Process Mining

Part III – Beyond control-flow mining

Organizational mining Discovery of social nets Extension algorithms

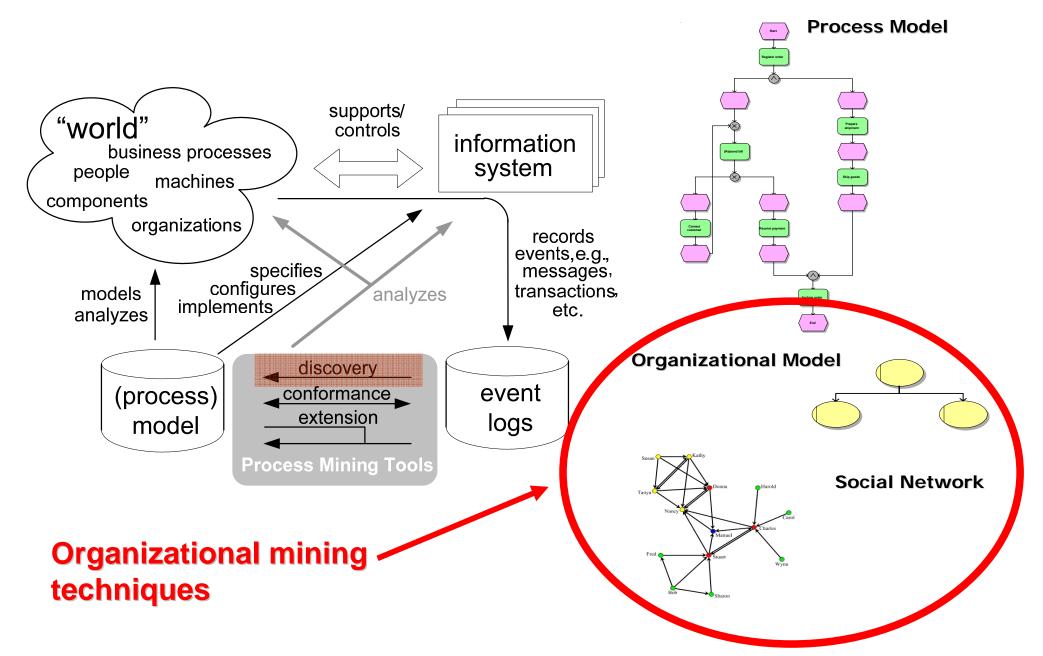




Outline

- 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



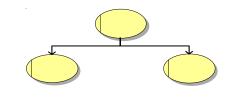


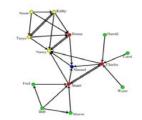


Organizational Mining Algorithms

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







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

File Mining Analysis Conversion Exports W	/indow Help	
Filtered repairExample.mxml (Simple fi Raw repairExample.mxml (unfiltered) Open new log repairExample.mxml	Partial Order Generator Partial Order Aggregator Partial Order Mining TimeUnit	ProN r d [
	Heuristics miner DWS mining plugin Association Rule Miner	
Processes	Genetic algorithm plugin Duplicate Tasks GA plug-in	Event filter
Only green processes will be used. DEFAULT	Alpha algorithm plugin Tsinghua-alpha algorithm plugin Alpha++ algorithm plugin	Only green events will be used. Register (complete) Analyze Defect (start)
Filter	Transition System Generator Region miner Language-based Region miner	Analyze Defect (complete) Repair (Complex) (start) Repair (Complex) (complete) Test Repair (start)
	Fuzzy Miner Cloud chamber miner Activity Clustering Miner	Test Repair (complete) Inform User (complete) Archive Repair (complete) Repair (Simple) (start)
Event types Event types may be removed	Change mining plugin Frequency abstraction miner	Repair (Simple) (complete) Restart Repair (complete)
or their enclosing instances discarded.	Social network miner StaffAssignmentMiner	
complete (keep)	Organizational Miner	
start (remove)	Workflow patterns miner Case data extraction plugin	
	Logreader Benchmark LogReader comparison driver	



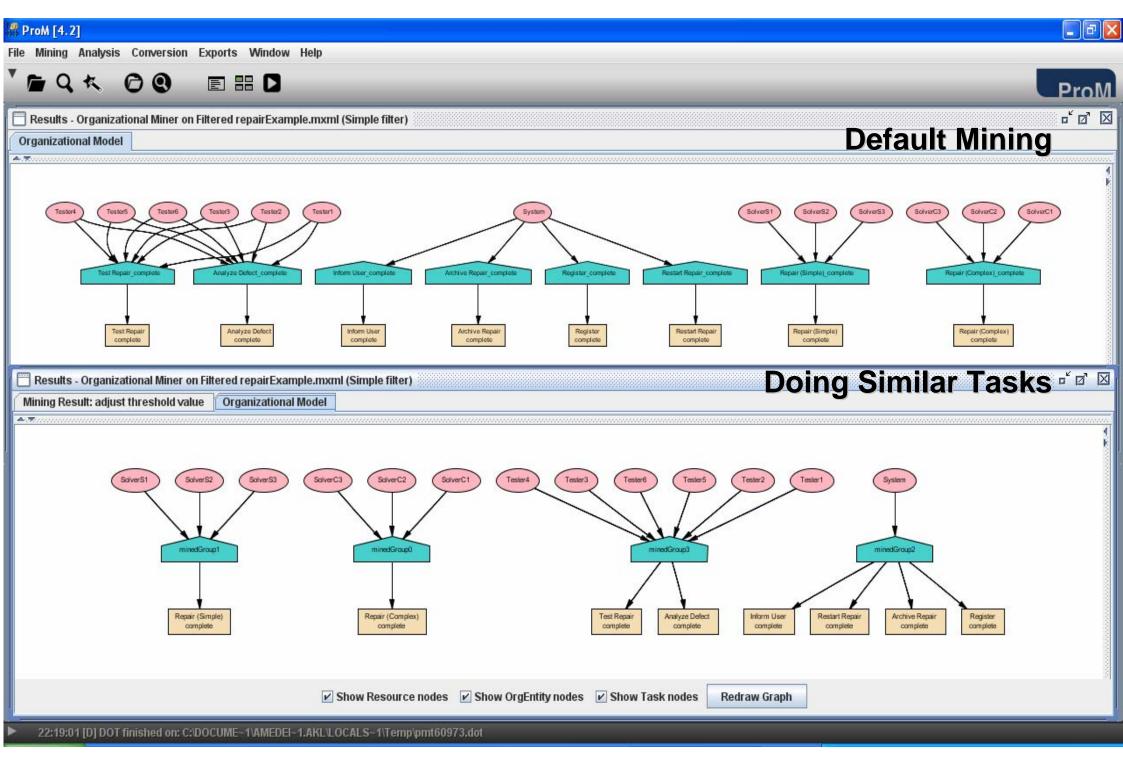
- Main idea: Which performers are executing which tasks
- Methods to mine roles
 Default mining
 - Doing Similar Tasks
- Methods to mine *teams*
 - Working together

