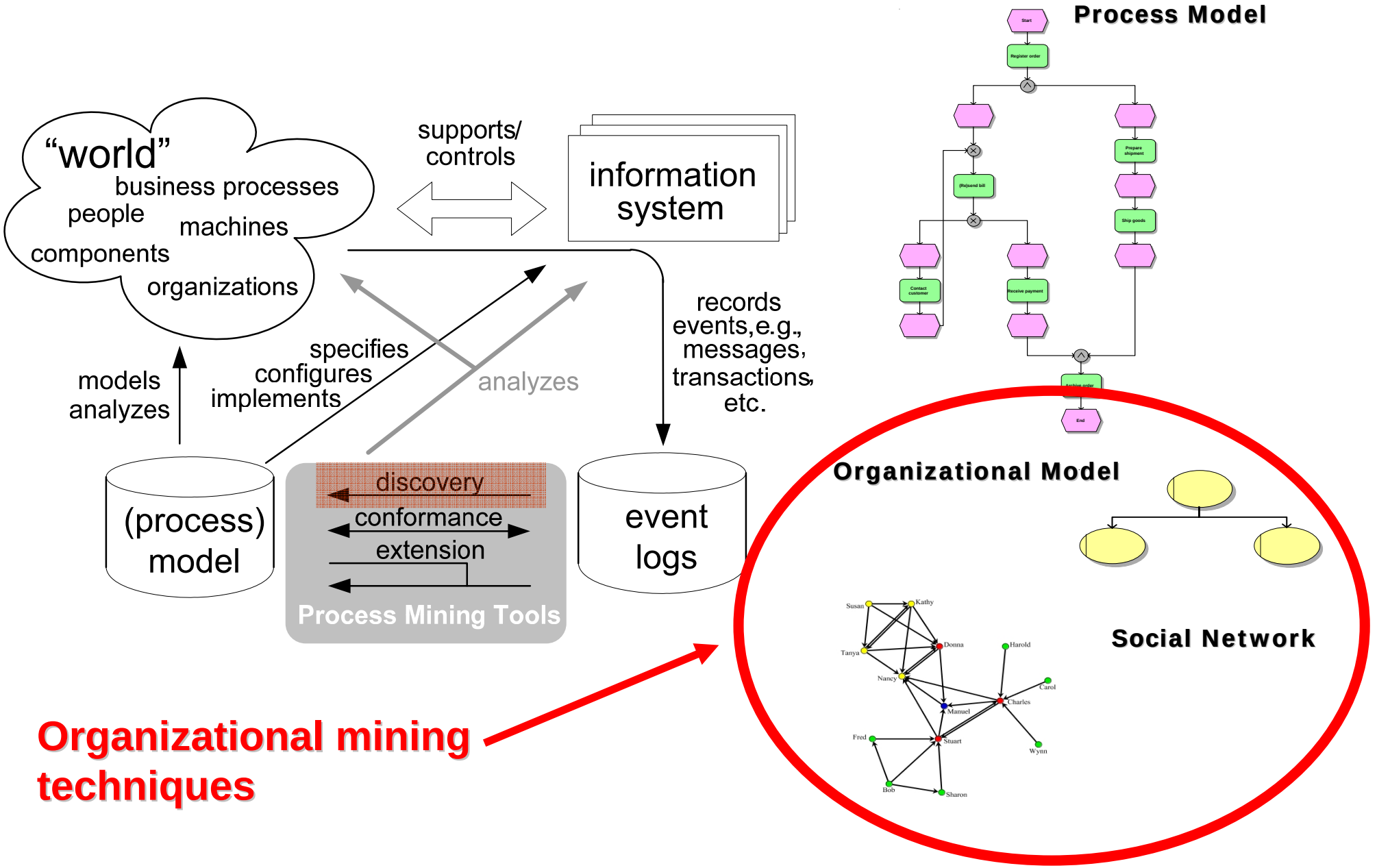
Process Mining

Part III – Beyond control-flow mining

Organizational mining
Discovery of social nets
Extension algorithms

Outline

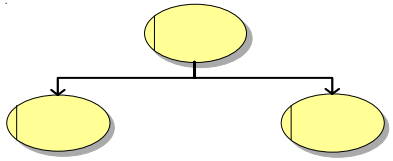
- **Part I – Introduction to Process Mining**
  - Context, motivation and goal
  - General characteristics of the analyzed processes and logs
  - Classification of Process Mining approaches
- **Part II – Workflow discovery**
  - Induction of basic Control Flow graphs
  - Other techniques (α-algorithm, Heuristic Miner, Fuzzy mining)
- **Part III – Beyond control-flow mining**
  - Organizational mining
  - Social net discovery
  - Extension algorithms
- **Part IV – Evaluation and validation of discovered models**
  - Conformance Check
  - Log-based property verification
- **Part V – Clustering-based Process Mining**
  - Discovery of hierarchical process models
  - Discovery of process taxonomies
  - Outlier detection

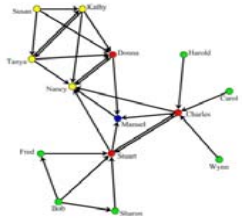


Organizational mining techniques

Organizational Mining Algorithms

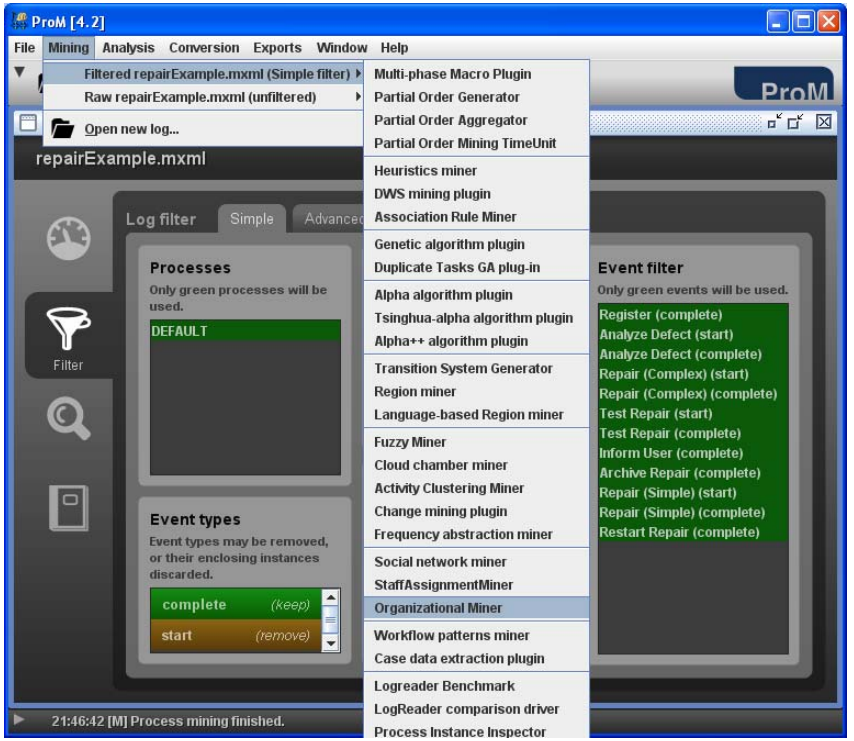
- Objective:
  - Discover the organizational model (i.e., roles, departments,etc.) without prior knowledge about the structure of the organization
  - Aid in understanding and improving social and organizational structures
- Two types of algorithms
  - Organizational Model
    - Mining of roles and teams in organizations
    - ProM Plug-in: Organizational Miner
  - Social Networks
    - Discovery of relationships among originators
    - ProM Plug-ins: *Social Network Miner* and *Analyze Social Network*





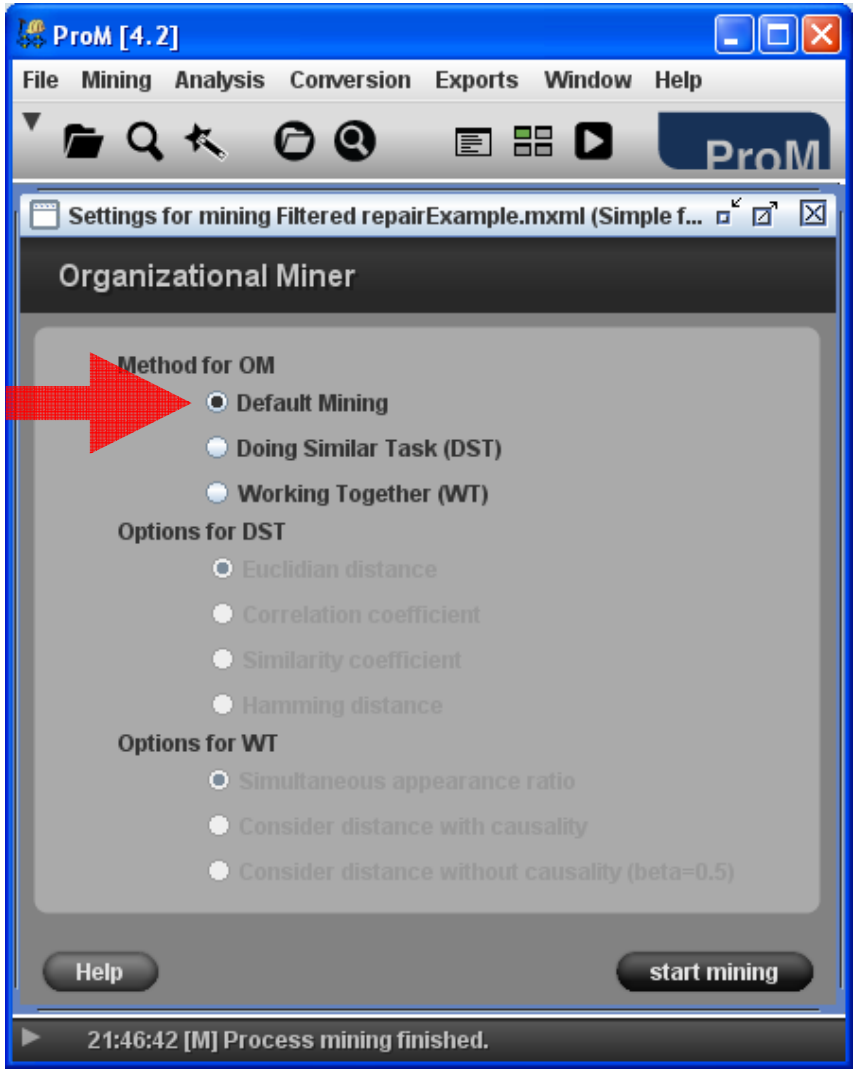
Organizational Miner

- Main idea: Which originators are executing which tasks
- Methods to mine ***roles***
  - Default mining
  - Doing Similar Tasks
- Methods to mine ***teams***
  - Working together



