VUB **STAR**.Lab

Software Process Validation: Comparing Process and Practice Models

Aldo de Moor
STARLab
Vrije Universiteit Brussel
ademoor@vub.ac.be

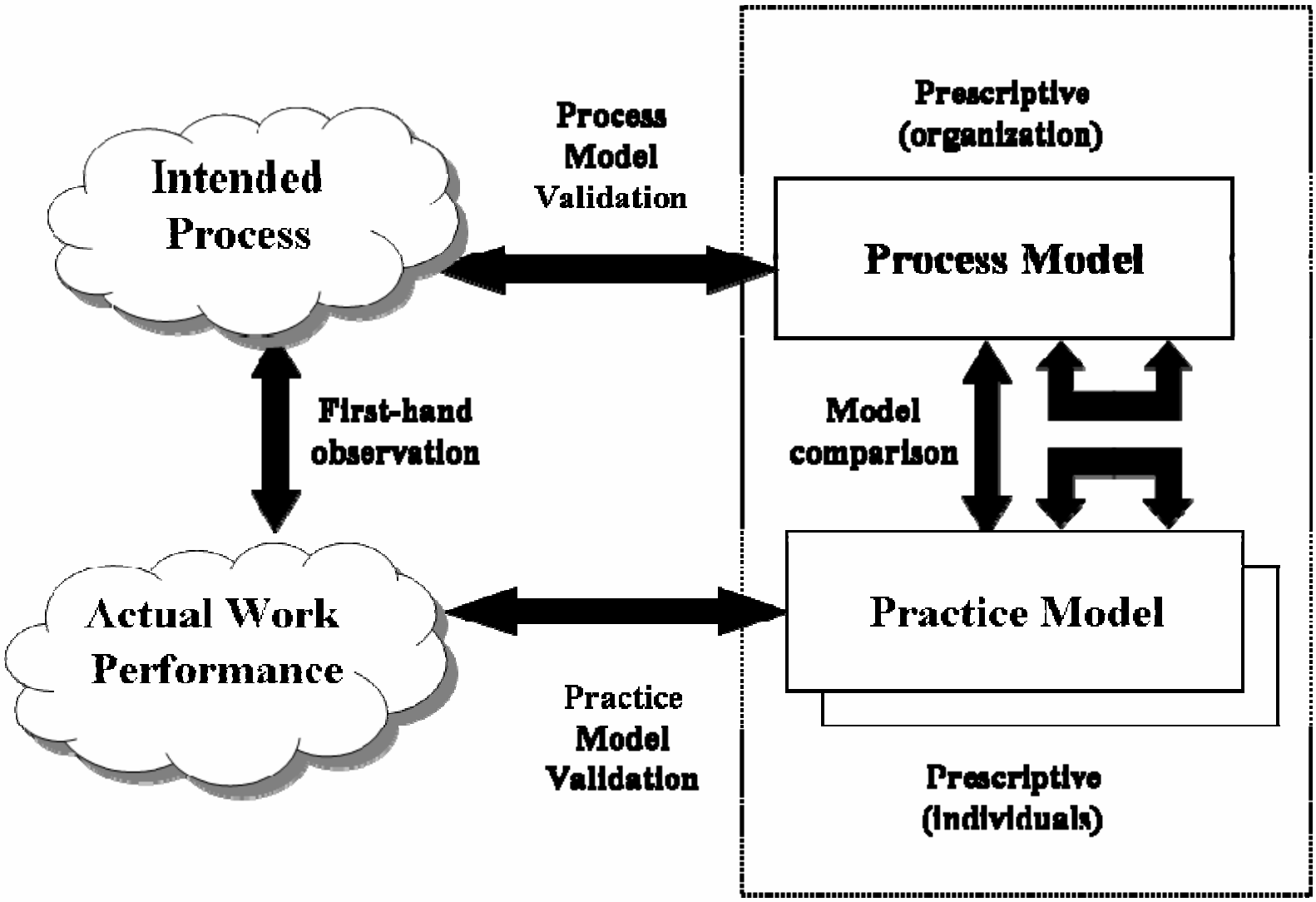
Harry Delugach
Dept. of Comp. Science
Univ.of Alabama in Huntsville
delugach@cs.uah.edu

