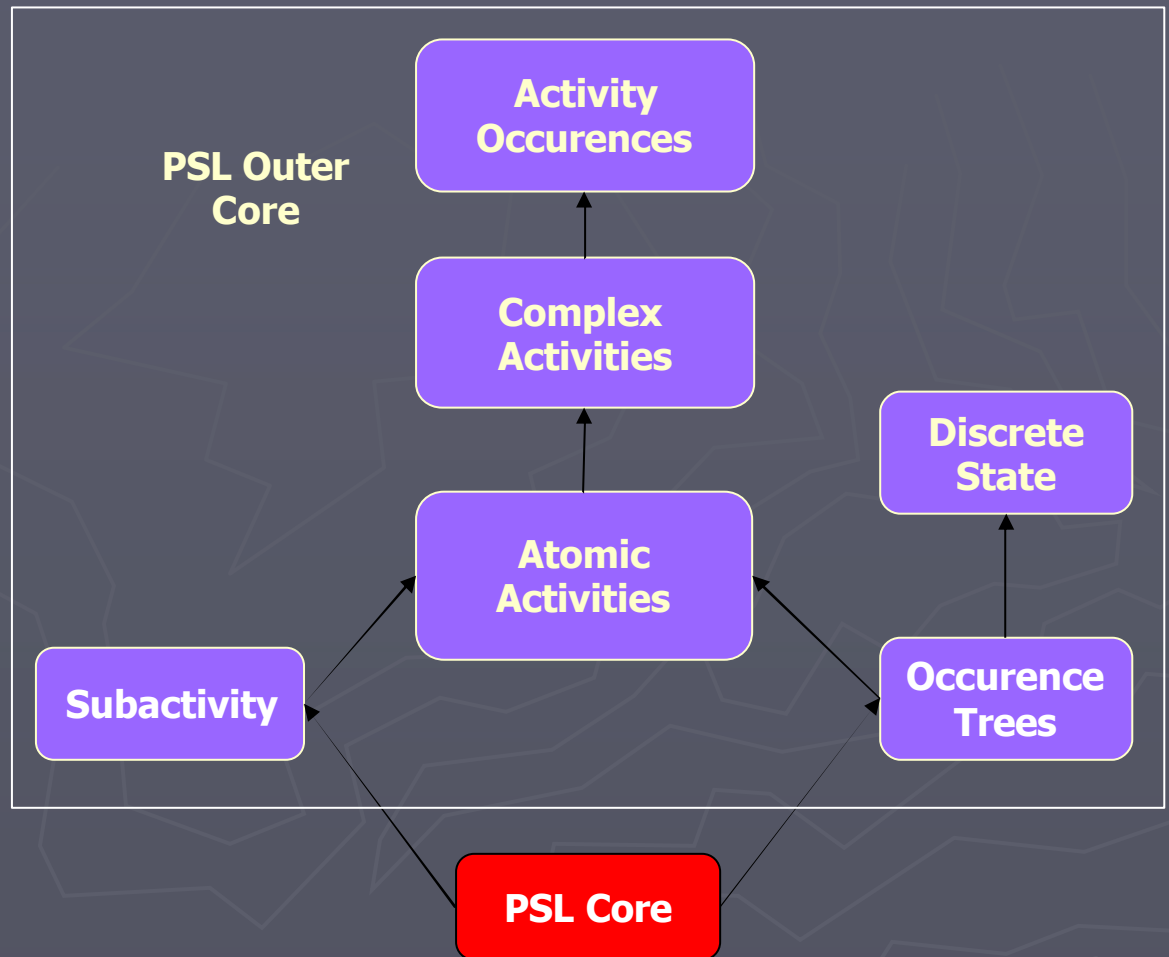
? Part 40 Series: Definitional Extensions of PSL

- Part 41 : Activity Extensions
- Part 42 : Temporal and State Extensions
- Part 43 : Activity Ordering and Duration Extensions
- Part 44 : Resource Roles
- Part 45 : Resource Sets
- Part 46 : Processor Activity Extensions