🐺 ProM [4.2]
File Mining Analysis Conversion Exports Window Help
🔲 Settings for mining Filtered repairExample.mxml (Simple f 🗗 🔽
Organizational Miner
Method for OM Default Mining
O Doing Similar Task (DST)
 Working Together (WT)
Options for DST
Euclidian distance
Correlation coefficient
 Similarity coefficient
Hamming distance
Options for WT
O Simultaneous appearance ratio
Consider distance with causality
Consider distance without causality (beta=0.5)
Help start mining
21:46:42 [M] Process mining finished.



- 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
🛅 Settings for mining Filtered repairExample.mxml (Simple f 🗗 🗹
Organizational Miner
Method for OM
 Default Mining
Doing Similar Task (DST)
 Working Together (WT)
Options for DST
 Euclidian distance
Correlation coefficient
 Similarity coefficient
 Hamming distance
Options for WT
Simultaneous appearance ratio
Consider distance with causality
Consider distance without causality (beta=0.5)
Help start mining
21:46:42 [M] Process mining finished.





- 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 An	alysis Conversion Exports Window Help
* 🖻 Q, ł	🔨 🕑 🖻 📰 🗖 🛛 🔍 ProM
Settings for	mining Filtered repairExample.mxml (Simple f 🗗 🗹
Organizati	ional Miner
Method	for OM O Default Mining
	🔿 Doing Similar Task (DST)
	Working Together (WT)
Options	for DST
	Euclidian distance
	Correlation coefficient
	O Similarity coefficient
Options	O Hamming distance
Options	Simultaneous appearance ratio
	 Consider distance with causality
	 Consider distance without causality (beta=0.5)
Help	start mining
▶ 21:46:42 [N	A] Process mining finished.



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

<۷>	Vorl	kflov	wLog>			
	<process></process>					
		<processinstance></processinstance>				
		<audittrailentry></audittrailentry>				
			<audittrailentry></audittrailentry>			
			<audittrailentry></audittrailentry>			
		<td>ProcessInstance></td>	ProcessInstance>			
		<p< td=""><td>rocessInstance/></td></p<>	rocessInstance/>			
		<p< td=""><td>rocessInstance/></td></p<>	rocessInstance/>			
	<td>Proc</td> <td>ess></td>	Proc	ess>			
</td <td>Wor</td> <td>kflo</td> <td>wLog></td>	Wor	kflo	wLog>			

<AuditTrailEntry>

<WorkflowModelElement/> Task A </Wf.M.E.>

<EventType> complete </EventType>

<TimeStamp> 2005-10-26T12:37:33... </TimeStamp>

<Originator> John Doe </Originator>

<Data>

<Attribute name="x"> 1 </Attribute>

<Attribute name="y"> whatever </Attribute>

</Data>

</AuditTrailEntry>



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



Aim:

 Monitor how individual process instances are routed between originators

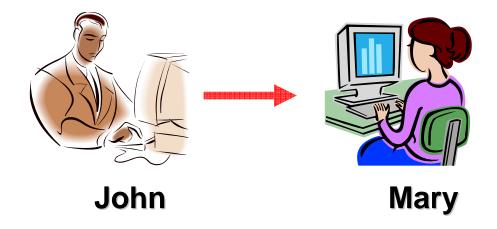
Metrics

- Handover of work
- Subcontracting
- Reassignment
- Working together
- Similar task