Software Process vs Practice

VUB STAR.Lab

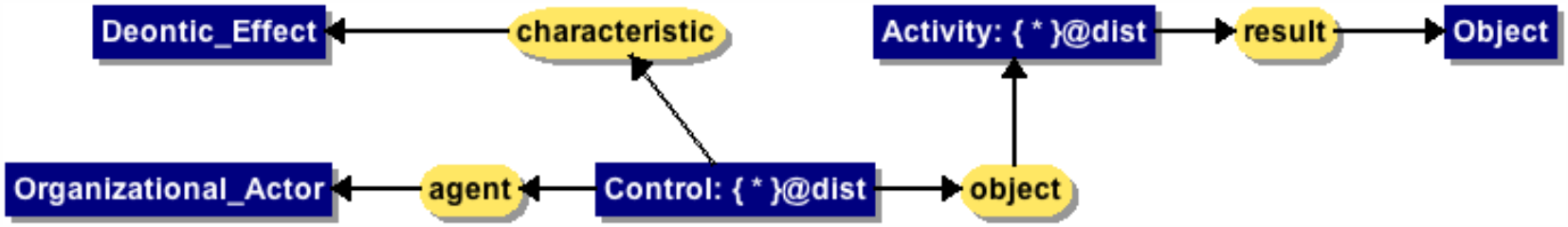
- Software process: the normative, prescribed way of doing things in a software development organization
  - Providing coherence
- Software practice: the way things really get done
  - Learning, innovation, dealing with contingencies
- Process model: intended software process
- Practice model: consolidate individual lessons learnt
- Aim: *detect* and *compare* differences between process and practice (models)
  - How large are the differences?
  - How significant are the differences?
  - Where do they occur?
  - According to whose viewpoint?