PSL Core, Outer-core, and Dependencies

PSL Core

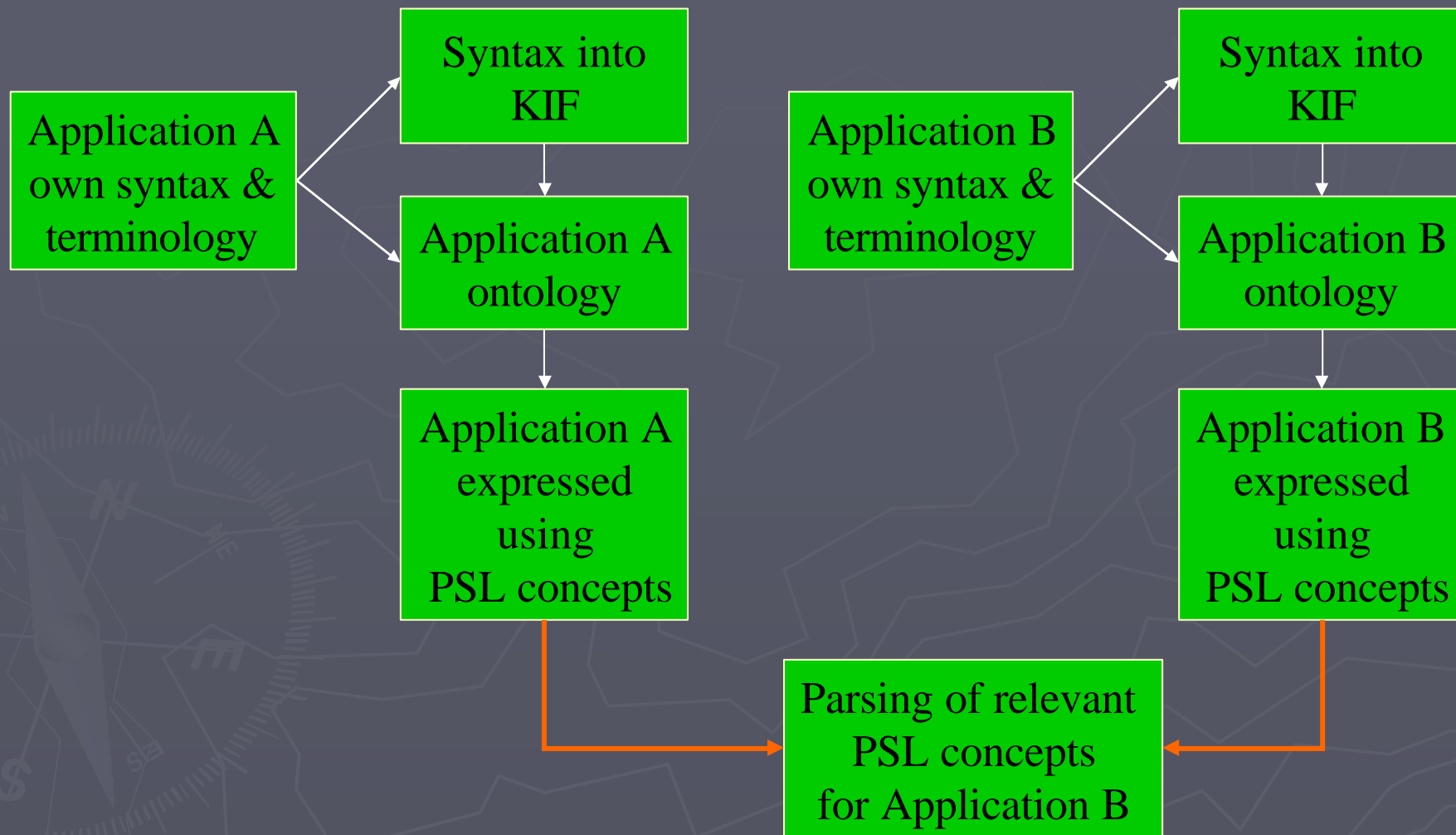
- ? Activity
- ? Activity_occurrence
- ? Timepoint
- ? Object
- ? Before
- ? Occurrence_of
- ? Participates_in
- ? Beginof
- ? Endof
- ? Inf+, Inf -



Using PSL for translating between two applications

- ? Syntactic translation for two applications
 - Application native syntax is parsed to PSL syntax
 - Application concepts are expressed in KIF
- ? Semantic translation
 - KIF definitions are written for application concepts using PSL reserved terminology and relations
 - A 20-question wizard assists in the process
- ? Application A concepts are translated to PSL
- ? Application B concepts are translated to PSL and a reverse index is created.