iele Mining Analysis Conversion Exports Window Help iele Mining Analysis Conversion Exports Window Help iele Mining Conversion Exports Window Help iele Mining States iele Mining Mining Filtered repairExample.mxml (Simple filter) using Social network min et al. iele iele iele iele iele iele iele ie	ProM [4.2]						
Settings for mining Filtered repairExample.mxml (Simple filter) using Social network min Social network miner Subcontracting Handover of work Working together Similar task Reassignment Consider causality Consider multiple transfers within one instance (unchecked: # of Instances where HW happened /# of instances checked: # of Instances under the originators /# of possible HWs) Consider only direct succession Beta: 0.5 Depth of calcutation SolverC1 SolverC2 240 SolverC3 SolverS3 215 SolverS3 SolverS4 SolverS4 SolverS4	le Mining Analysis	s Conversion Expo	orts Window Help				
Social network miner Subcontracting Handover of work Working together Similar task Reassignment Consider causality Consider multiple transfers within one instance Unrobecked: # of instances where HW happened /# of instances checked: # of HW between originators /# of possible HW(s) Consider only direct succession Beta: Depth of calculation; 5 Select All SolverC1 SolverC2 240 SolverC3 183 SolverC3 183 SolverS3 215 System 3369 Tester1 406 Tester3 404	■ Q K	00			Pr	٥N	
Social network miner Subcontracting Handover of work Working together Similar task Reassignment Consider causality Consider multiple transfers within one instance Unrobecked: # of instances where HW happened /# of instances checked: # of HW between originators /# of possible HW(s) Consider only direct succession Beta: Depth of calculation; 5 Select All SolverC1 SolverC2 240 SolverC3 183 SolverC3 183 SolverS3 215 System 3369 Tester1 406 Tester3 404	Settings for mining	g Filtered repairExam	ple.mxml (Simple filte	r) using Social r	etwork min 🖬 🛛	a" [
Subcontracting Handover of work Working together Similar task Reassignment Consider causality Consider multiple transfers within one instance (unchecked: # of instances where HW happened / # of instances obeloked: # of instances of the breaken originators / # of possible HWs) Consider only direct succession Beta: 0.5 Depth of calcutation: Select All SolverC1 249 SolverC3 SolverS3 215 SolverS3 SolverS4 SolverS4 SolverS4 SolverS4 SolverS4 SolverS4 SolverS4 S				,		_	
Select All Originators Frequency Selected Select All SolverS1 267 ✓ Deselect All SolverS2 215 ✓ System 3369 ✓ ✓ Minimum = 183 ✓ SolverS2 216 System 3369 ✓ ✓ Tester1 406 ✓ ✓ Tester3 404 ✓ ✓	Social network miner						
Subcontracting Consider causality Consider multiple transfers within one instance Unchecked: # of instances where HW happend / # of instances checked: # of HW between originators /# of possible HW(s) Consider only direct succession Beta: 0.5 Depth of calculation: 5 Select All Originators Frequency Selected SolverC1 240 ✓ SolverC3 183 ✓ SolverS1 267 ✓ SolverS2 215 ✓ System 3369 ✓ Tester1 406 ✓ Tester2 406 ✓ Tester3 404 ✓							
Consider multiple transfers within one instance (unchecked: # of instances where HW happend / # of instances checked: # of HW between originators /# of possible HW(s) ✓ Consider only direct succession Beta: 0.5 Depth of calculations 5 Select All Originators Frequency Select All SolverC1 240 SolverC3 183 ✓ SolverC3 183 ✓ SolverS1 267 ✓ SolverS3 215 ✓ SolverS3 215 ✓ System 3369 ✓ Tester1 406 ✓ Tester3 404 ✓	Subcontracting	Handover of work	Working together	Similar task	Reassignment		
Consider multiple transfers within one instance (unchecked: # of instances where HW happend / # of instances checked: # of HW between originators /# of possible HWs) Consider only direct succession Beta: 0.5 Depth of calculation: 5 Select All Originators Frequency Select All SolverC1 249 SolverC3 183 ✓ SolverC3 183 ✓ SolverS1 267 ✓ SolverS3 215 ✓ System 3369 ✓ Tester1 406 ✓ Tester2 406 ✓ Tester3 404 ✓							
Consider multiple transfers within one instance (unchecked: # of instances where HW happend / # of instances checked: # of HW between originators /# of possible HWs) Consider only direct succession Beta: 0.5 Depth of calculation: 5 Select All Originators Frequency Select All SolverC1 249 SolverC3 183 ✓ SolverC3 183 ✓ SolverS1 267 ✓ SolverS3 215 ✓ System 3369 ✓ Tester1 406 ✓ Tester2 406 ✓ Tester3 404 ✓		Consider caus	ality				
Select All Originators Frequency Selected SolverC3 183 ✓ SolverC3 183 ✓ SolverC3 183 ✓ SolverC3 183 ✓ SolverS3 215 ✓ SolverS3 216 ✓ SolverS3 215 ✓ SolverS3 215 ✓ SolverS3 215 ✓ SolverS3 216 ✓ SolverS3 216 ✓ <t< td=""><th></th><th></th><td>2</td><td>na inotanaa</td><td></td><td></td></t<>			2	na inotanaa			
checked: # of HW between originators /# of possible HW(s) ✓ Consider only direct succession Beta: 0.5 Depth of calculation: 5 Ø Originators Frequency Selected SolverC1 240 ✓ SolverC2 240 ✓ SolverC3 183 ✓ SolverS1 267 ✓ SolverS2 215 ✓ SolverS3 215 ✓ System 3369 ✓ Tester1 406 ✓ Tester2 406 ✓ Tester3 404 ✓							
Beta: 0.5 Depth of calculation: 5 Select All Originators Frequency Selected SolverC1 249 ✓ SolverC2 240 ✓ SolverC3 183 ✓ SolverC3 183 ✓ SolverS1 267 ✓ SolverS2 215 ✓ SolverS3 215 ✓ SolverS3 215 ✓ SolverS3 216 ✓ System 3369 ✓ Tester1 406 ✓ Tester2 406 ✓ Tester3 404 ✓							
Depth of calculation: 5 Select All Originators Frequency Selected SolverC1 249 ✓ SolverC2 240 ✓ SolverC3 183 ✓ SolverC3 183 ✓ SolverS3 215 ✓ SolverS3 215 ✓ SolverS3 215 ✓ SolverS3 215 ✓ SolverS3 216 ✓ SolverS3 216 ✓ System 3369 ✓ Tester1 406 ✓ Tester2 406 ✓ Tester3 404 ✓		Consider only	direct succession				
Select All Originators Frequency Selected SolverC1 249 ✓ SolverC2 240 ✓ SolverC3 183 ✓ SolverC3 183 ✓ SolverS1 267 ✓ SolverS2 215 ✓ SolverS3 215 ✓ SolverS3 215 ✓ SolverS3 216 ✓ SolverS3 215 ✓ SolverS3 216 ✓ Tester1 406 ✓ Tester3 404 ✓							
Originators Frequency Selected SolverC1 249 ✓ SolverC2 240 ✓ SolverC3 183 ✓ SolverS1 267 ✓ SolverS2 215 ✓ SolverS3 215 ✓ SolverS3 215 ✓ SolverS3 216 ✓ SolverS3 216 ✓ Tester1 406 ✓ Tester2 406 ✓ Tester3 404 ✓							
Select All SolverC1 249 ✓ Select All SolverC2 240 ✓ Deselect All SolverS1 267 ✓ Filter SolverS3 215 ✓ Minimum = 183 ✓ SolverS3 215 Maximum = 3369 ✓ Tester1 406 ✓ Tester2 406 ✓ Tester3 404 ✓			liculation: 5				
Select All SolverC1 249 ✓ Select All SolverC2 240 ✓ Deselect All SolverS3 183 ✓ SolverS3 215 ✓ SolverS3 215 ✓ SolverS3 215 ✓ SolverS3 216 ✓ Minimum = 183 ✓ 3369 ✓ Tester1 406 Tester2 406 ✓ Tester3		4					
Select All SolverC2 240 ✓ SolverC3 183 ✓ SolverC3 183 ✓ SolverS1 267 ✓ SolverS2 215 ✓ SolverS3 215 ✓ SolverS3 215 ✓ SolverS1 406 ✓ Tester1 406 ✓ Tester2 406 ✓ Tester3 404 ✓				quency			
Select All SolverC3 183 ✓ Deselect All SolverS1 267 ✓ SolverS2 215 ✓ SolverS3 215 Filter System 3369 ✓ Tester1 406 ✓ Maximum = 3369 ✓ Tester2 406 ✓ Tester3 404 ✓							
Select All SolverS1 267 ✓ Deselect All SolverS2 215 ✓ Filter SolverS3 215 ✓ Minimum = 183 Tester1 406 ✓ Maximum = 3369 ✓ Tester2 406 ✓							
Deselect All SolverS3 215 ✓ Filter System 3369 ✓ Minimum = 183 Tester1 406 ✓ Tester2 406 ✓ ✓ Tester3 404 ✓	Select All						
SolverS3 215 ✓ Filter System 3369 ✓ Minimum = 183 Tester1 406 ✓ Maximum = 3369 Tester2 406 ✓	Decelect All	SolverS2	215		V		
Minimum = 183 Tester1 406 ✓ Maximum = 3369 Tester3 404 ✓	Deselect All	SolverS3			v		
Minimum = 183 Tester2 406 Image: Constraint of the state	Filter						
Maximum = 3369 Tester2 406 V Tester3 404 V	Minimum =	192					
		Tester2					
Help	indaman -	lester3	404		V		
	Help				start minin	a	