Software Process Validation



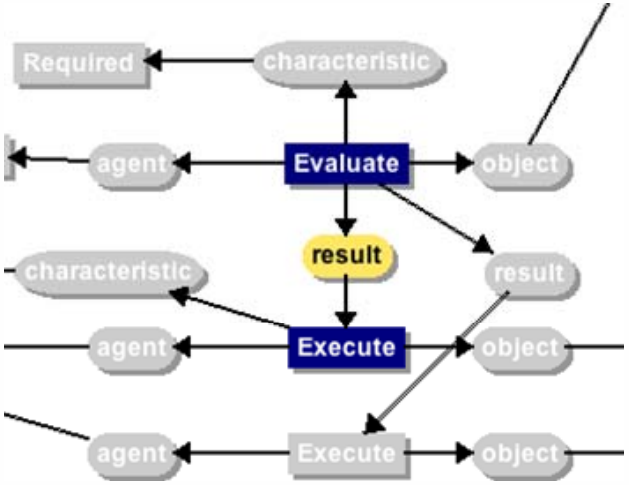
Key concepts

- Workflows > **Steps** > Activities



- Activity pattern (definitional, schematic)
- Difference graphs: using color

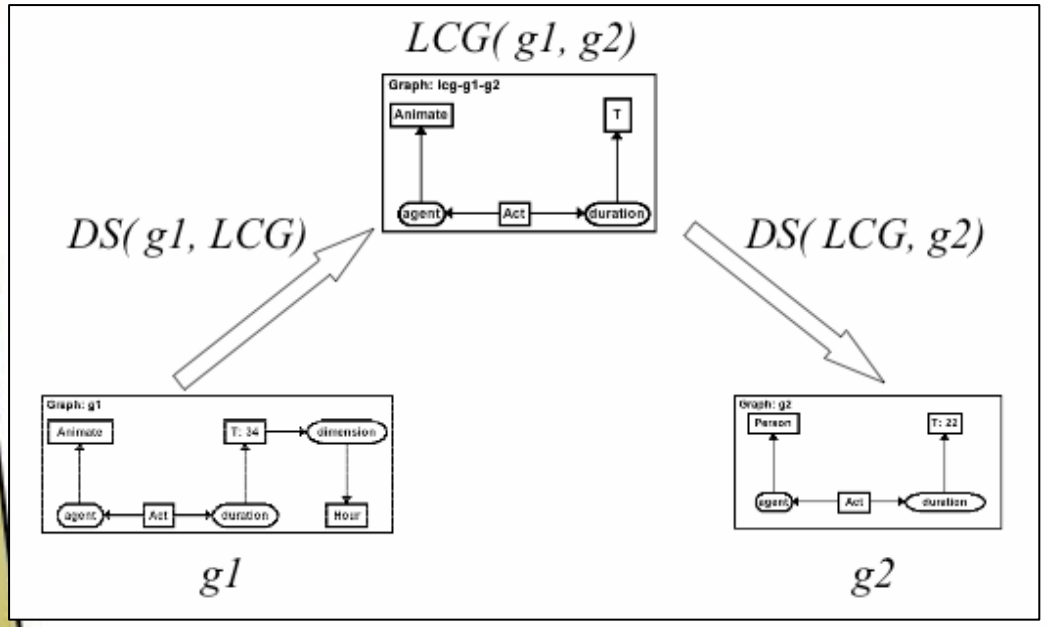
Coloring difference graphs (ICCS 2005)



Difference operation	Name	To be highlighted	Where
Specialize concept	**S-C**	Concept	Both graphs
Specialize relation	**S-R**	Relation	Both graphs
Generalize concept	**G-C**	Concept	Both graphs
Generalize relation	**G-R**	Relation	Both graphs
Insert subgraph	**I-G**	Added subgraph	Result graph
Delete subgraph	**D-G**	Deleted subgraph	Source graph
Insert co-referent link	**I-L**	Linked concepts	Result graph
Delete co-referent link	**D-L**	Un-linked concepts	Source graph
Move into context	**M-I**	Moved elements	Result graph
Move from context	**M-O**	Moved elements	Source graph

Calculating difference sequences (ICCS 2005)

VUB STAR.Lab



- Obtain the LCG of the two graphs
- Find a sequence of difference operations that transforms the first graph into the LCG
- Find a sequence of difference operations that transforms the LCG into the second graph.
- Concatenate the two sequences.

DS(G1, G2) :

G-C:	Generalize **Reqts_Engineer: *r** to **Reqs_Engineer**
G-C:	Generalize **Reqts_Engineer: *s** to **Reqs_Engineer**
D-G:	Delete **notEqual** relation
S-C:	Specialize **Software_Engineer** to **Software_Engineer: Jerry**
S-C:	Specialize **Reqts_Engineer** to **Reqts_Engineer: Jerry**