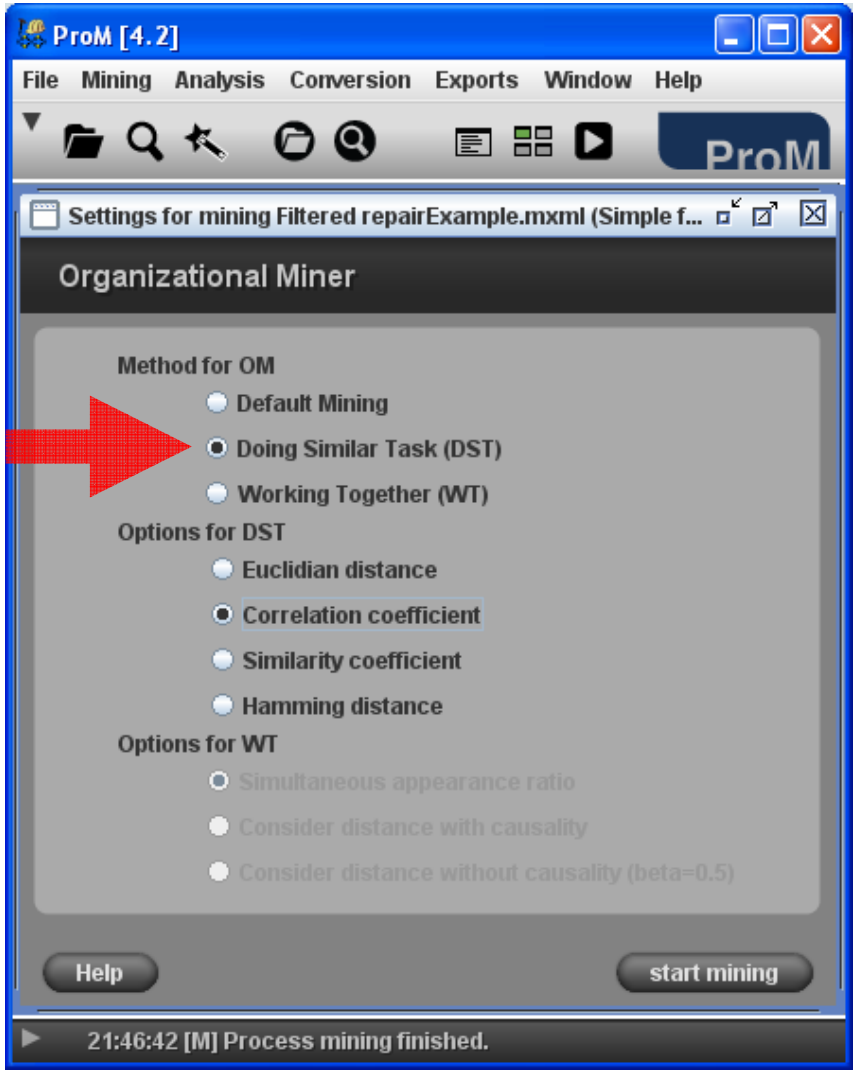
Organizational Miner

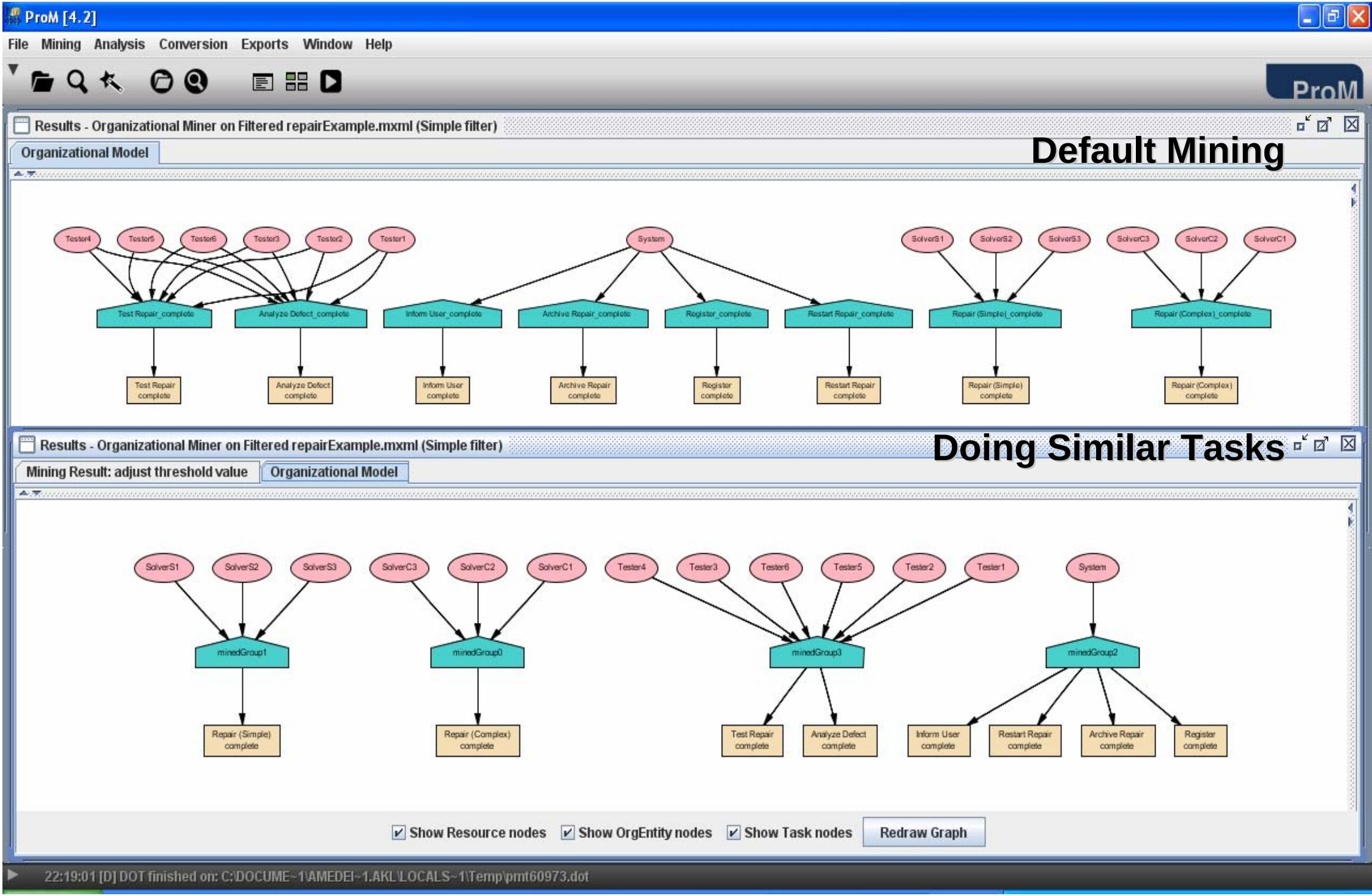
- Main idea: Which performers are executing which tasks
- Methods to mine ***roles***
  - ***Default mining***
  - Doing Similar Tasks
- Methods to mine ***teams***
  - Working together



Organizational Miner

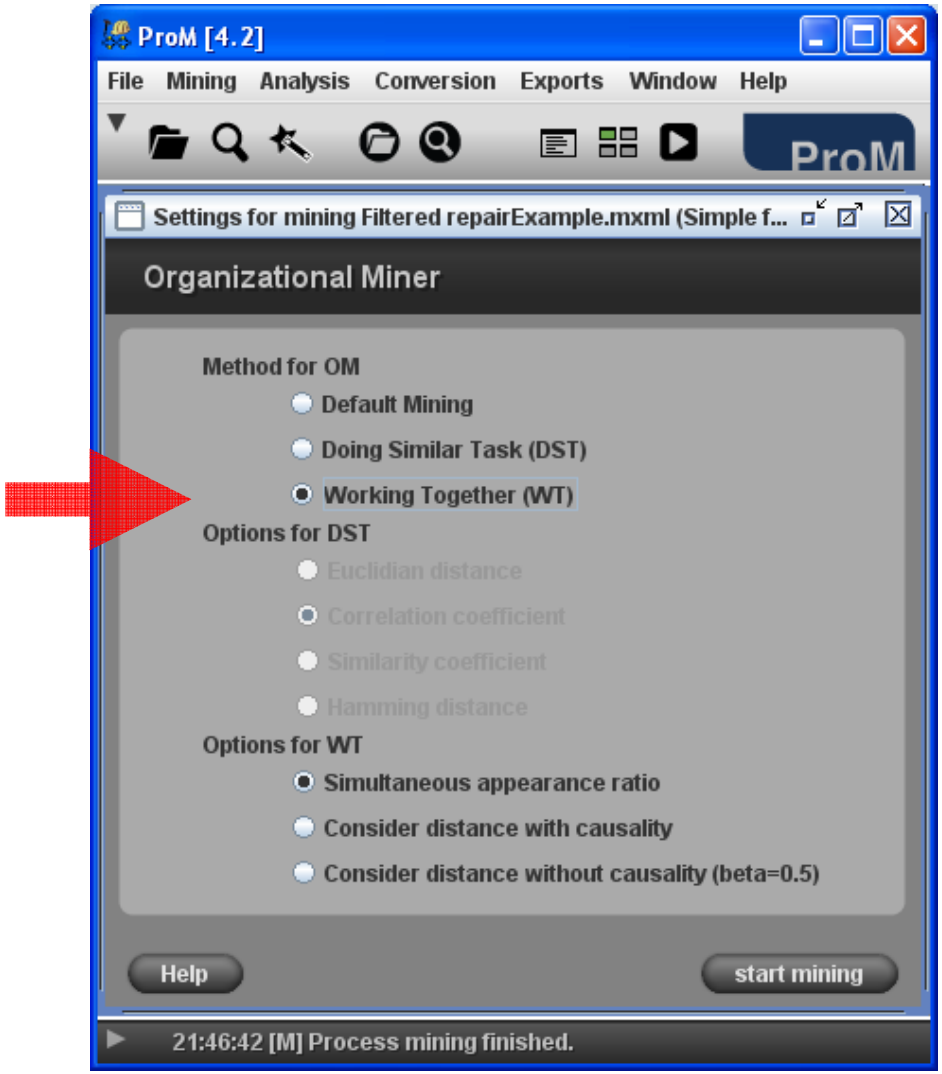
- Main idea: Which performers are executing which tasks
- Methods to mine ***roles***
  - Default mining
  - ***Doing Similar Tasks***
- Methods to mine ***teams***
  - Working together




ProM [4.2]
File Mining Analysis Conversion Exports Window Help
ProM
Results - Organizational Miner on Filtered repairExample.mxml (Simple filter)
Organizational Model
Default Mining
Tester4 Tester5 Tester6 Tester3 Tester2 Tester1
System
SolverS1 SolverS2 SolverS3
SolverC3 SolverC2 SolverC1
Test Repair_complete
Analyze Defect_complete
Inform User_complete
Archive Repair_complete
Register_complete
Restart Repair_complete
Repair (Simple)_complete
Repair (Complex)_complete
Test Repair complete
Analyze Defect complete
Inform User complete
Archive Repair complete
Register complete
Restart Repair complete
Repair (Simple) complete
Repair (Complex) complete
Results - Organizational Miner on Filtered repairExample.mxml (Simple filter)
Doing Similar Tasks
Mining Result: adjust threshold value
Organizational Model
SolverS1 SolverS2 SolverS3
SolverC3 SolverC2 SolverC1
Tester4 Tester3 Tester6 Tester5 Tester2 Tester1
System
minedGroup1
minedGroup0
minedGroup3
minedGroup2
Repair (Simple) complete
Repair (Complex) complete
Test Repair complete
Analyze Defect complete
Inform User complete
Restart Repair complete
Archive Repair complete
Register complete
Show Resource nodes
Show OrgEntity nodes
Show Task nodes
Redraw Graph
22:19:01 [D] DOT finished on: C:\DOCUME~1\AMEDEI~1.AKL\LOCALS~1\Temp\pmt60973.dot

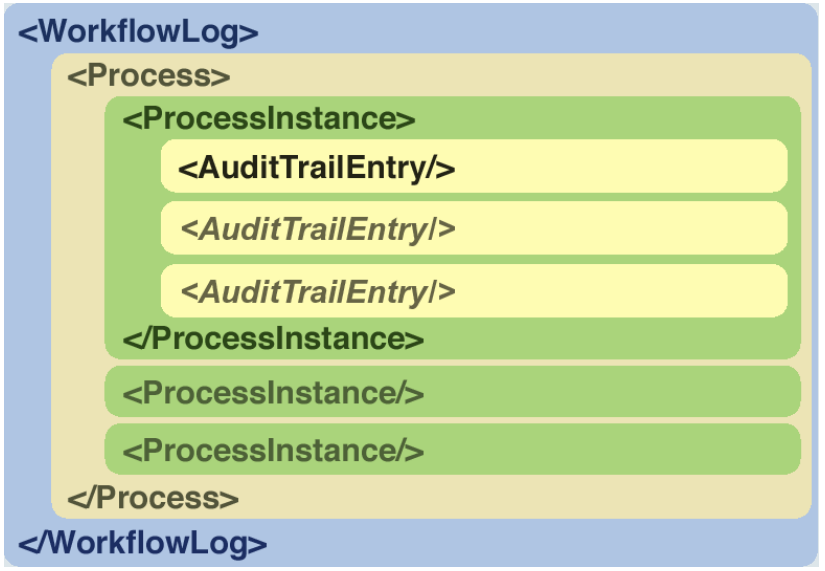
Organizational Miner

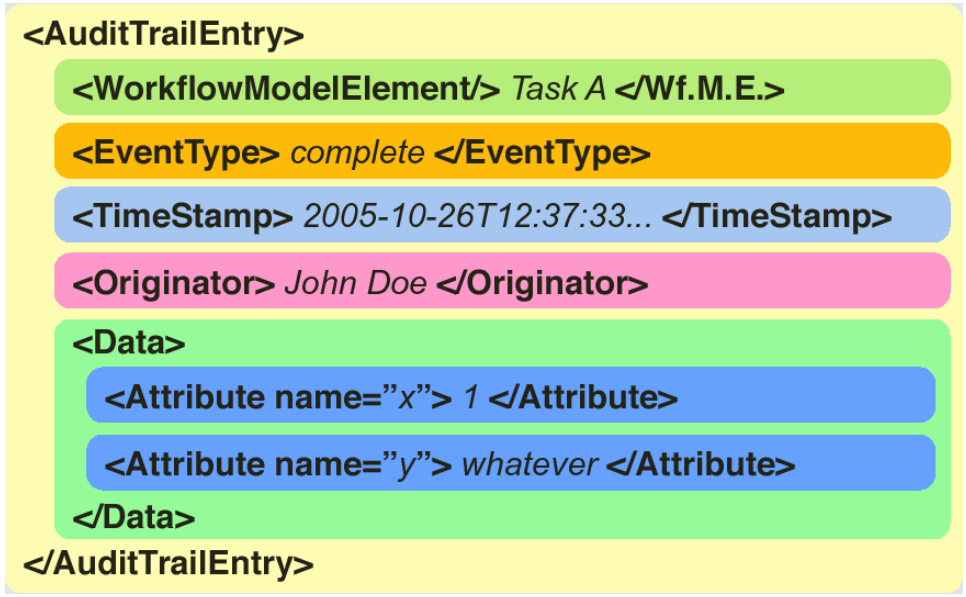
- Main idea: Which performers are executing which tasks
- Methods to mine ***roles***
  - Default mining
  - Doing Similar Tasks
- Methods to mine ***teams***
  - ***Working together***



Organizational Miner

Why is the notion of process instances necessary to mine teams but unnecessary to mine roles?