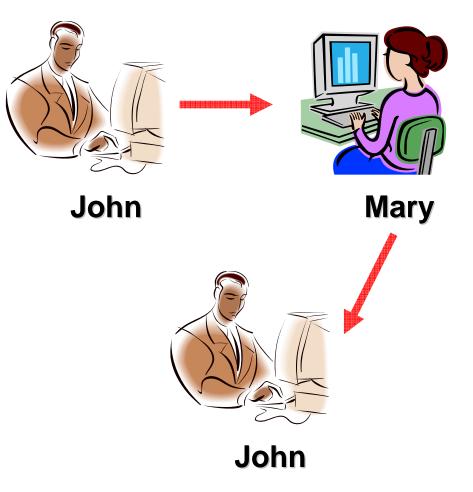


- Aim: Monitor how individual process instances are routed between originators
- Metrics
 - Handover of work
 - Subcontracting
 - Reassignment
 - Working together
 - Similar task



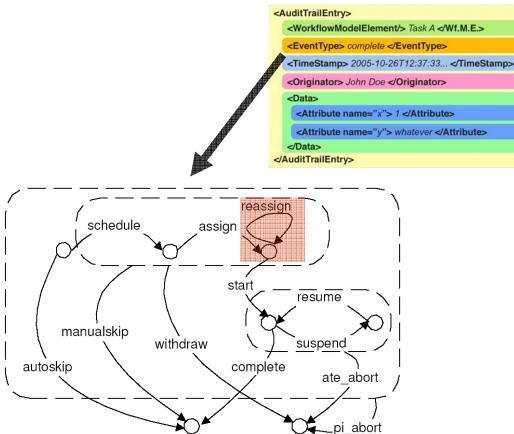


- Aim: Monitor how individual process instances are routed between originators
- Metrics
 - Handover of work
 - Subcontracting
 - Reassignment
 - Working together
 - Similar task





- Aim: Monitor how individual process instances are routed between originators
 AuditTrailEntry> (WorkflowModelElement/> Task A
- Metrics
 - Handover of work
 - Subcontracting
 - Reassignment
 - Working together
 - Similar task





- 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!

Settings for mini			
Social netwo		e.mxml (Simple filter) usir	ng Social network min
Social netwol	rk miner		
Subcontracting	Handover of work	Working together Sim	ilar task Reassignme
	(unchecked: # of insta	e transfers within one ins ances where HW/happened / # of	instances
	Consider only dir	etween onginators /# of possible ect succession	fws)
	Consider only dir	ect succession	y Selected
	Consider only dir	ect succession	y Selected
	Consider only dir	ect succession	y Selected
Select All	Consider only dir	ect succession 793855 198855 tors Frequence 249 240 183	y Selected
Select All	Consider only dir	ect succession	y Selected
Select All	Consider only dir Originat SolverC1 SolverC3 SolverC3 SolverS1 SolverS1	ect succession	y Selected
Deselect A	Image: Consider only direction Image: Consider only direction SolverC1 SolverC2 SolverC3 SolverS1 SolverS2 SolverS3	ect succession 2005 249 249 240 183 267 215 215	y Selected
Deselect A Filter	Consider only dir Consider only dir Originat SolverC1 SolverC3 SolverS1 SolverS2 SolverS3 System Tocted	ect succession	y Selected
Deselect A	Image: Consider only direction Image: Consider only direction SolverC1 SolverC2 SolverC3 SolverS1 SolverS2 SolverS3	ect succession 2005 249 249 240 183 267 215 215	y Selected



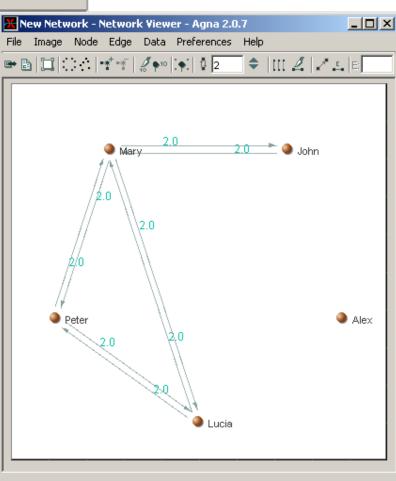
Social Network Miner: Example

- 🗆 ×

	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

<u>File Edit</u> 🕒 🔒 🦊 🛤 💽 📑 Log File Number 2 CLIQUES Minimum Set Size: Input dataset: C:\Program Files WARNING: Valued graph. All values > 0 treated 2 cliques found. 1: Lucia Peter Mary 2: John Mary Mary Actor-by-Actor Clique Co-Membership Matrix Ź.O. 12345 JALPM 1 John 10001 2 Alex 0 0 0 0 0 3 Lucia 0 0 1 1 1 4 Peter 0 0 1 1 1 5 Mary 10112 HIERARCHICAL CLUSTERING OF EQUIVALENCE MATRIX Peter LΡ 2.0 AueJM lctoa eiehr xarny 23415 Level 1.000 . XXX XXX 0.667 . XXXXXXX XXXXXXXXX 0.000 • Drag and drop nodes to change position.