A formal method for software process validation

VUB STAR.Lab

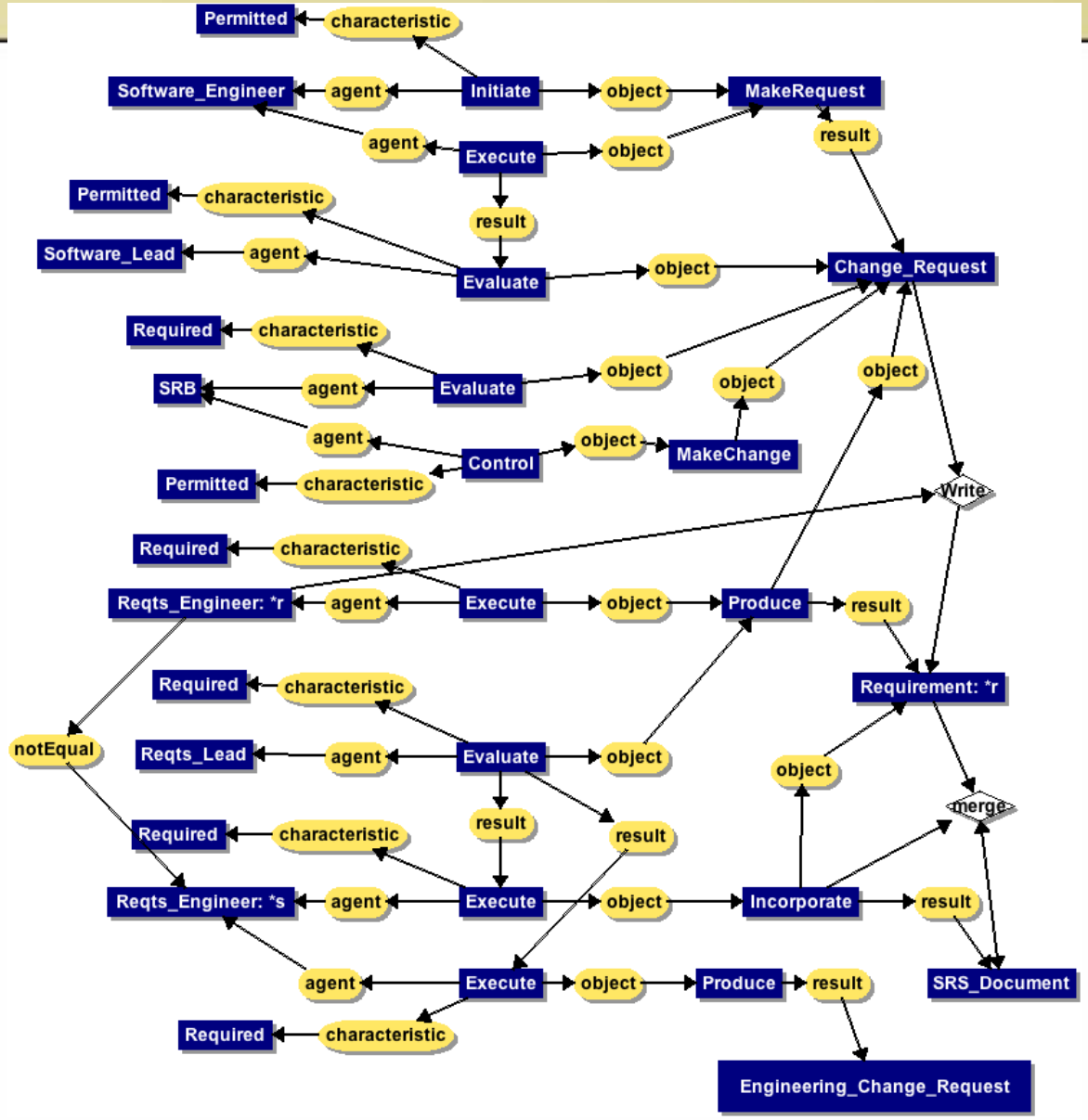
- **Create activity patterns for the model**
- **Create two models**
  - Models can be either process or practice models
  - Identify model steps
  - Build models by instantiating or specialising general activity patterns for each step
- **Compare the models**
  - Identify activities in the model, and delineate the steps in a sequence
  - Compare steps between the models
  - Create difference graphs
- **Interpret differences between the models**
  - Present difference graphs to software professionals
  - Adapt process and/or practice models
  - Launch change program

