The Information Systems Modeling Suite: Modeling the Interplay between Information and Processes

Jan Martijn van der Werf
Artem Polyvyanyy
Information system: structure and process

- **Information modeling**
  - Structure of data
  - Data retrieval and querying
  - Data manipulation
...

- **Process modeling**
  - Flow of tasks
  - Who manipulates data, and when?
  - How is data transferred from A to B?

How to model and simulate their interplay?

Data models

Process models
A purchase process system

- For each product:
  - create order, let suppliers bid, select the best offer
  - Constraint: at least two different suppliers should bid!
A purchase process system

For each product:
create order, let suppliers bid, select the best offer
Constraint: at least two different suppliers should bid!

How to model and simulate their interplay?
Information Systems Modeling Language

1. Information model
2. Process model: Petri nets with identifiers
3. Specification to define the manipulations

How to model and simulate their interplay?
Information model

∀o ∈ order, s₁ ∈ supplier : (receives(s₁, o) ⇒ ∃s₂ ∈ supplier: (bids(s₂, o) ∧ s₁ ≠ s₂))

- Information model
  - Entity types
  - Relation types
  - Constraints
    - Subtype
    - Uniqueness
    - Mandatory

- Population
  - Sets for each entity & relation
  - Valid if all constraints hold

- Transaction
  - Add & remove entries from sets
  - Valid: if it results in a valid population
Petri nets with identifiers

- Each place has a *length*, denoted by $C(p)$
  - Implied by the vector of variables on the arcs
- Variable vectors on arcs
  - “Creator” variable: only on output
- Binding:
  - Maps each variable to ID
  - If “new”, use fresh ID
Petri nets with identifiers

- Each place has a \textit{length}, denoted by $C(p)$
  \textit{Implied by the vector of variables on the arcs}
- Variable vectors on arcs
  \textit{“Creator” variable: only on output}
- Binding:
  \textit{Maps each variable to ID}
  \textit{If “new”, use fresh ID}
Petri nets with identifiers

- Each place has a *length*, denoted by $C(p)$
  Implied by the vector of variables on the arcs
- Variable vectors on arcs
  "Creator" variable: only on output
- Binding:
  Maps each variable to ID
  If "new", use fresh ID
Combining information and process models

Identifiers point to entities
Combining information and process models

Identifiers point to entities

Transaction:
```
sendBid(supplier: supplier, order: order) {
    insert (s, o) into bids;
}
```
Combining information and process models

∀o ∈ order, s₁ ∈ supplier : (receives(s₁, o) ⇒ ∃s₂ ∈ supplier: bids(s₂, o) ∧ s₁ ≠ s₂)

transaction selectBid(supplier: supplier, order: order) {
    insert (s, o) into receives;
}

Create order
Request bid
Select best bid
Deliver
Process model: Petri net with identifiers (built on top of ePNK)

Information model: sets and relations
Information Systems Modeling Suite

Information model: sets and relations

Information model: Constraints in First-Order Logic
Transitions firings that will violate constraints
Conclusions

- Information Systems Modeling Suite
  - Internal Petri net simulator
  - Internal automated prover on finite sets
- Interplay between information and processes
- Future work
  - Incorporate information modeling in the suite itself
  - Develop modeling & analysis strategies
  - Extensive experimentation with students
The Information Systems Modeling Suite: Modeling the Interplay between Information and Processes

www.informationsystem.org/ismsuite/

Jan Martijn van der Werf
Artem Polyvyanyy
The information in this presentation has been compiled with the utmost care, but no rights can be derived from its contents.