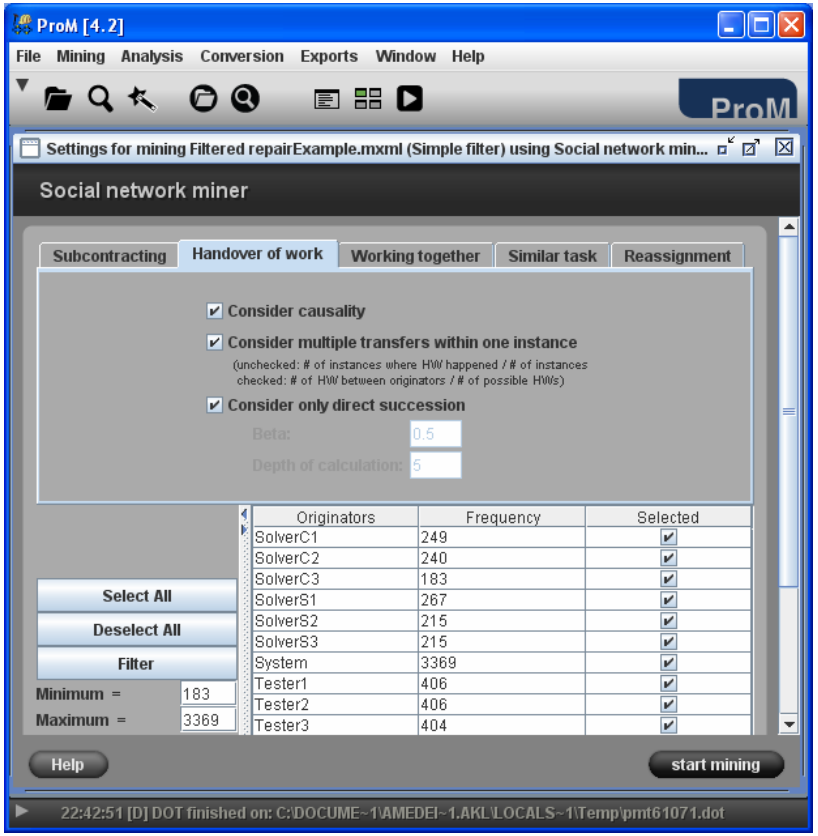


Outline

- **Part I – Introduction to Process Mining**
  - Context, motivation and goal
  - General characteristics of the analyzed processes and logs
  - Classification of Process Mining approaches
- **Part II – Workflow discovery**
  - Induction of basic Control Flow graphs
  - Other techniques (α-algorithm, Heuristic Miner, Fuzzy mining)
- **Part III – Beyond control-flow mining**
  - Organizational mining
  - Social net discovery
  - Extension algorithms
- **Part IV – Evaluation and validation of discovered models**
  - Conformance Check
  - Log-based property verification
- **Part V – Clustering-based Process Mining**
  - Discovery of hierarchical process models
  - Discovery of process taxonomies
  - Outlier detection

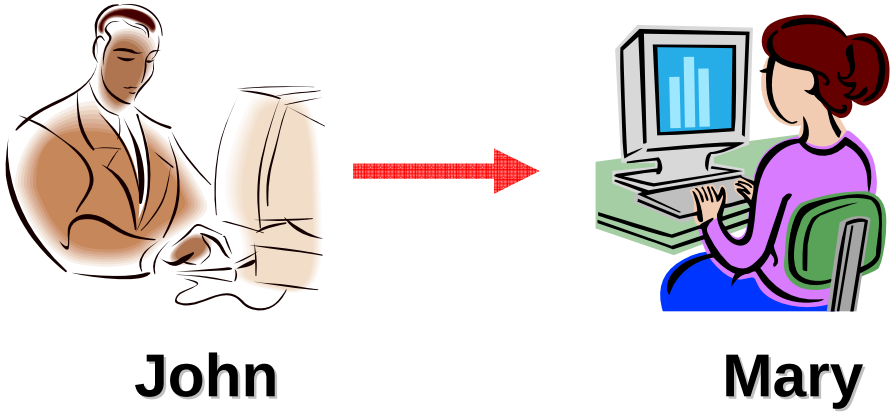
Social Network Miner

- Aim:
  - Monitor how individual process instances are routed between originators
- Metrics
  - Handover of work
  - Subcontracting
  - Reassignment
  - Working together
  - Similar task



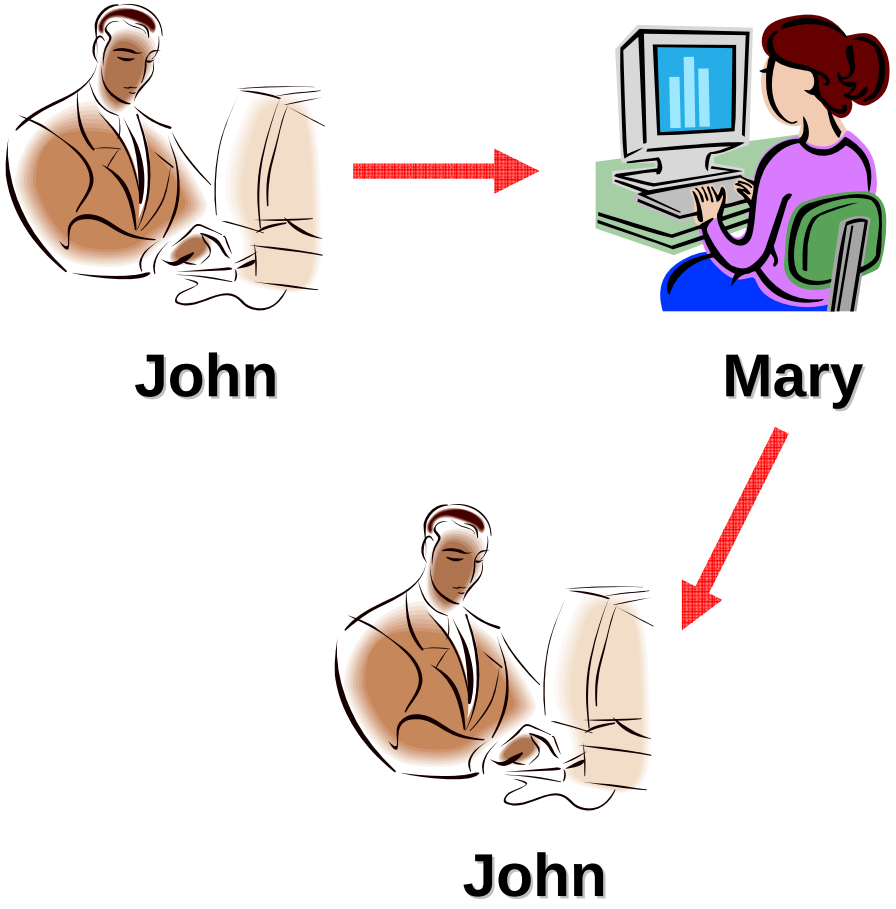
Social Network Miner

- Aim: Monitor how individual process instances are routed between originators
- Metrics
  - ***Handover of work***
  - Subcontracting
  - Reassignment
  - Working together
  - Similar task



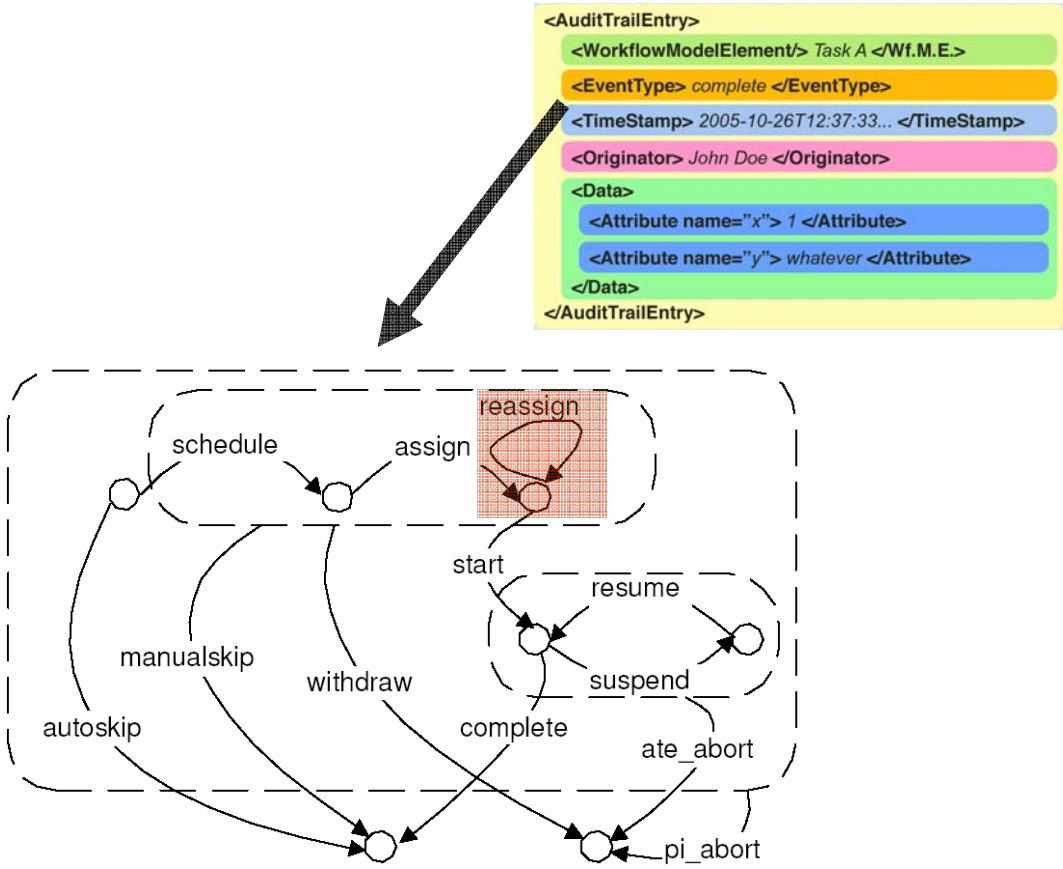
Social Network Miner

- Aim: Monitor how individual process instances are routed between originators
- Metrics
  - Handover of work
  - ***Subcontracting***
  - Reassignment
  - Working together
  - Similar task



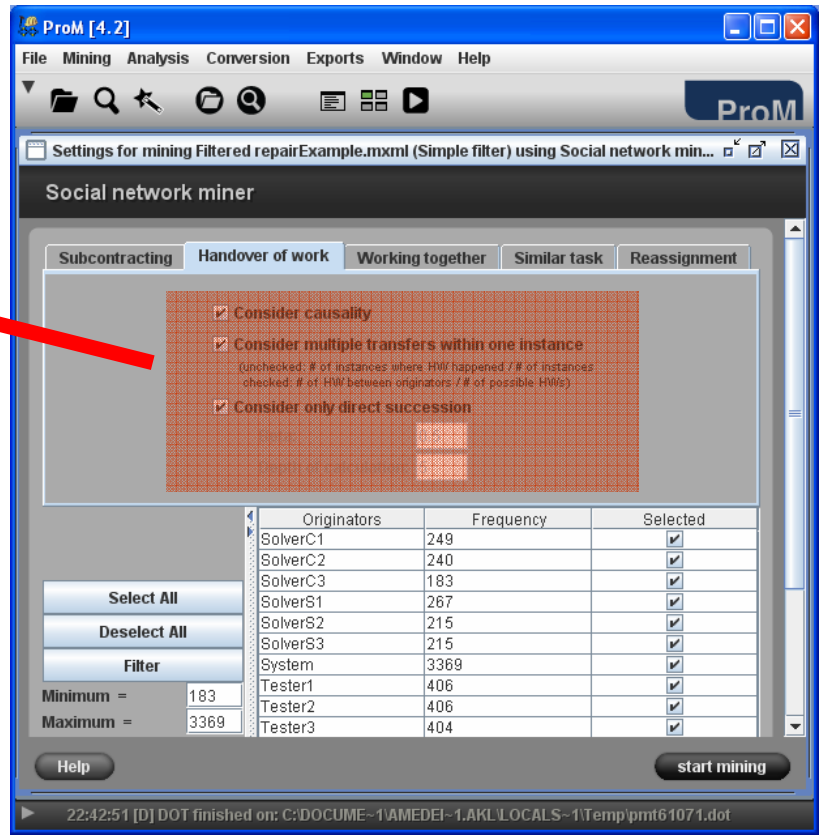
Social Network Miner

- Aim: Monitor how individual process instances are routed between originators
- Metrics
  - Handover of work
  - Subcontracting
  - ***Reassignment***
  - Working together
  - Similar task