Case: Aerospace Software Engineering

VUB STAR.Lab

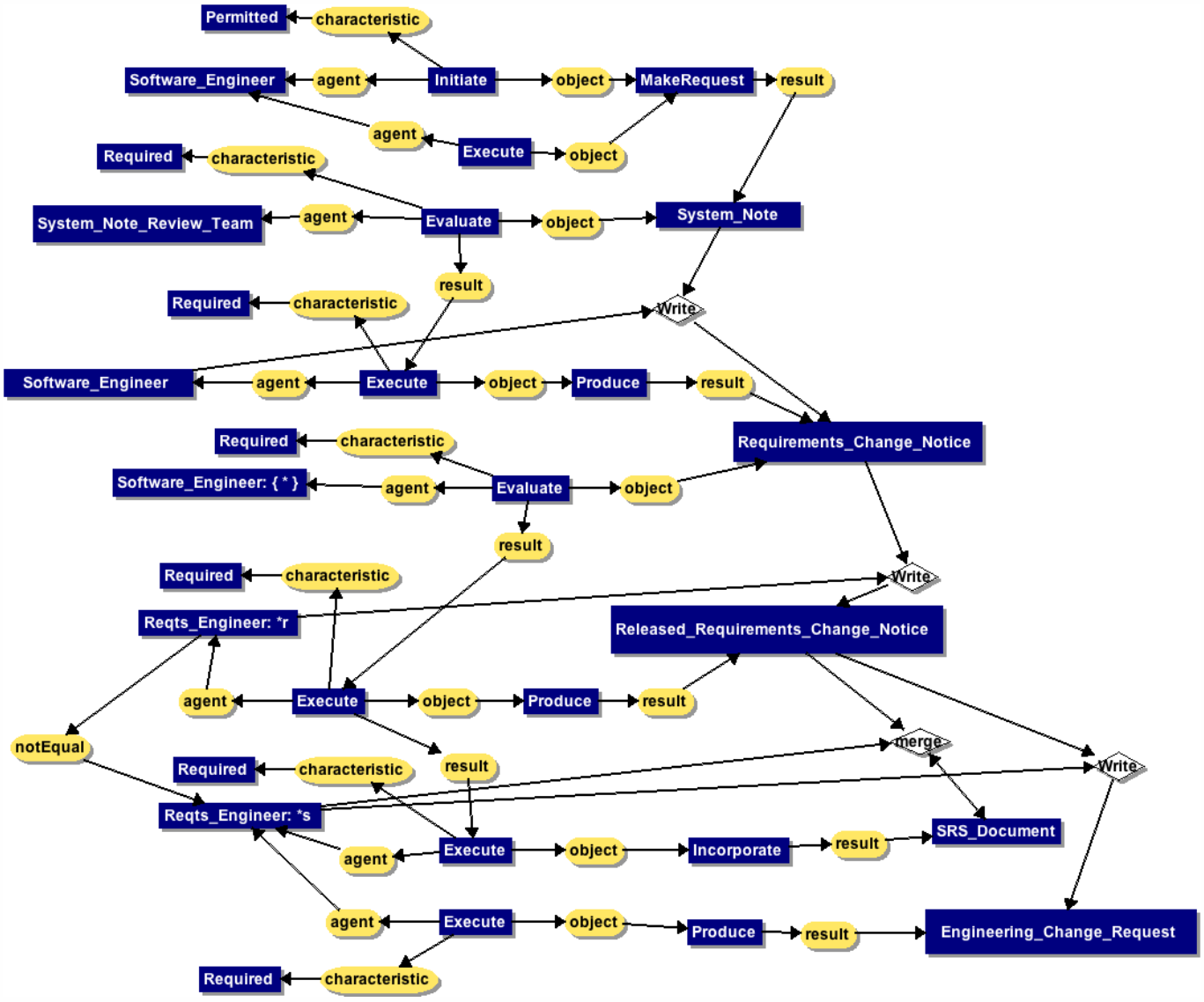
- Small internal software development group (10-20 persons)
- Multiple roles per person
- Proj. A: non-mission critical environmental system softw.
- Proj. B: mission-critical propulsion system softw.
- Examined documents, interviewed manager
- Compared process-practice models Proj. A (ICCS 2005)
- Compared process models Proj. A and B (EMMSAD 2006)