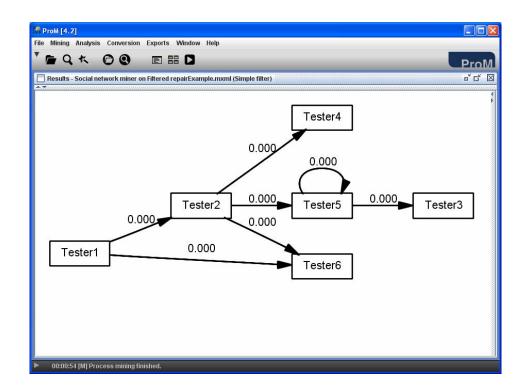
📅 Output Log #2



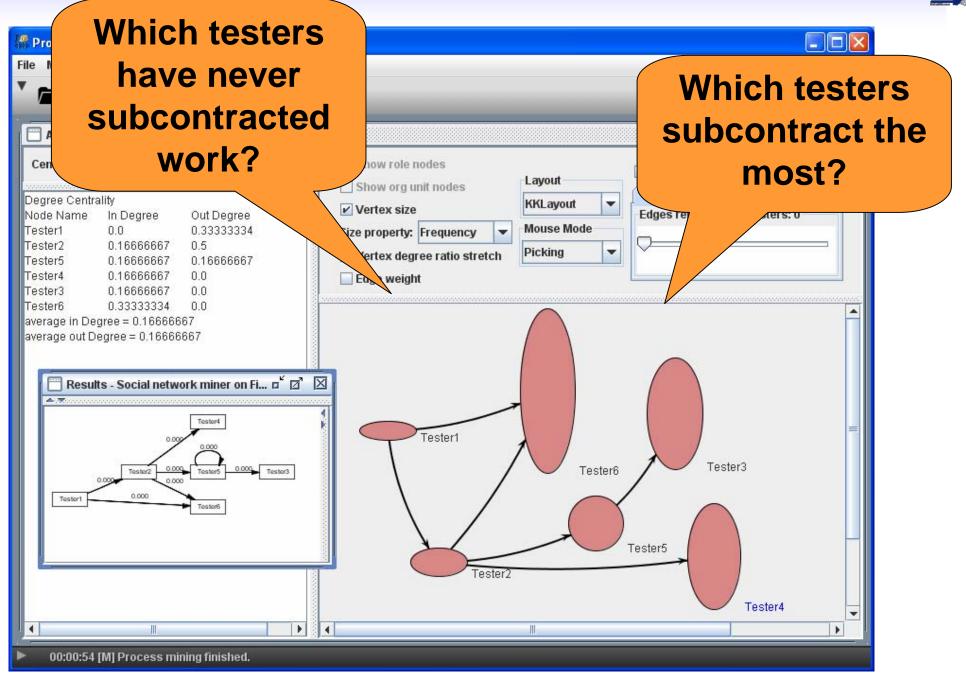


Plugin Analyze Social Network

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









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

specifies

discovery

conformance extension

configures.

implements,

business processes

machines

organizations

(process)

model

"world"

people

components

models

analyzes

Process Mining Tools Enhance existing models with information discovered from logs

analyzes

supports/

controls

information

system

event

logs

records events,e.g.,

messages,

transactions,

etc.

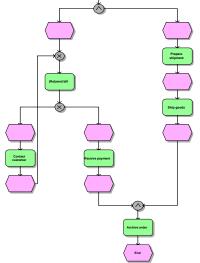
- The Decision Point Analysis plug-in can discover the "business rules" for the moments of choice in a process model
- The Performance Analysis with Petri Nets plug-in provides various KPIs w.r.t. the execution of processes

Register order

Bottlenecks/

Business Rules

Process Model





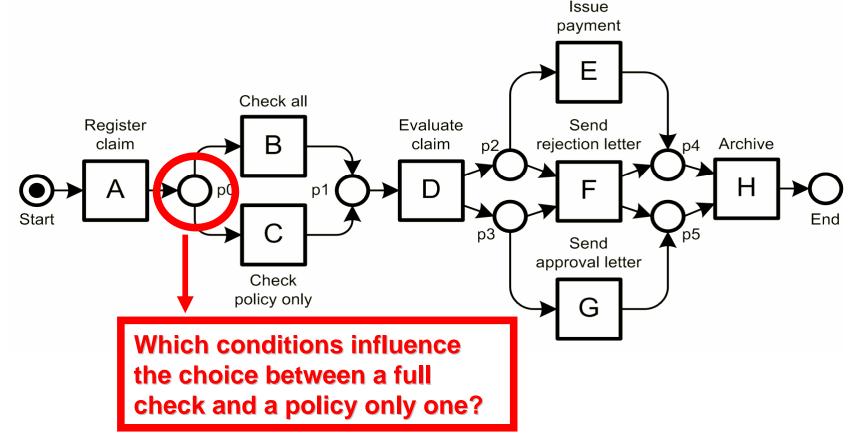
Performance





Decision Point Analysis: Main Idea

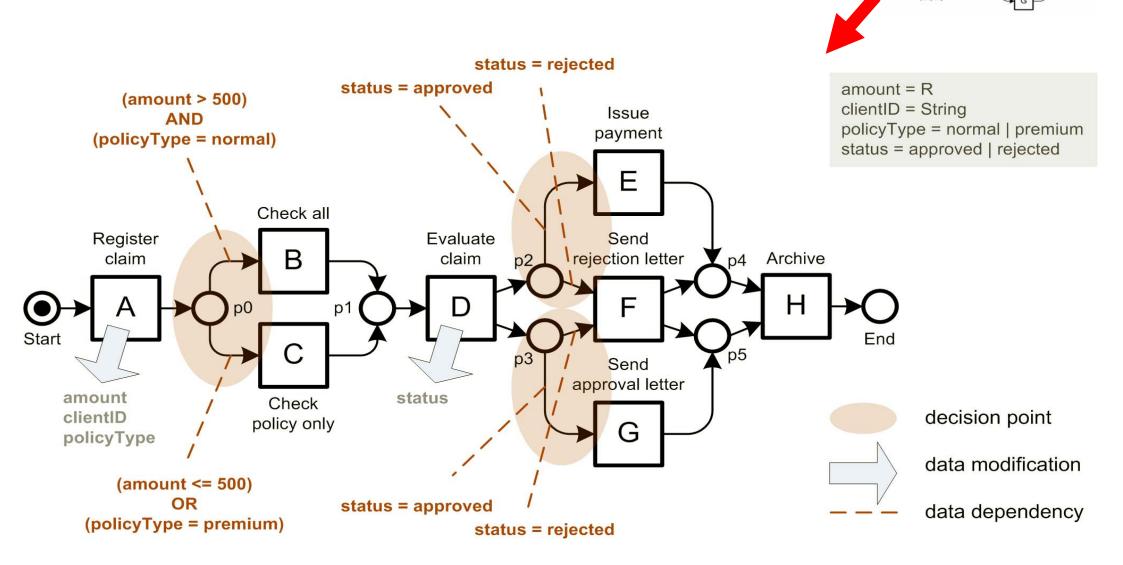
 Detection of data dependencies that affect the rounting the routing of process instances