Social Network Miner

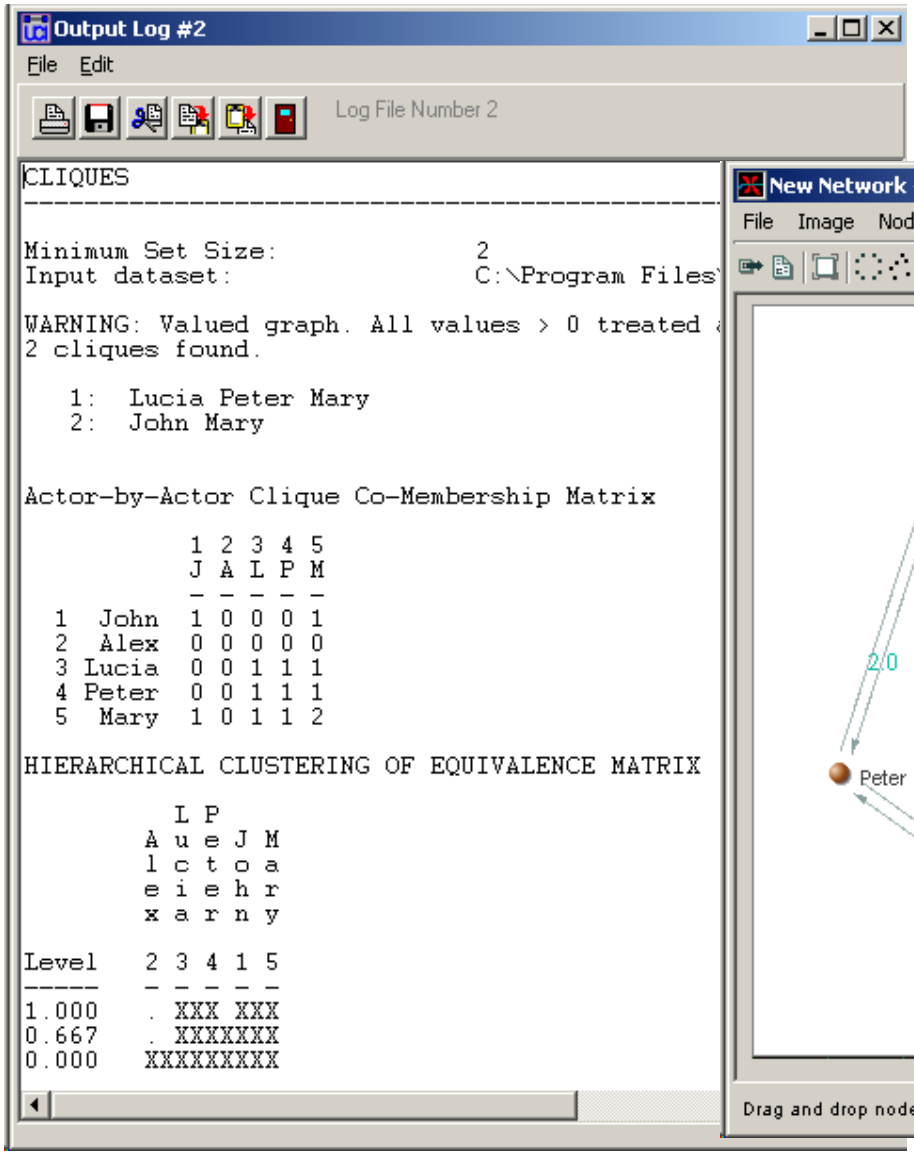
- Aim: Monitor how individual process instances are routed between originators
- Metrics
  - Handover of work
  - Subcontracting
  - Reassignment
  - Working together
  - Similar task

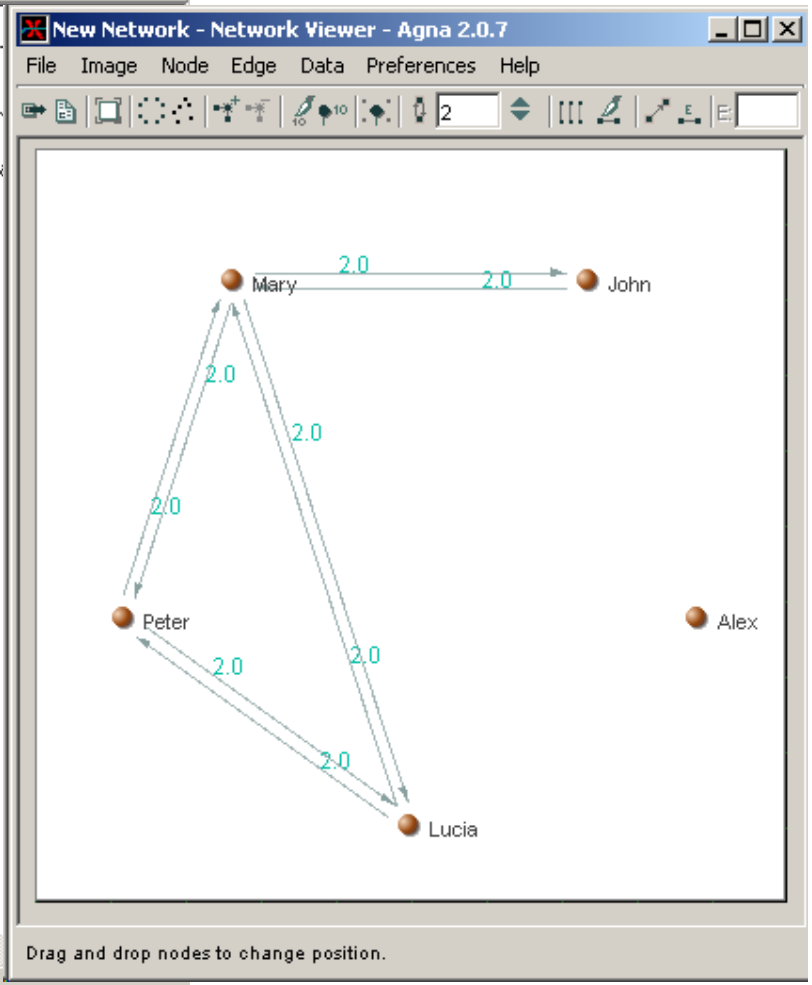


Based on ordering relations derived from a log!

Social Network Miner: Example

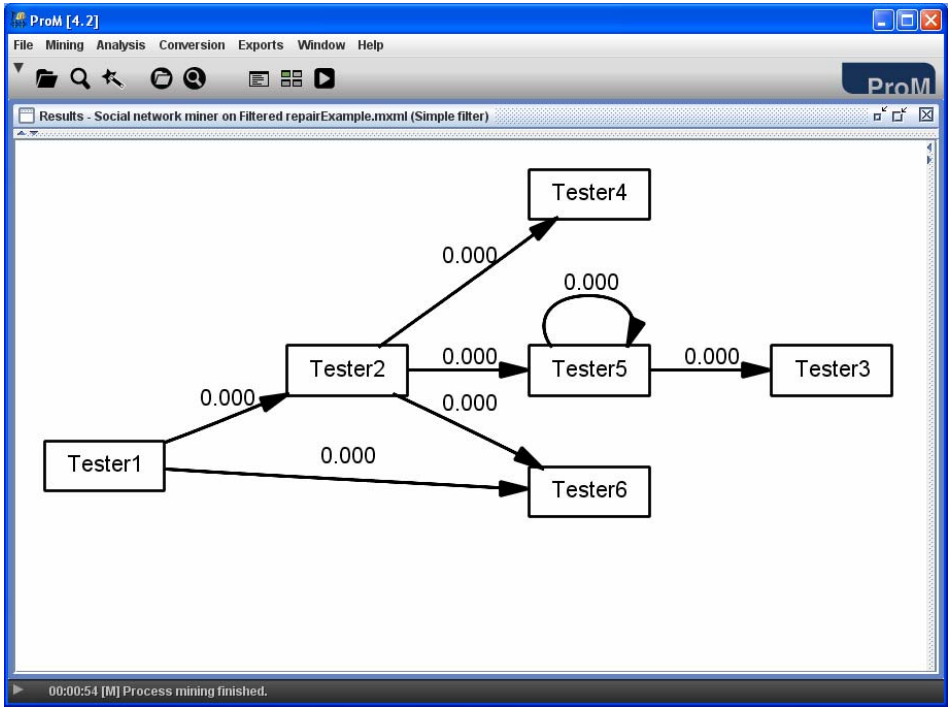
	John	Alex	Lucia	Peter	Mary
John	0	0	0	0	2
Alex	0	0	0	0	0
Lucia	0	0	0	2	2
Peter	0	0	2	0	2
Mary	2	0	2	2	0





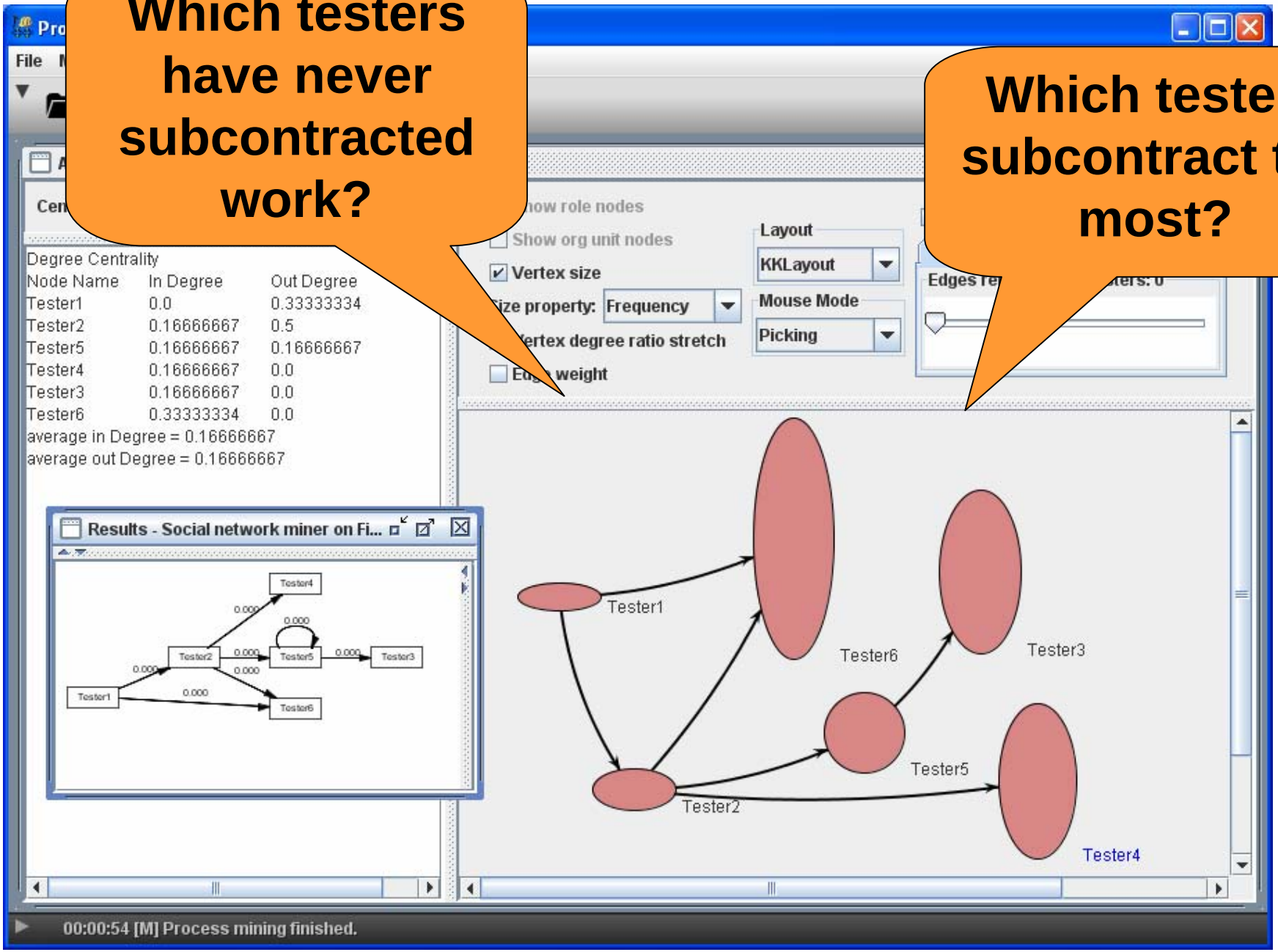
Plugin *Analyze Social Network*

- Better graphical view for the results of the Social Network Miner
- Includes different metrics to measure centrality of nodes
- Example: subcontracting



Which testers have never subcontracted work?