Process Exchange using PSL



Expressing Application A concepts using PSL

1

```
(forall (?r)
  (=> (inject_mold)
      (rA ?r)))
```

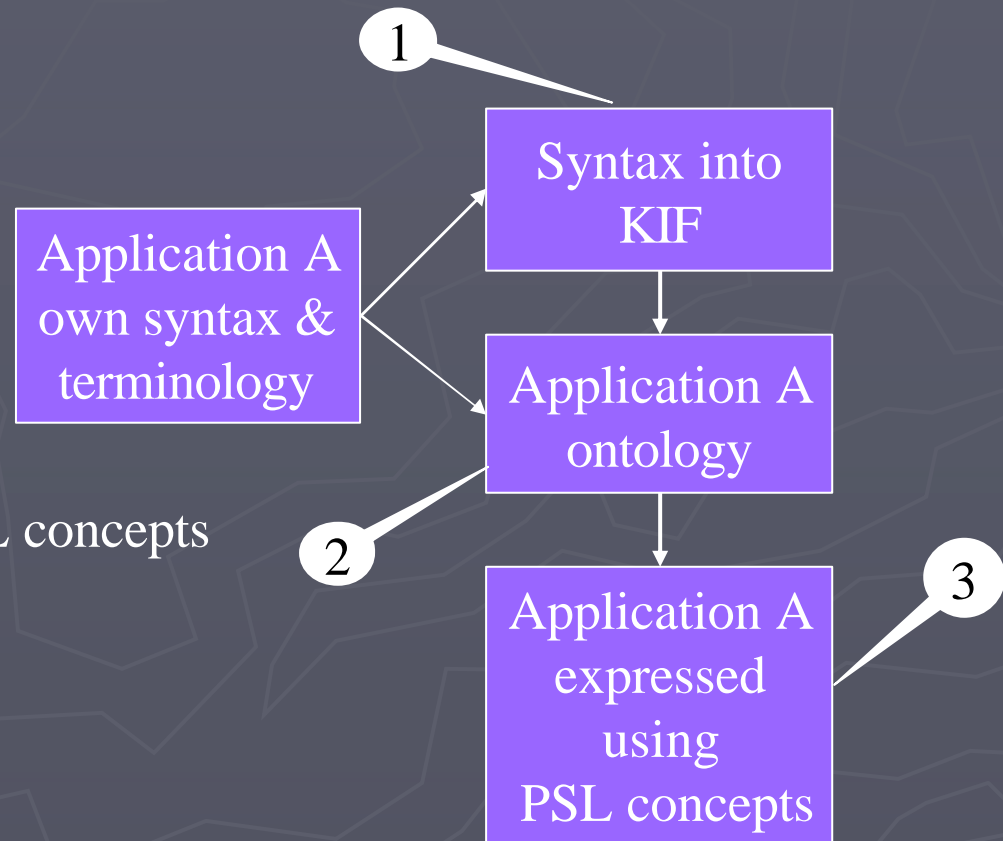
2

```
(forall (?r ?a)
  (<=> (rA ?r)
      (exists (?a)
        (reusable ?r ?a))))
```

3

Unconditional mapping to PSL concepts

```
(forall (?r)
  (<=> (rA ?r)
      (and resource ?r)
      (reusable ?r)))
```