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

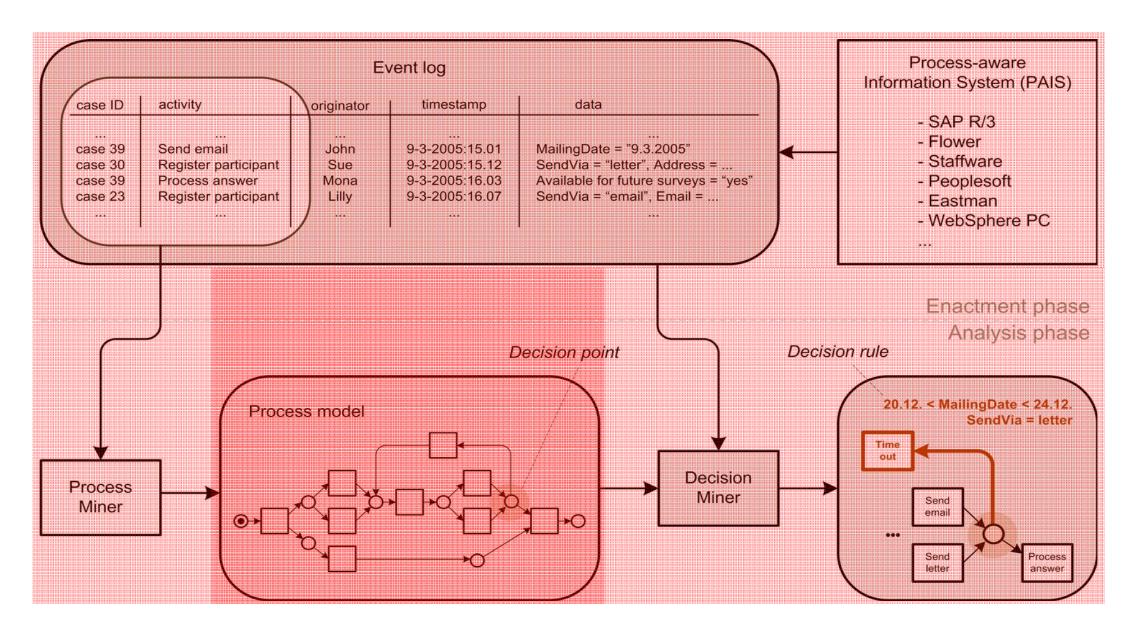


Decision Point Analysis: Motivation





Decision Point Analysis: Approach



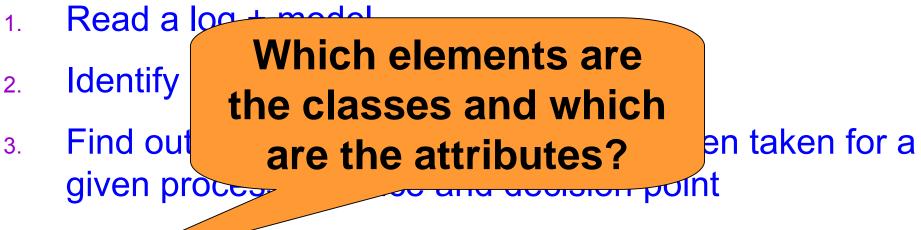


Decision Point Analysis

- 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

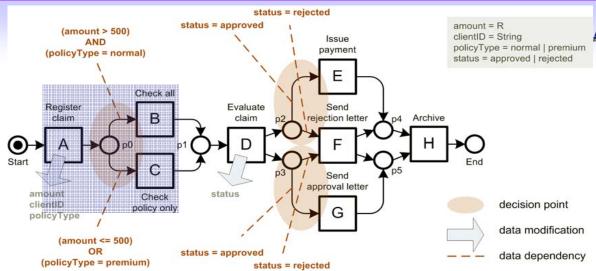


Decision Point Analysis



- 4. Discourse the rules for each decision point
- 5. Return the enhanced model with the discovered rules

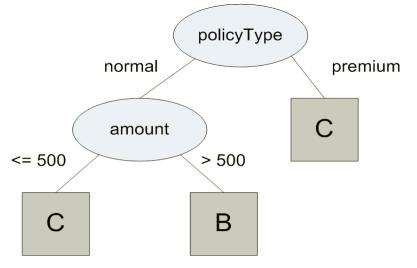
Step 4



Training examples for decision point "p0"

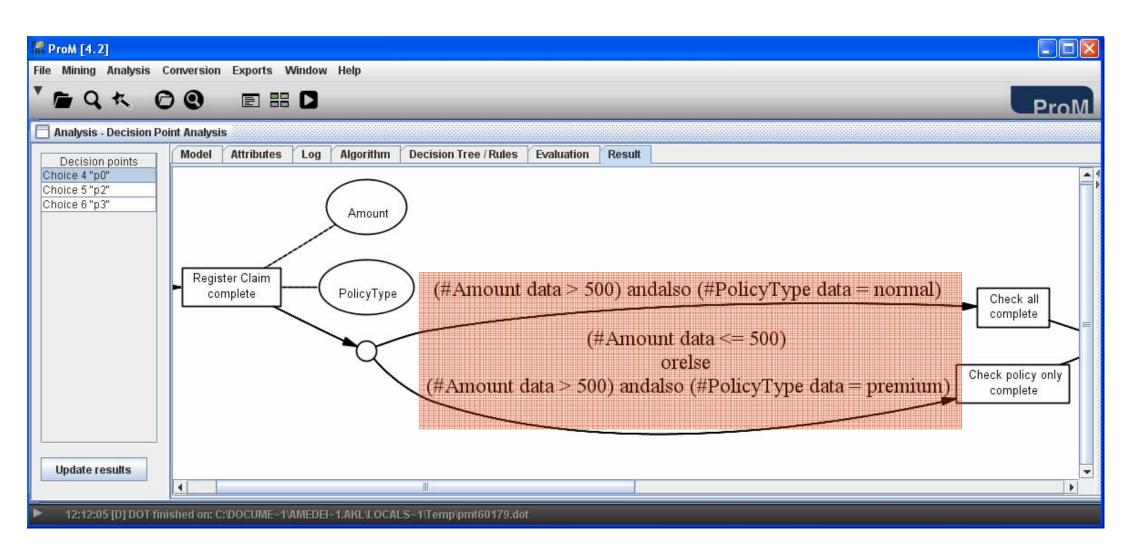
amount	clientID	policyType	class
1000	C567894938	premium	С
700	C938609223	normal	В
550	C135697567	normal	В
500	C568120443	normal	С
50	C493823084	normal	С
200	C945675110	premium	С