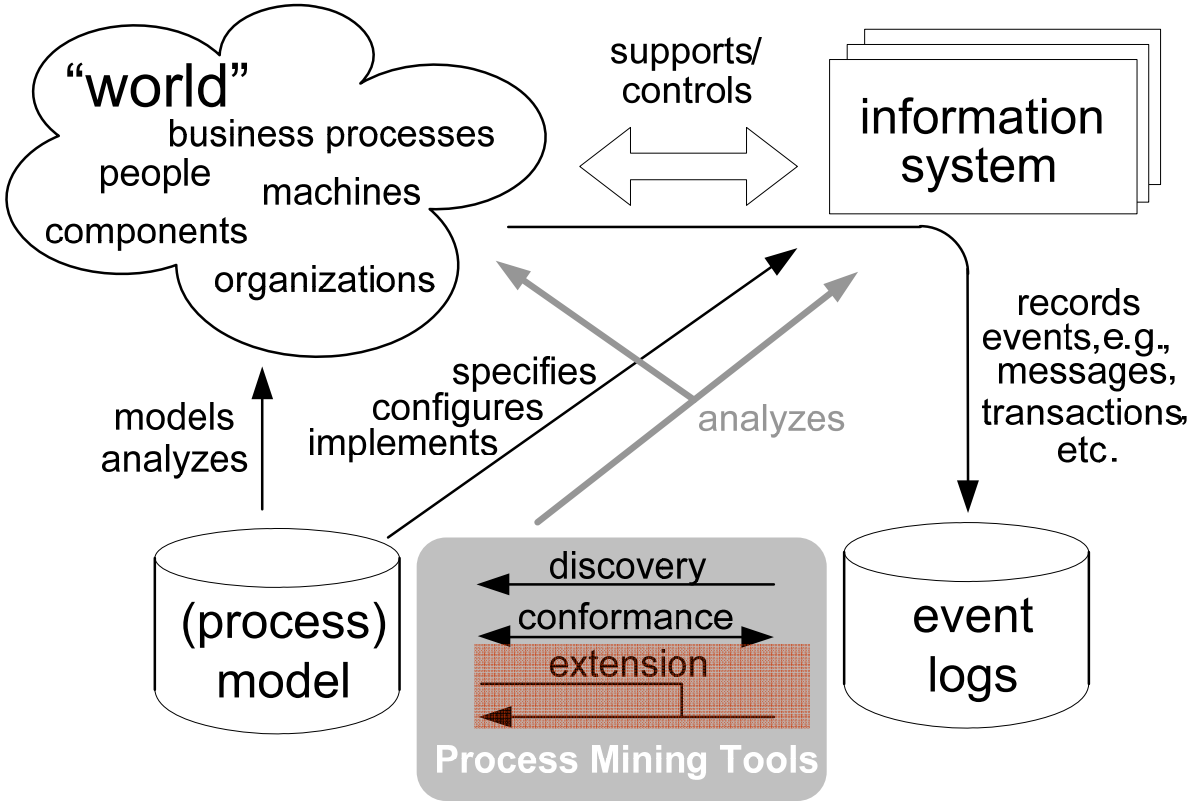
Which testers subcontract the most?



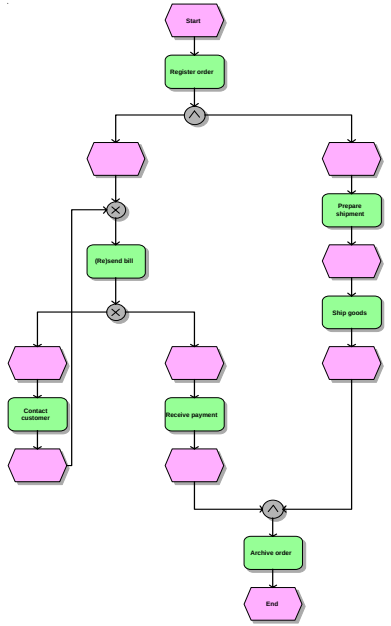
Outline

- Part I – Introduction to Process Mining
  - Context, motivation and goal
  - General characteristics of the analyzed processes and logs
  - Classification of Process Mining approaches
- Part II – Workflow discovery
  - Induction of basic Control Flow graphs
  - Other techniques (α-algorithm, Heuristic Miner, Fuzzy mining)
- **Part III – Beyond control-flow mining**
  - Organizational mining
  - Social net discovery
  - **Extension algorithms**
- **Part IV – Evaluation and validation of discovered models**
  - Conformance Check
  - Log-based property verification
- **Part V – Clustering-based Process Mining**
  - Discovery of hierarchical process models
  - Discovery of process taxonomies
  - Outlier detection in a process mining setting

Extension techniques



Bottlenecks / Business Rules Process Model



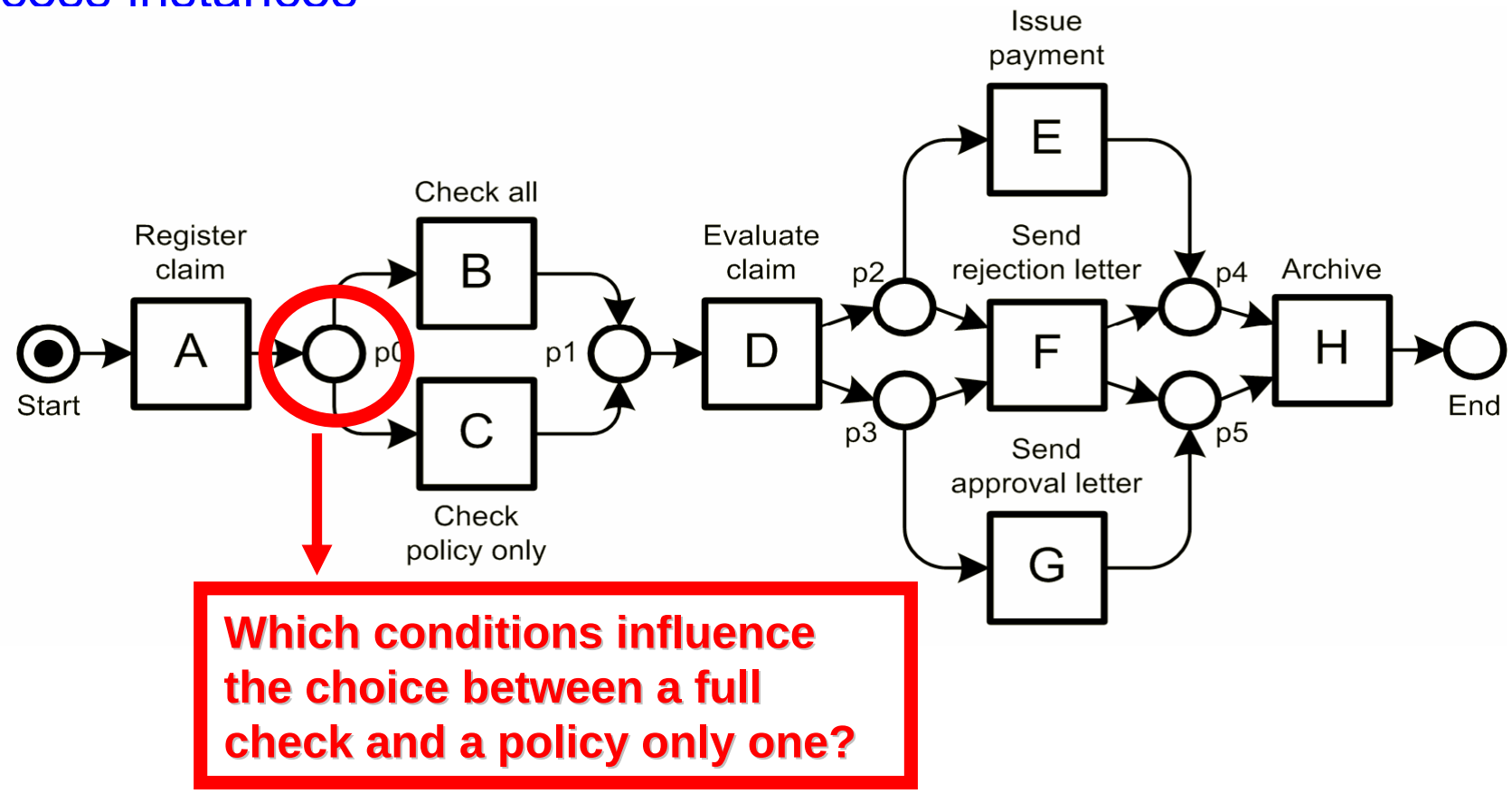
Performance Analysis

Enhance existing models with information discovered from logs

- The *Decision Point Analysis* plug-in can discover the "business rules" for the moments of choice in a process model
- The *Performance Analysis with Petri Nets* plug-in provides various KPIs w.r.t. the execution of processes

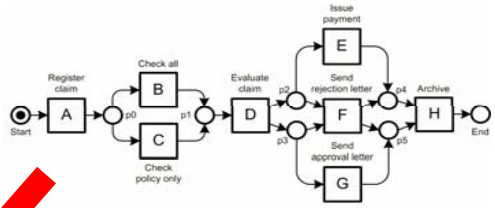
Decision Point Analysis: Main Idea

- Detection of data dependencies that affect the rounting the routing of process instances

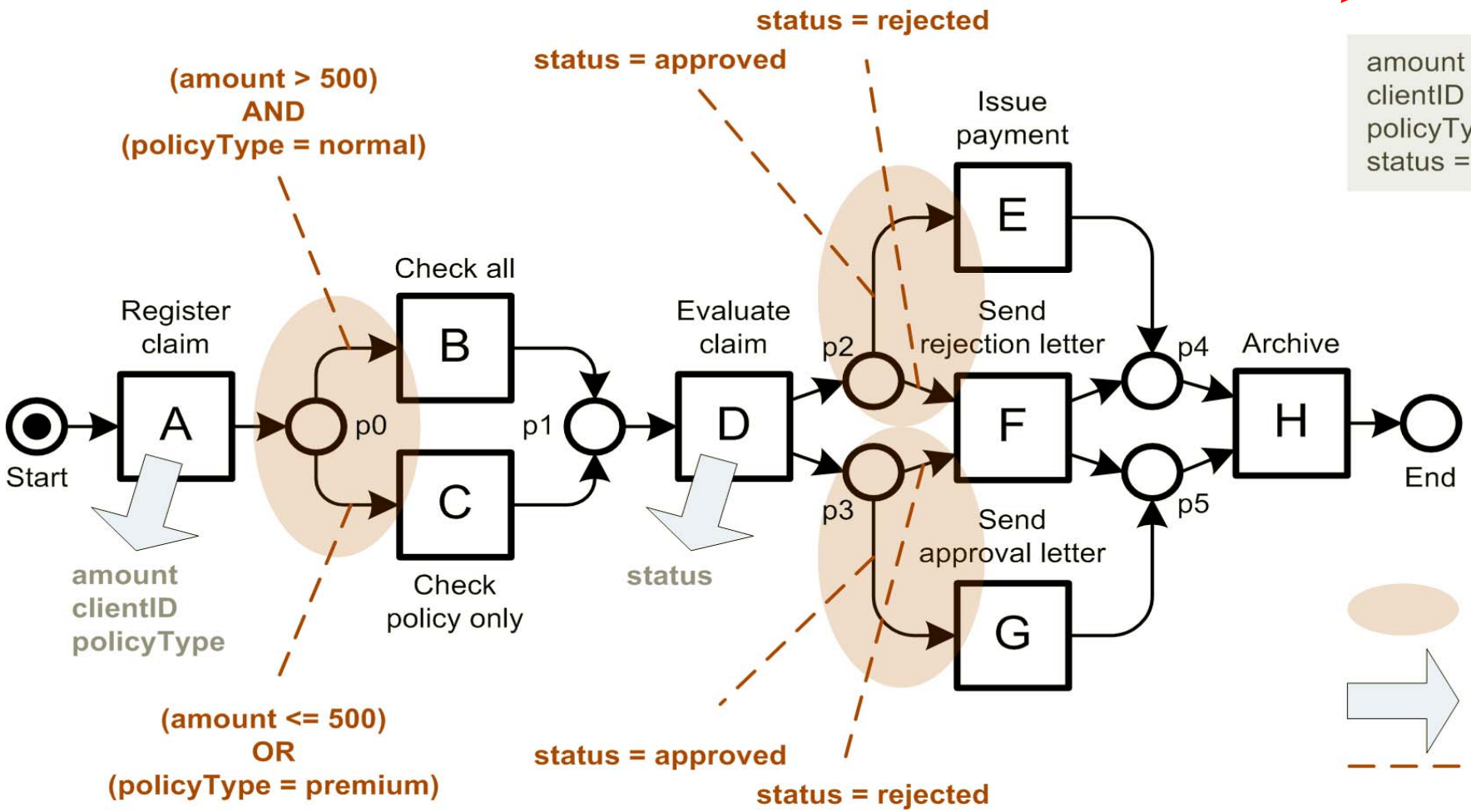


- Motivations
  - Make tacit knowledge explicit
  - Better understand the process model