Proj A Process Model



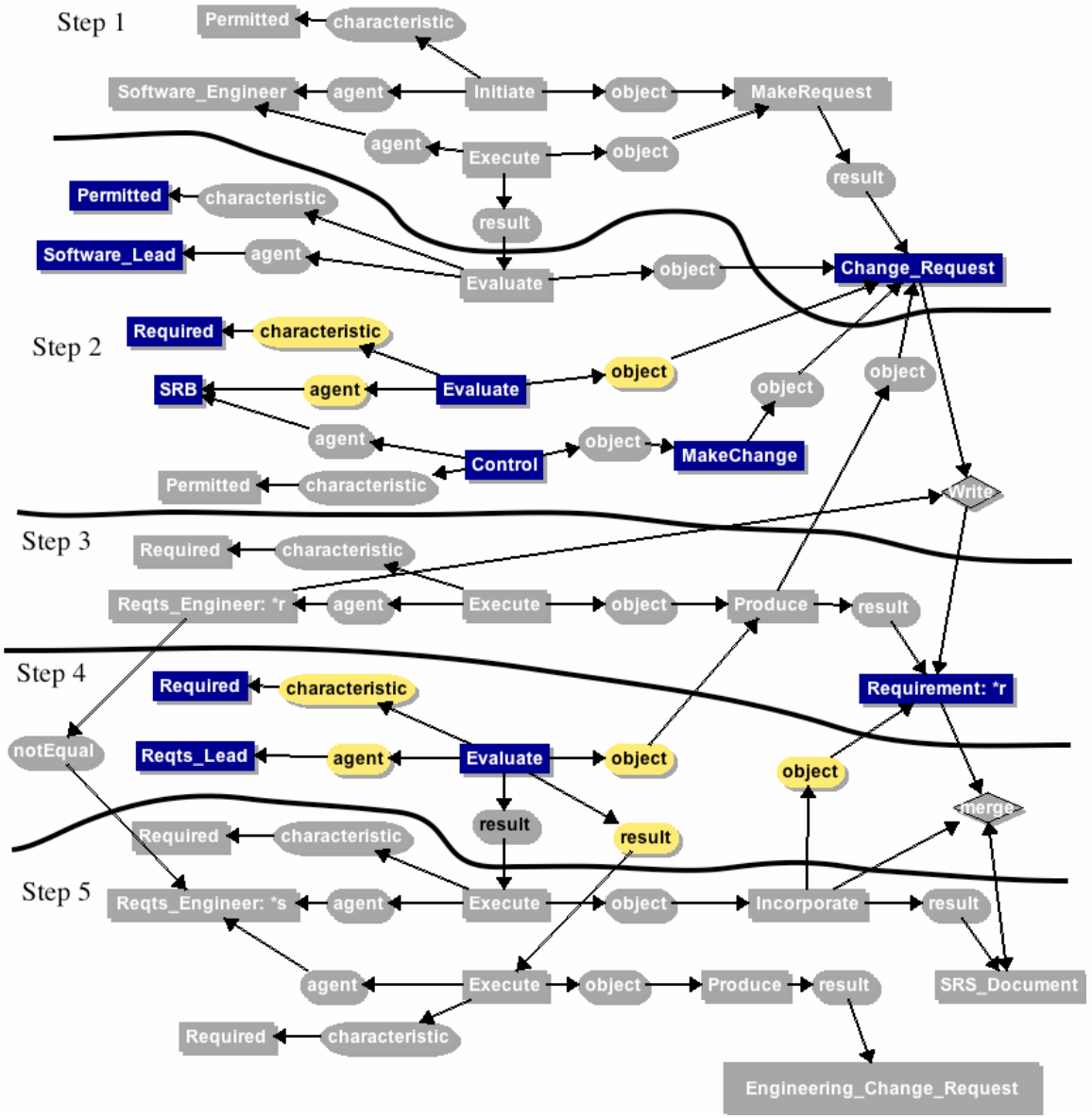
Proj B Process Model

VUB STAR.Lab



Difference graph Proj A-B process models (Proj A perspective)

VUB STAR.Lab



Conclusions

VUB STAR.Lab

- Software process validation essential
- Formal comparison of process-practice models is key
- Identified differences interpreted by software professionals help identify problems, opportunies for improvement
- Research agenda
  - Ontology? What core patterns (activity etc.) needed?
  - Knowledge representation and analysis:
    - Conceptual graph features (concept/relation type hierarchies, context, actors...)
    - Automating analysis (mappings, algorithms)
  - Visualization (delineating steps etc.)?
  - Elicitation / interpretation methodology