Discovered decision tree for point "p0"



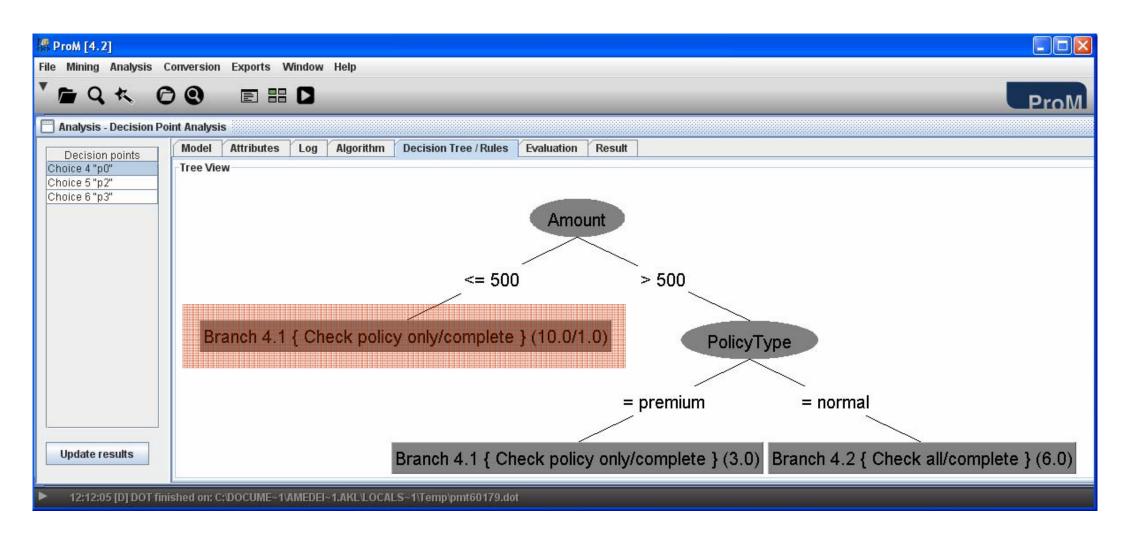


Decision Point Analysis: Example in ProM



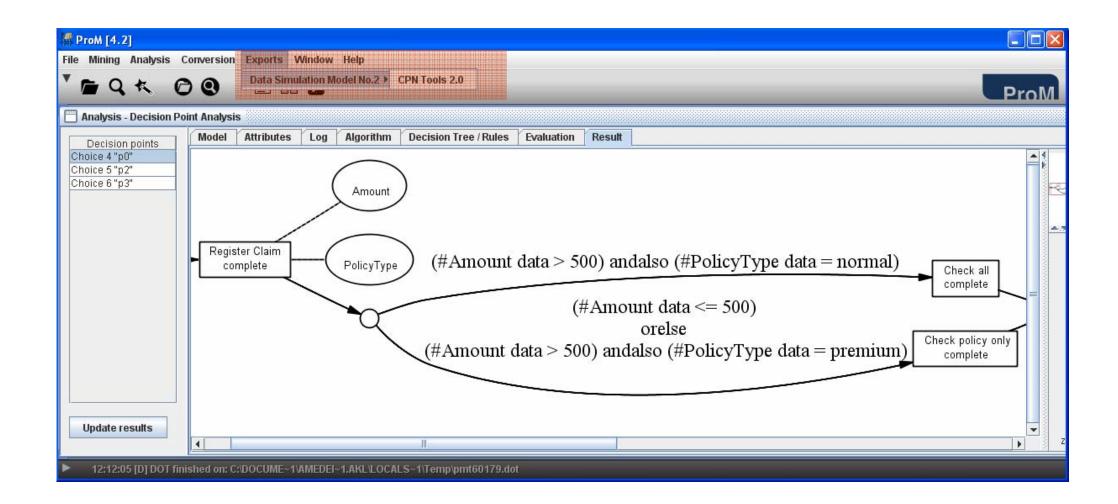


Decision Point Analysis: Example in ProM





Decision Point Analysis





Extension techniques

- Decision Miner
- Performance Analysis

Performance analysis: pattern visualization





Performance Analysis with Petri Nets

Motivation

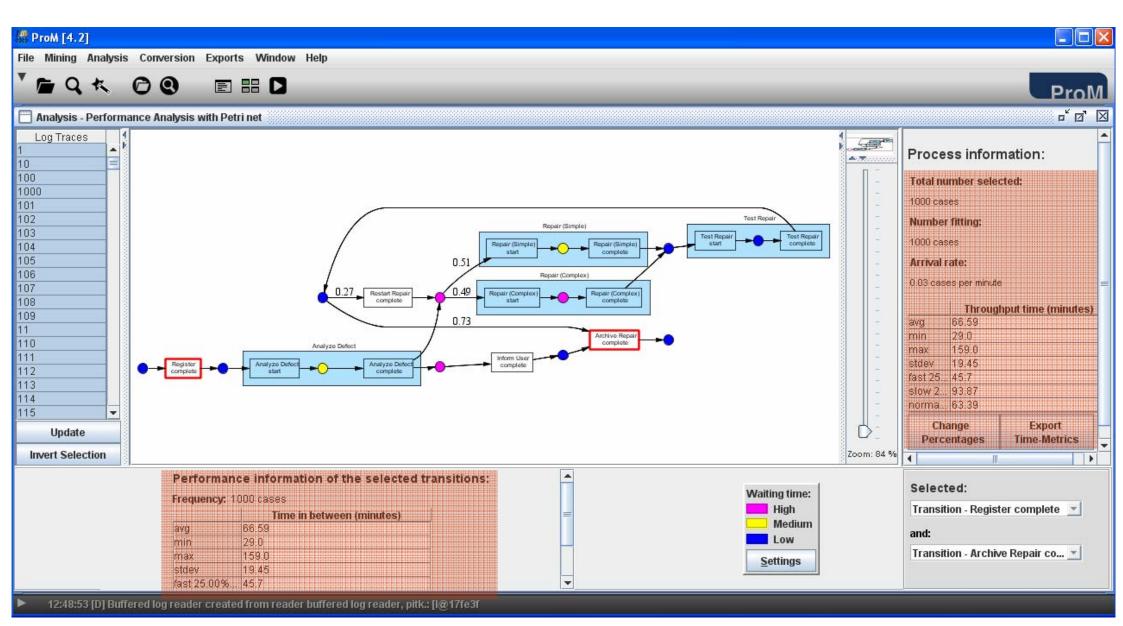
 Provide different Key Performance Indicators (KPIs) relating to the execution of processes