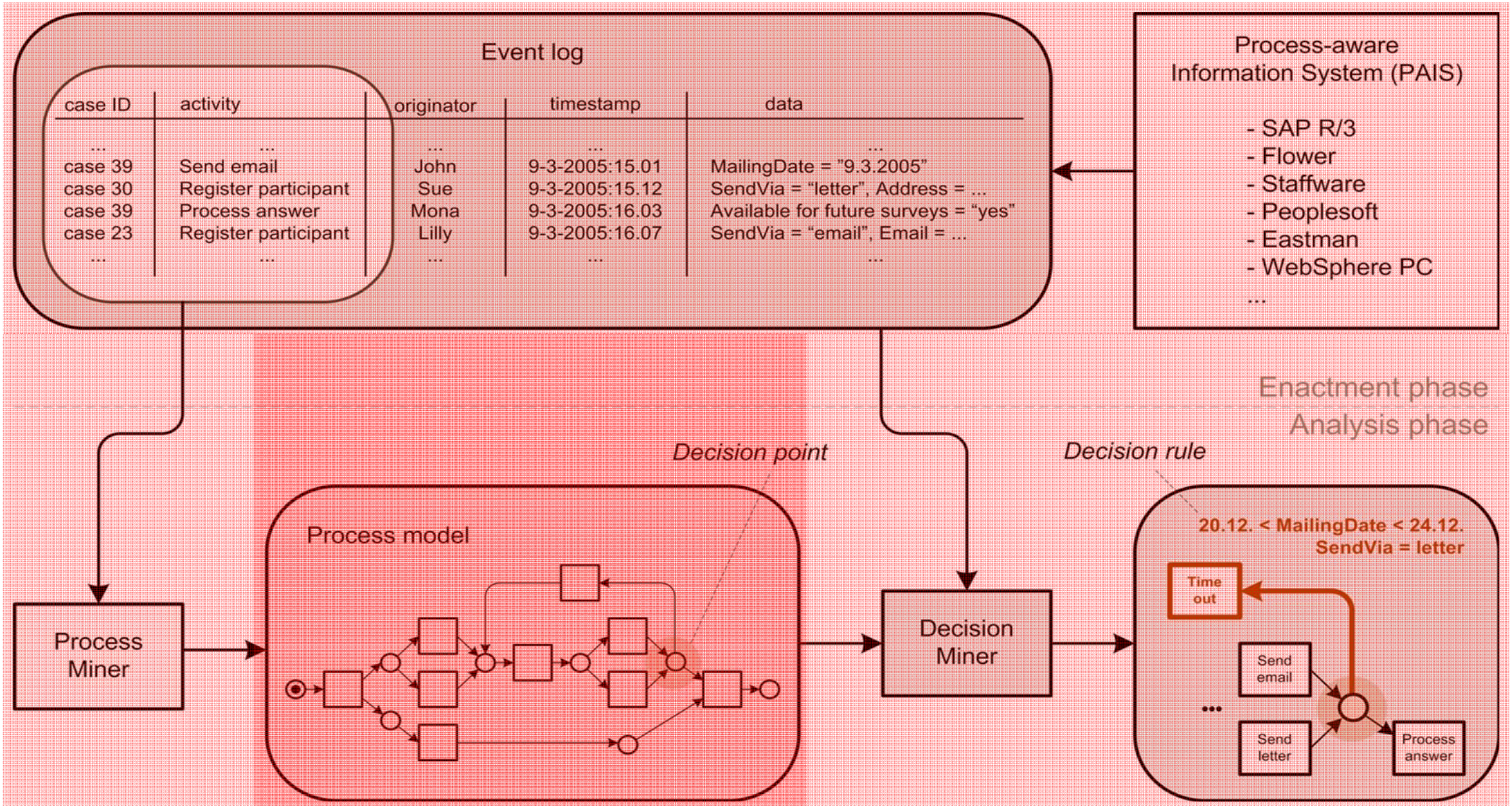
Decision Point Analysis: Motivation



amount = R
clientID = String
policyType = normal | premium
status = approved | rejected



Decision Point Analysis: Approach



Decision Point Analysis

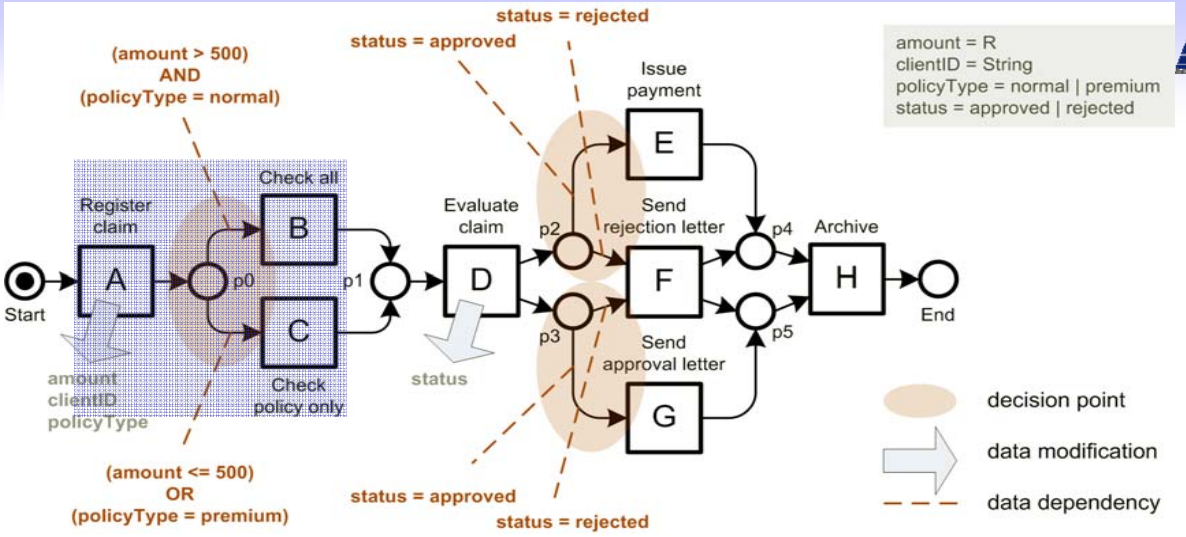
1. Read a log + model
2. Identify the decision points in a model
3. Find out which alternative branch has been taken for a given process instance and decision point
4. Discover the rules for each decision point
5. Return the enhanced model with the discovered rules

Decision Point Analysis

1. Read a log + model
2. Identify
3. Find out en taken for a given proces and decision point
4. Discover the rules for each decision point
5. Return the enhanced model with the discovered rules

Which elements are the classes and which are the attributes?

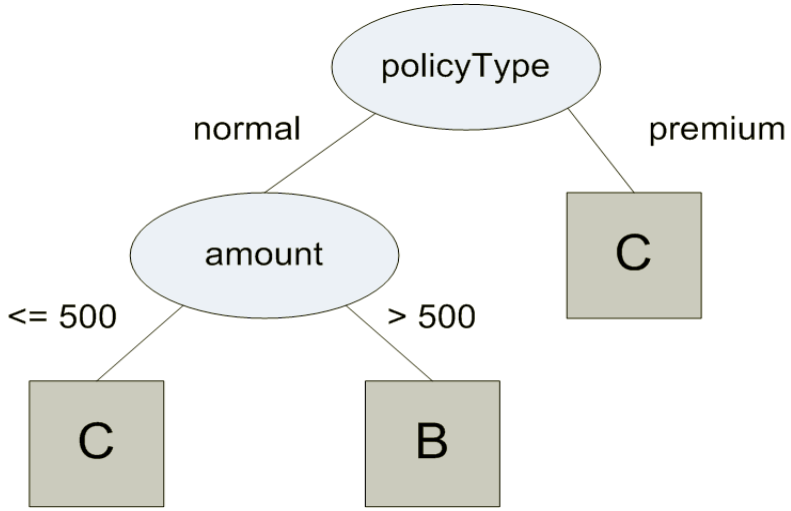
Step 4



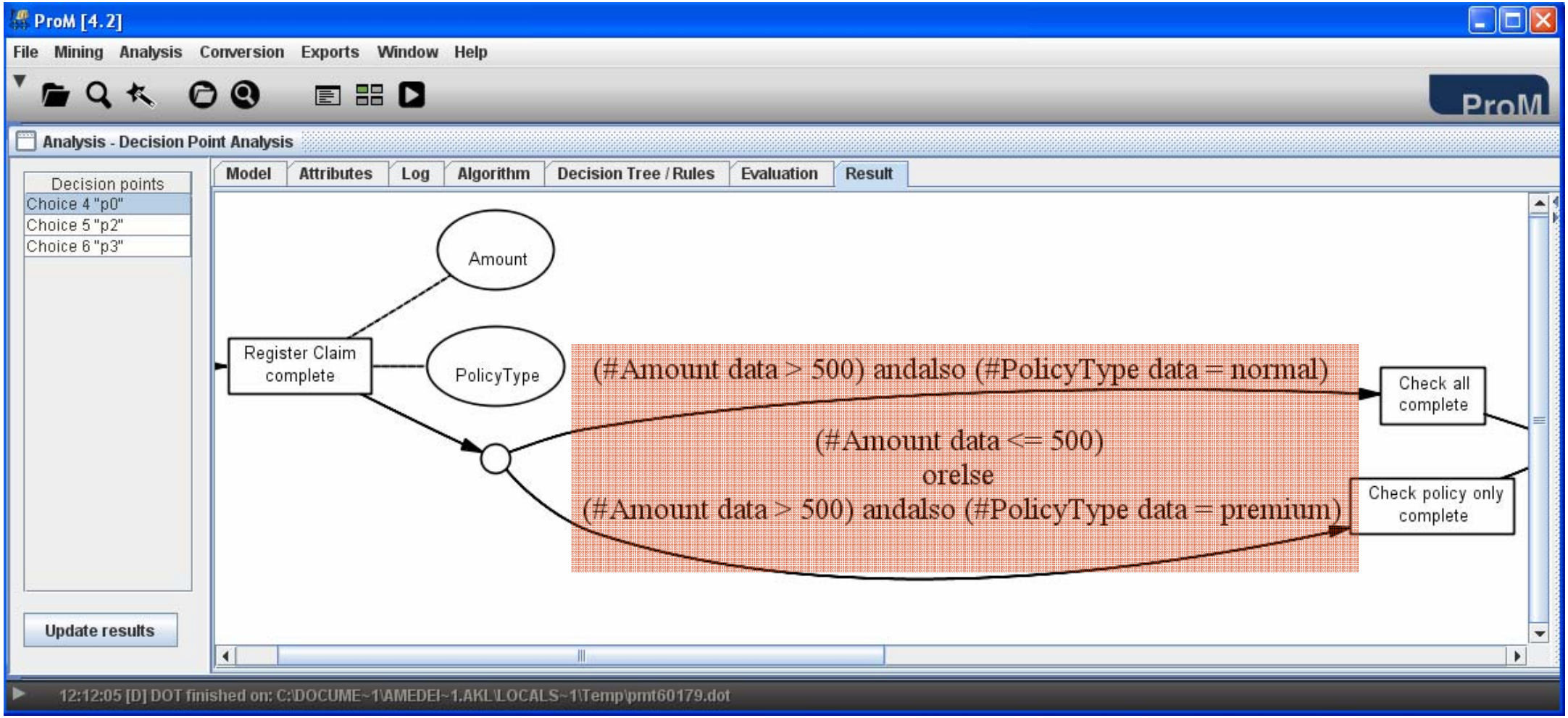
Training examples for decision point "p0"

amount	clientID	policyType	class
1000	C567894938	premium	C
700	C938609223	normal	B
550	C135697567	normal	B
500	C568120443	normal	C
50	C493823084	normal	C
200	C945675110	premium	C