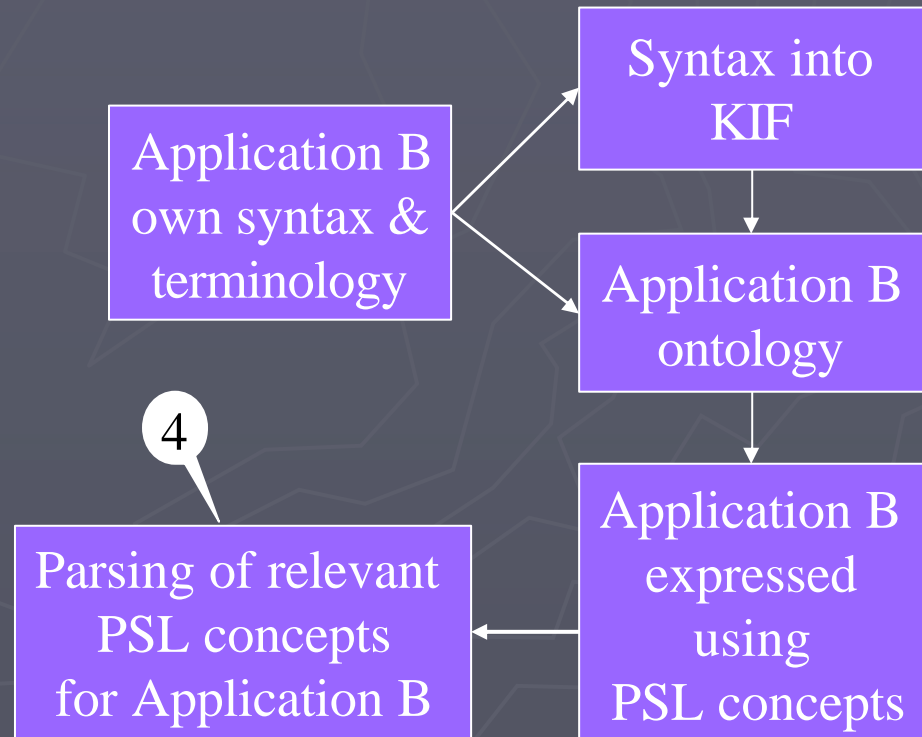


Expressing Application B concepts using PSL

4

One additional step:
inverting the table

(Application B \Rightarrow PSL)
(PSL \Rightarrow Application B)



Twenty Question Wizard

PSL Manufacturing Systems Integration Division

NIST National Institute of Standards and Technology

20 Questions

home >> 20 questions >> atomic activities

Classes of Atomic Activities

Class Name:

Constraints on Atomic Activity Occurrences

The first set of questions characterizes the conditions under which atomic activities can possibly occur; these are often referred to as the preconditions for the activity.

- 1. Unconstrained Atomic Activity Occurrences**

Are there any constraints on the occurrence of the atomic activity?

 - There are no constraints on the occurrences of the atomic activity; the activity is always possible.
 - There are constraints on the occurrences of the atomic activity, but the activity is always possible after occurrences of certain other activities.
 - There are constraints on the occurrences of the atomic activity, there are circumstances under which the activity is not possible and cannot occur, no matter what other activity occurrence preceded it.
- 2. Constraints on Atomic Activity Occurrences based on State**

Are the constraints on the occurrence of the atomic activity based only on the state prior to the activity occurrence?

 - Any occurrence of the activity depends only on fluents that hold prior to the activity occurrence.

Privacy/Security Notice/Disclaimer.
NIST is an agency of the U.S. Commerce Department's Technology Administration.
Technical Inquiries: PSL Team

Interoperability in Construction

- ? Scenario: The design and construction of an office building
- ? Includes an exchange of data regarding fitting a metal door to a wall frame
- ? Use of an AutoCAD software
- ? Use of Microsoft Project for the planning phase

Credit: Genet Tesfagaber, Loughborough University

AutoCAD Process "door frame assembly" using PSL

```
(forall (?a)
  (⇔(doorframe_assembly)
    (and (activity ?a)
      (constrained ?a)
      (markov_precond ?a)
      (rigid_time ?a)
      (rigid_mixed ?a)
      (context_free ?a)
      (markov_effects ?a)
      (non-temporal ?a)
      (rigid_mixed effects ?a)
    )))
```

the activity occurrence is not possible under certain circumstances

the occurrence depends on the state of other activities

the occurrence does not depend on the duration of other activities

MS Project Task "door frame assembly" using PSL

```
(forall (?a)
  (↔(doorframe_assembly)
    (and (activity ?a)
      (constrained ?a)
      (markov_precond ?a)
      (time_precond ?a)
      (mixed_precond ?a)
      (context_free ?a)
      (rigid_state_effects ?a)
      (rigid_time_effects ?a)
      (rigid_mixed_effects ?a)
    )))
```

the activity occurrence is not possible under certain circumstances

the occurrence depends on the state of other activities

the initiation depends on the duration of other activities

Future challenges and issues

- ? Logically defined specification of processes
 - Mathematical operators and logical format
 - Quality/success is measurable (consistency checking)
- ? Automation of translation has been prototyped
 - Implemented translation of processes between an IDEF3 planner and a C++ scheduler
- ? May serve as an ontology for an agent-based system
- ? Implementation of translators in the products of software vendors
 - Socio-cultural issues prevail
- ? Need to reach critical mass for design and associated software applications
- ? Currently no Web implementation or use of state-of-the-art Web protocol
- ? Diffusing and accessing the standard

Acknowledgements and contacts

- ? Pr. Anne-Francoise Cutting-Decelle
- ? Pr. Robert Young
 - University of Loughborough
- ? Jean-Jacques Michel
 - Idpiconseil, ISO SC4 convener for PSL and Mandate
- ? Michael Gruninger (NIST)
- ? <http://www.mel.nist.gov/psl>
- ? <http://www.csm.ornl.gov/~7lp>