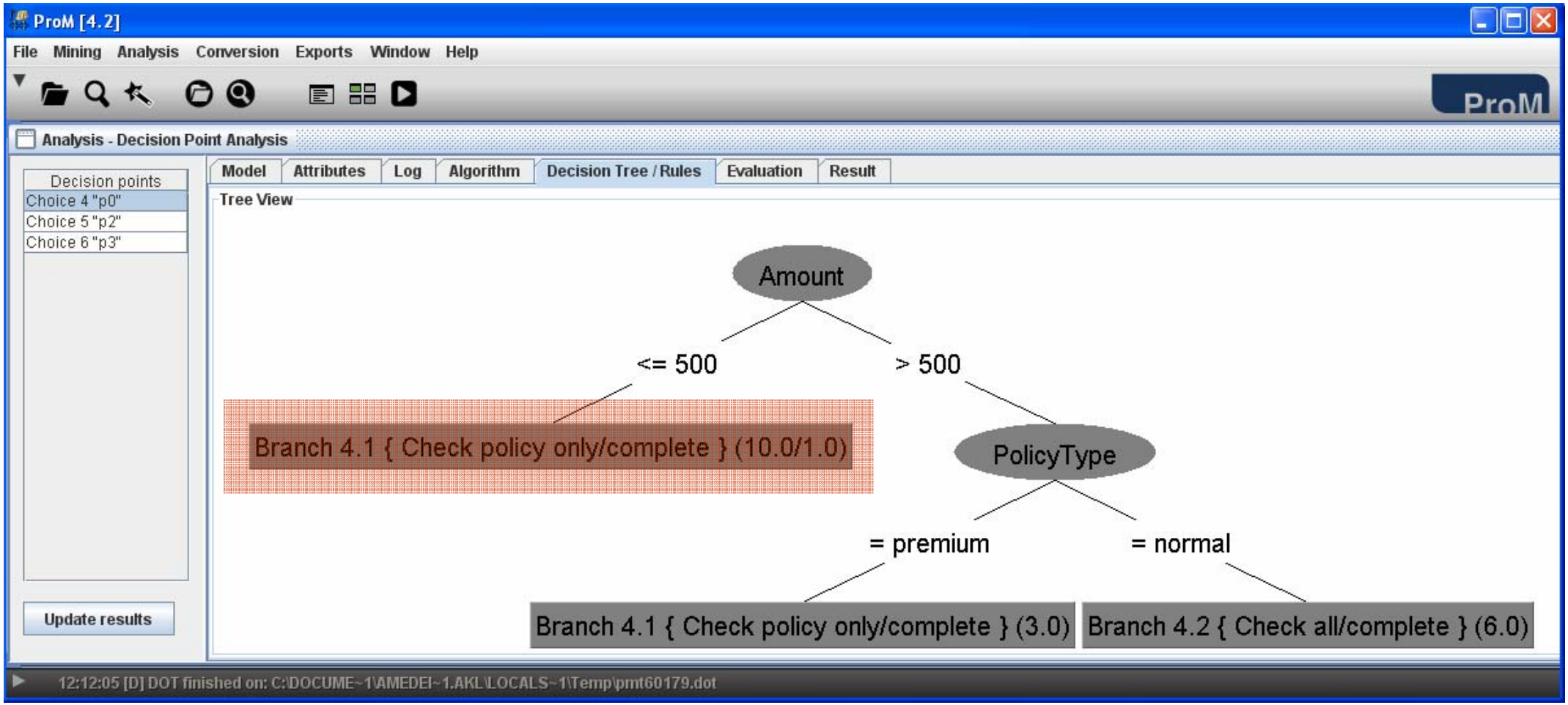
Discovered decision tree for point "p0"



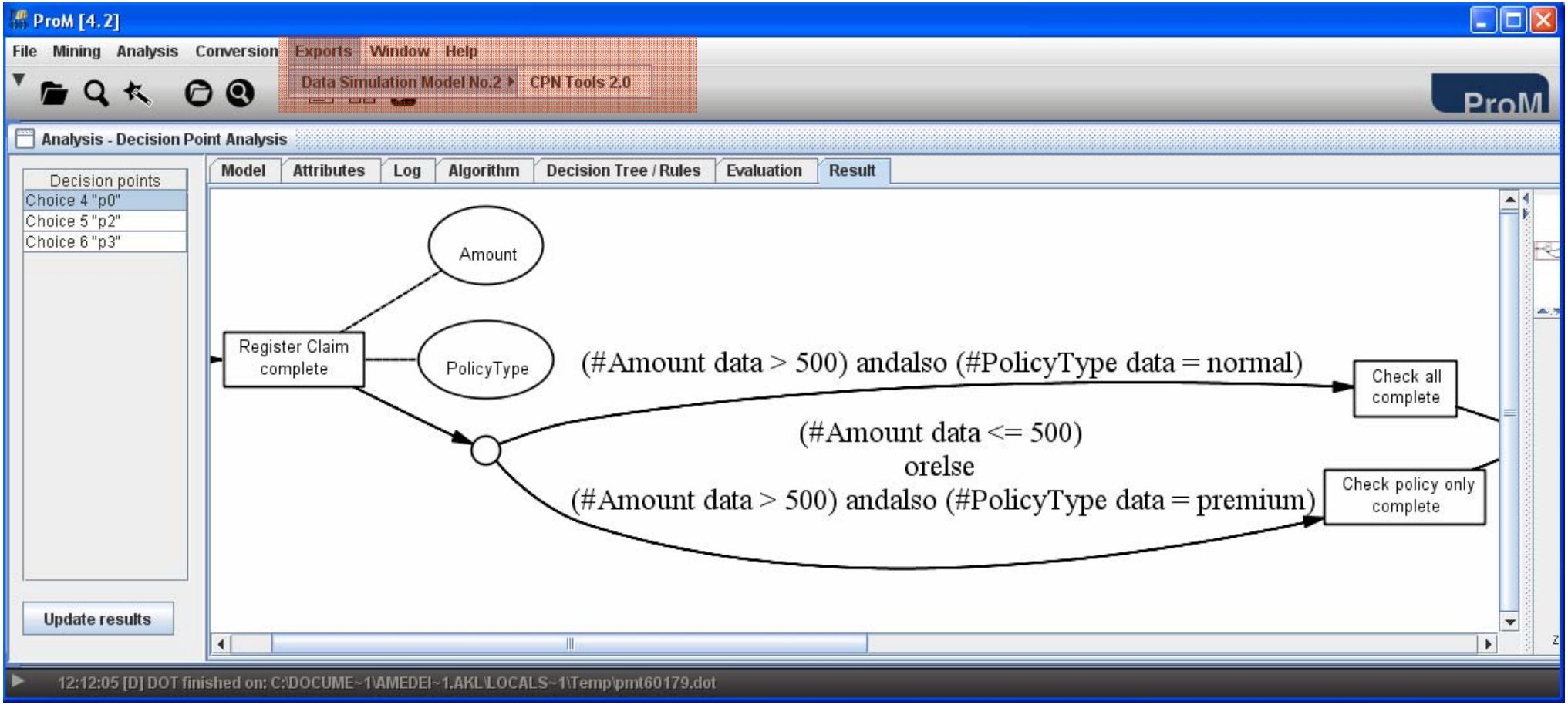
Decision Point Analysis: Example in ProM



Decision Point Analysis: Example in ProM



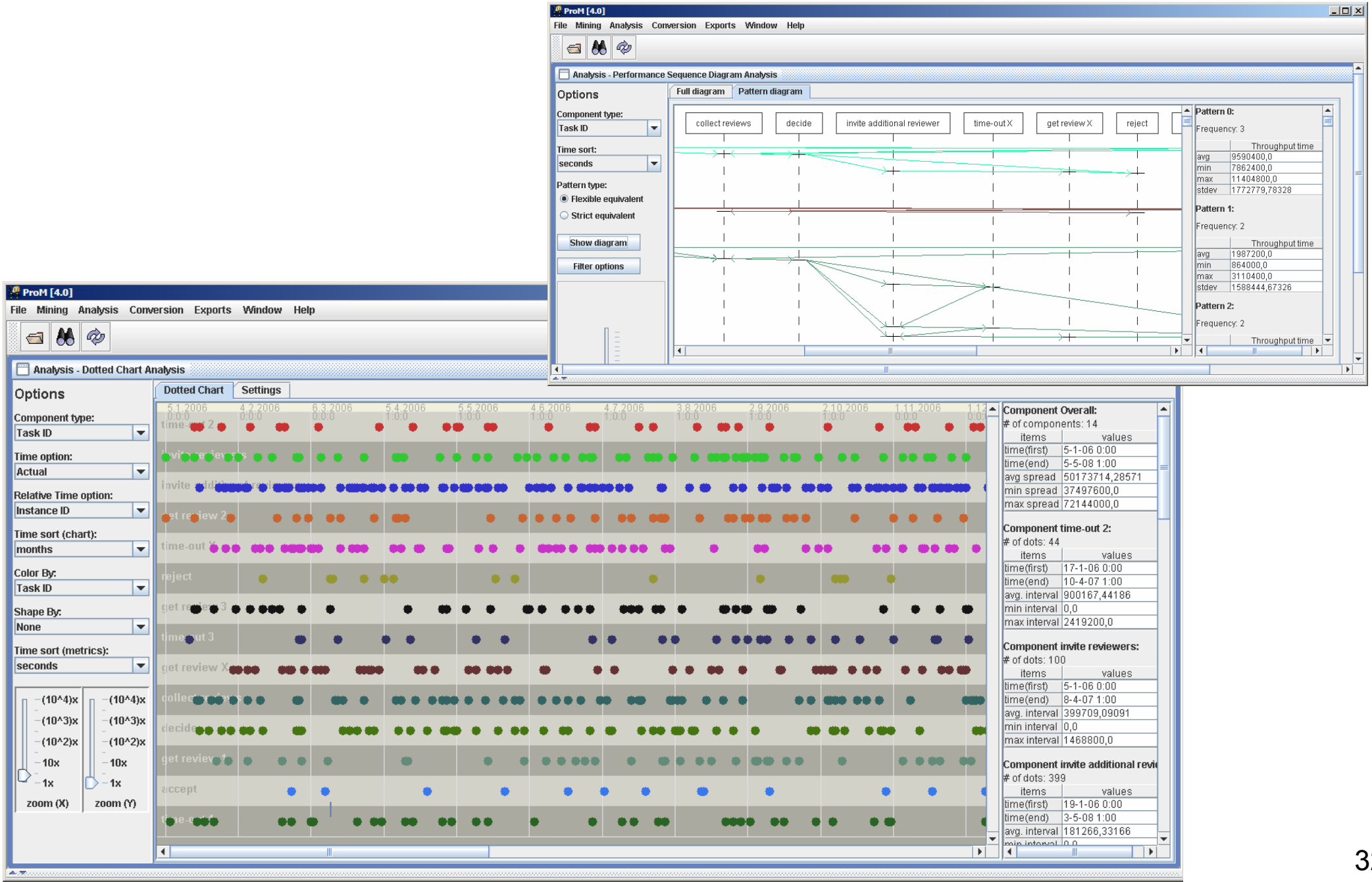
Decision Point Analysis



Extension techniques

- Decision Miner
- Performance Analysis

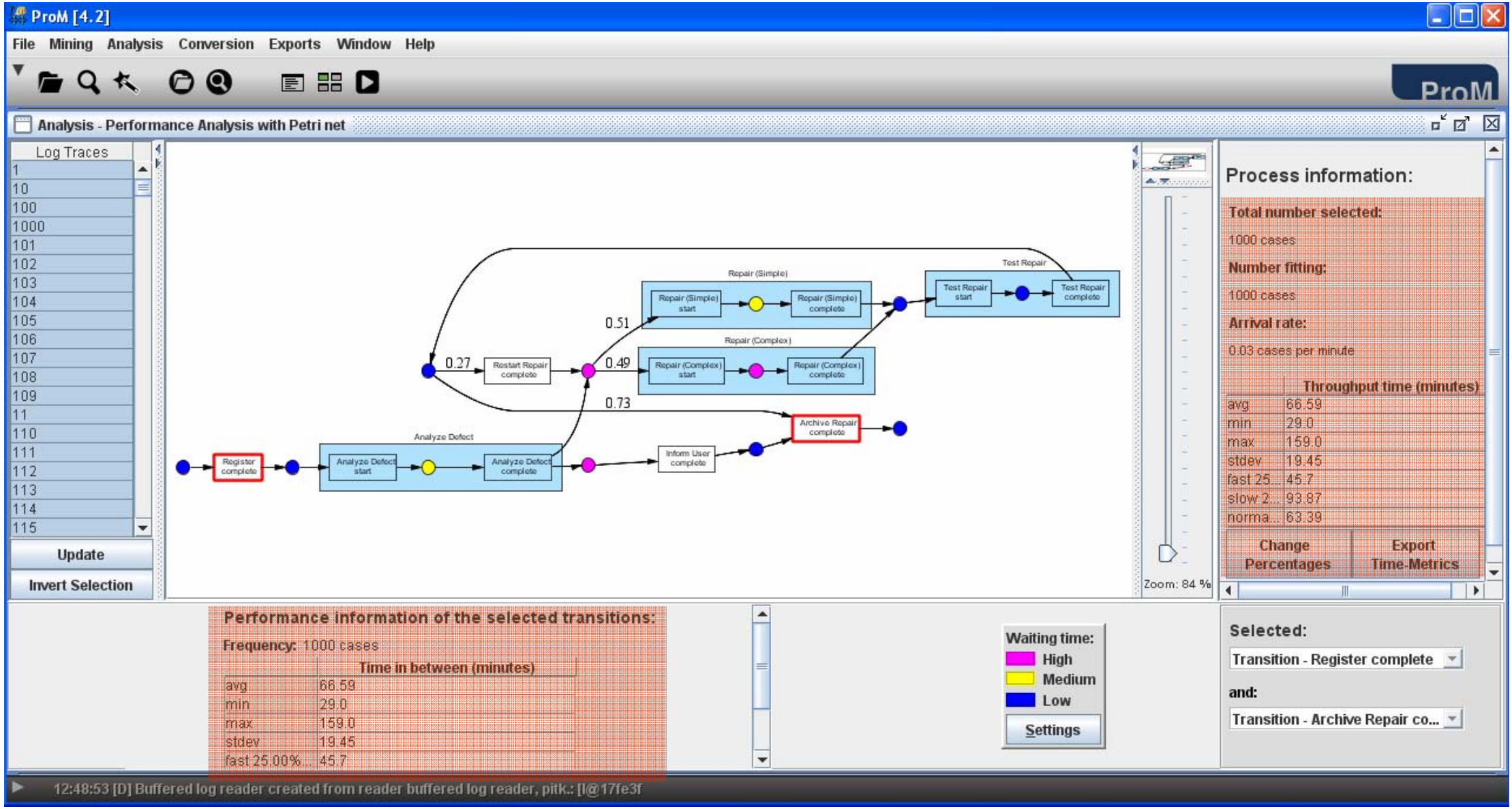
Performance analysis: pattern visualization



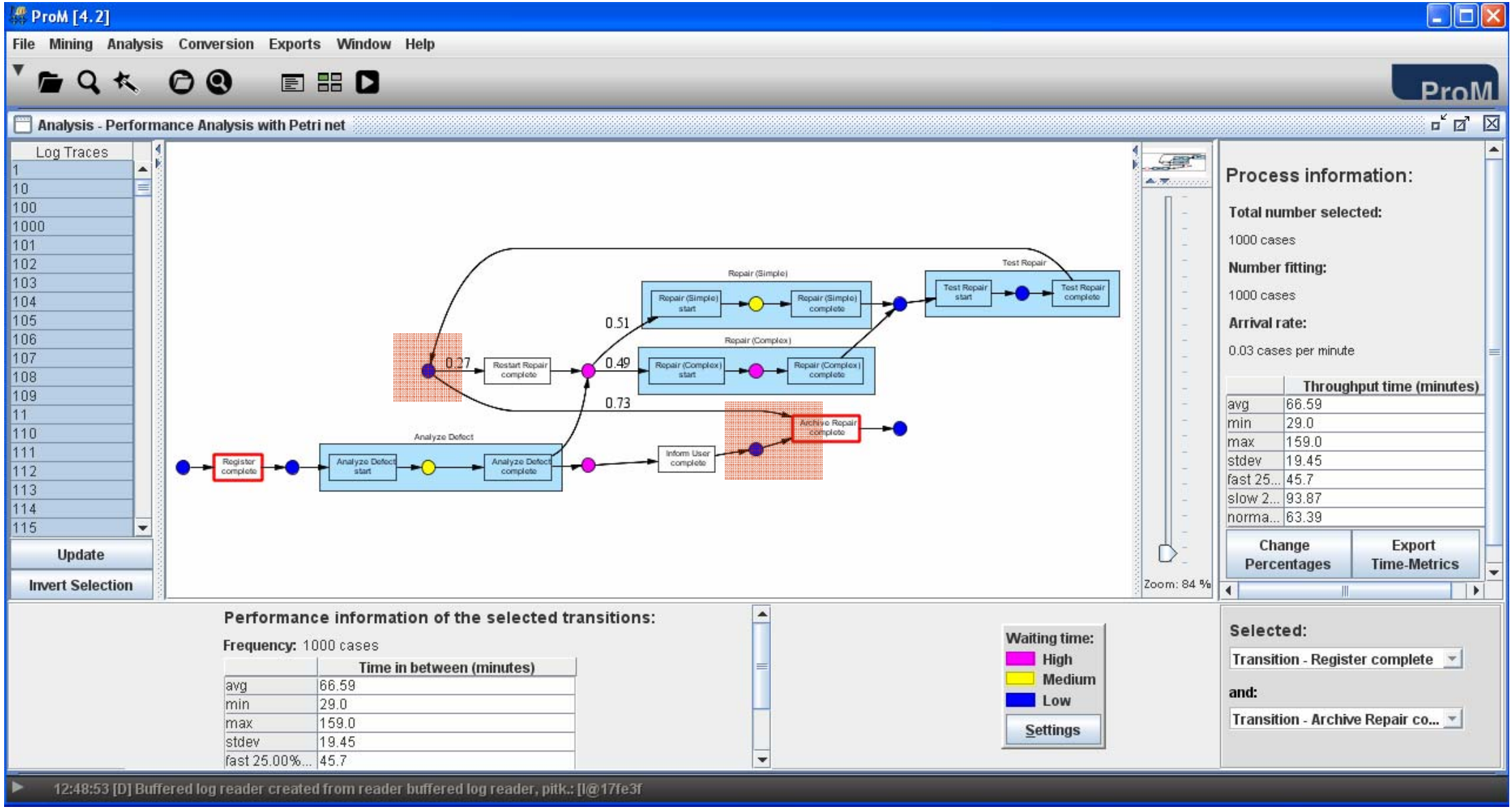
Performance Analysis with Petri Nets

- Motivation
  - Provide different Key Performance Indicators (KPIs) relating to the execution of processes
- Main idea
  - Replay the log in a model and detect
    - Bottlenecks
    - Throughput times
    - Execution times
    - Waiting times
    - Synchronization times
    - Path probabilities etc

Bottlenecks – Throughput Times

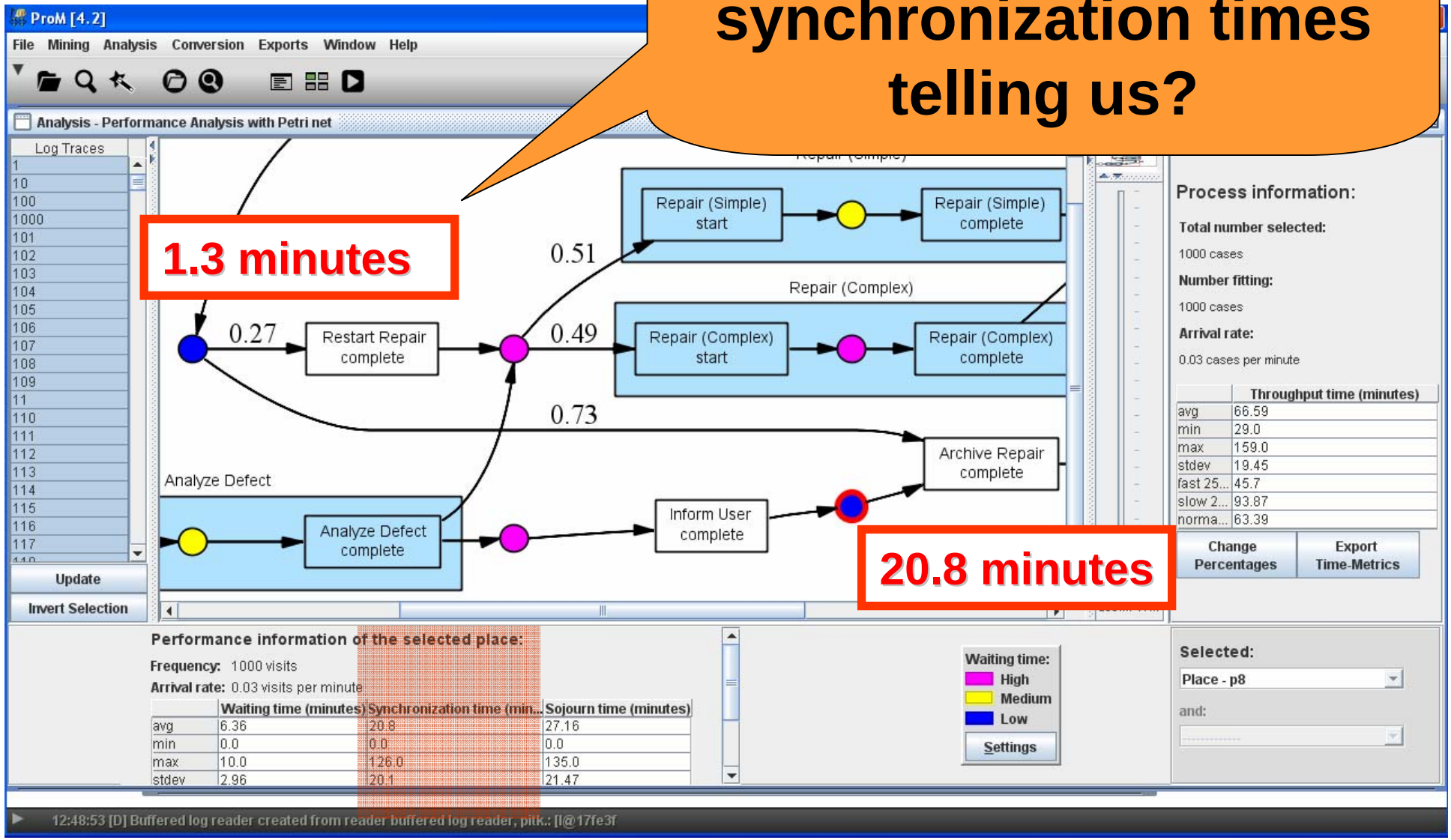


Bottlenecks – Synchronization Times

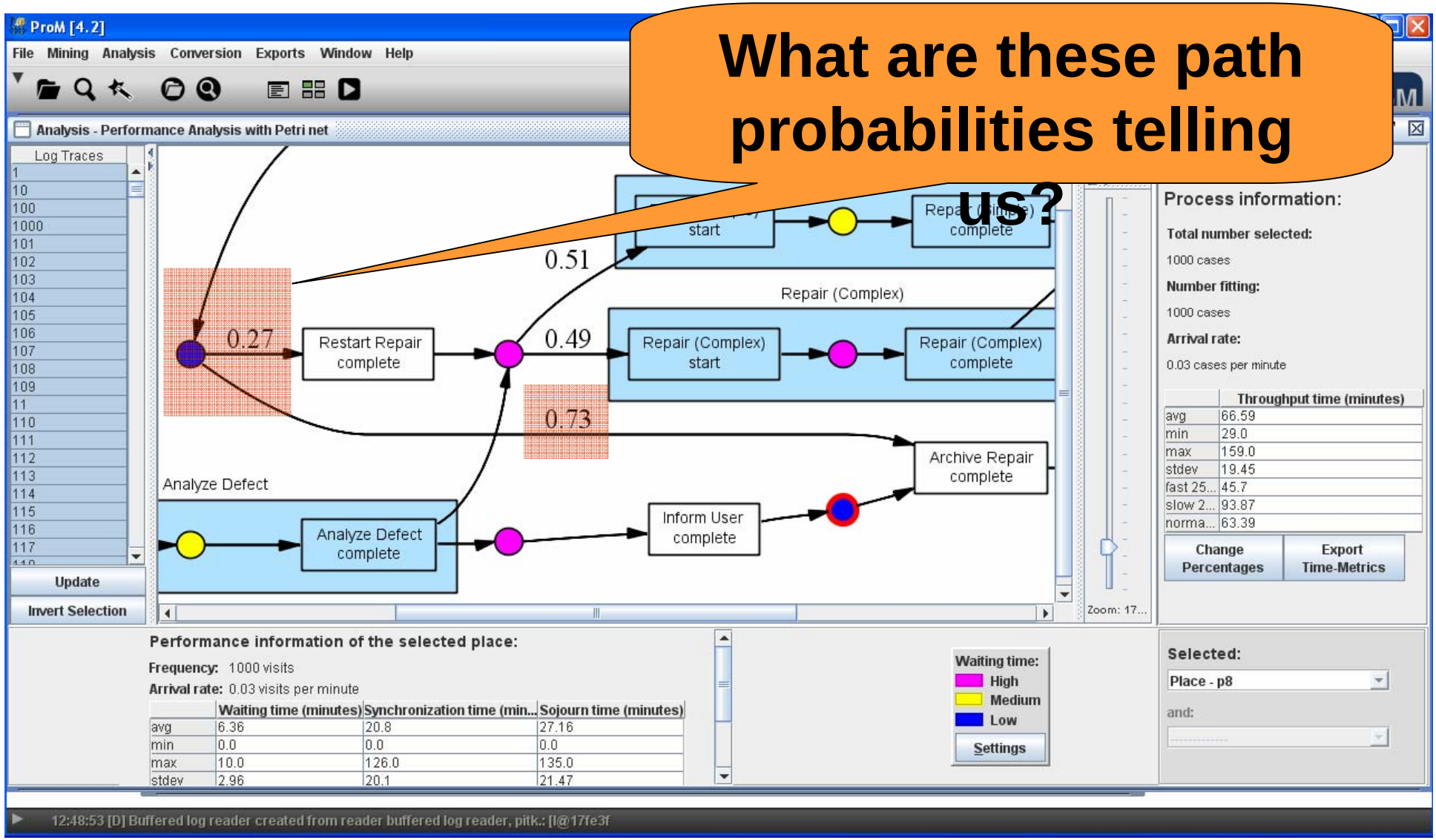


Bottlenecks – Synchronization Times

What are these average synchronization times telling us?



Bottlenecks – Path Probabilities



Performance Analysis with Petri Nets