Main idea

- Replay the log in a model and detect
 - Bottlenecks
 - Throughput times
 - Execution times
 - Waiting times
 - Synchronization times
 - Path probabilities etc

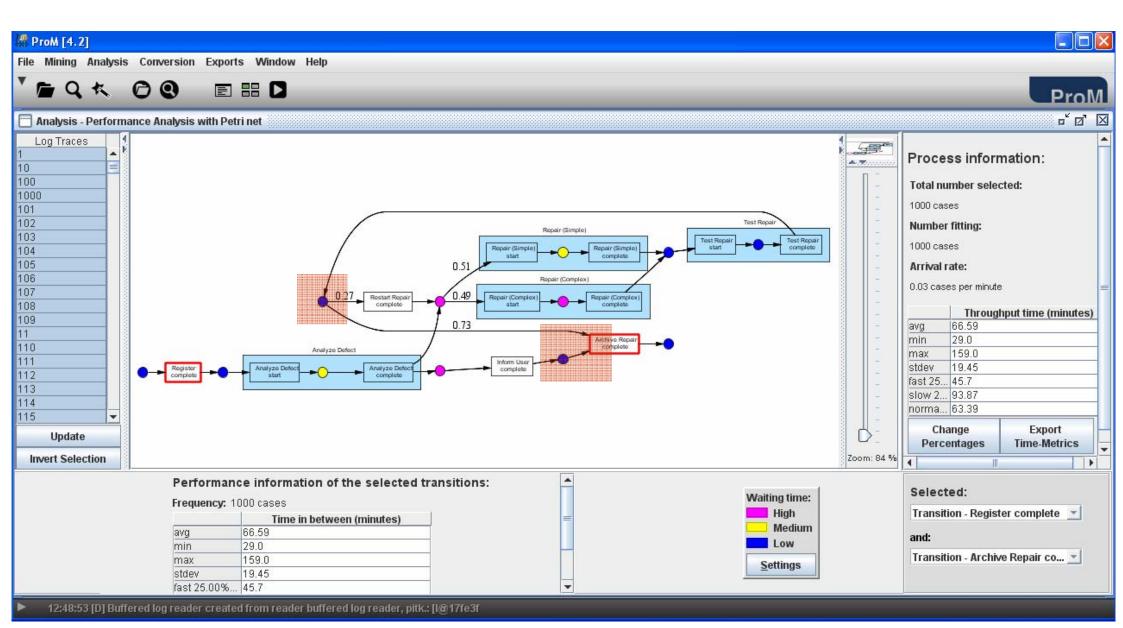


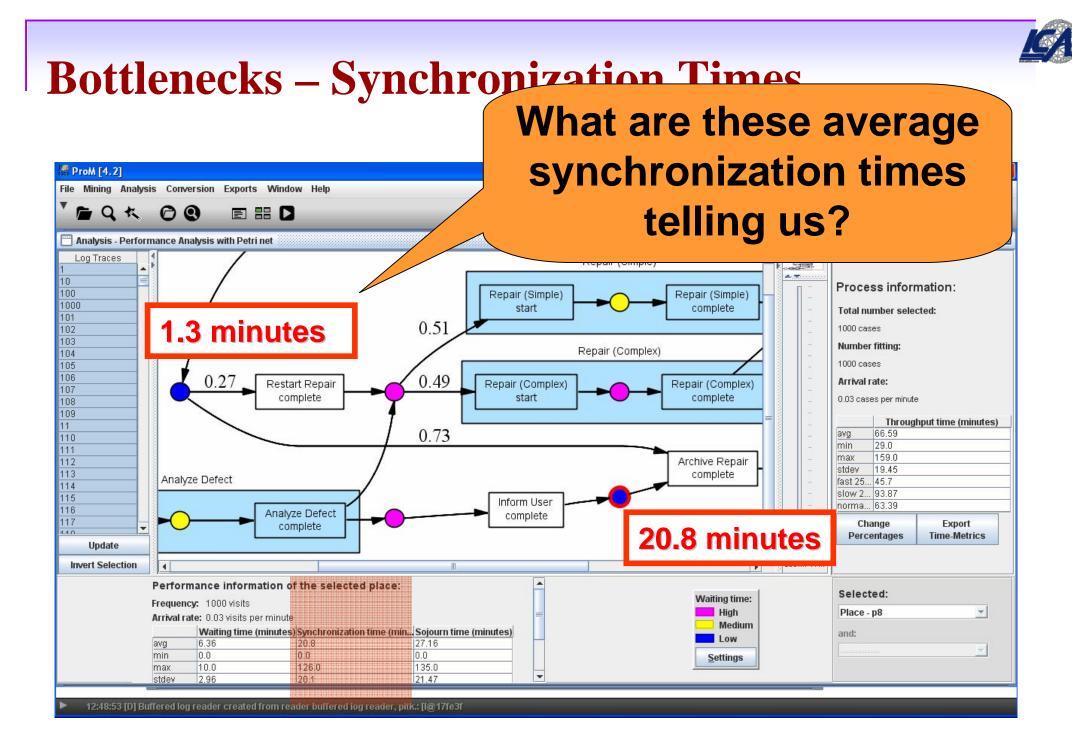
Bottlenecks – Throughput Times





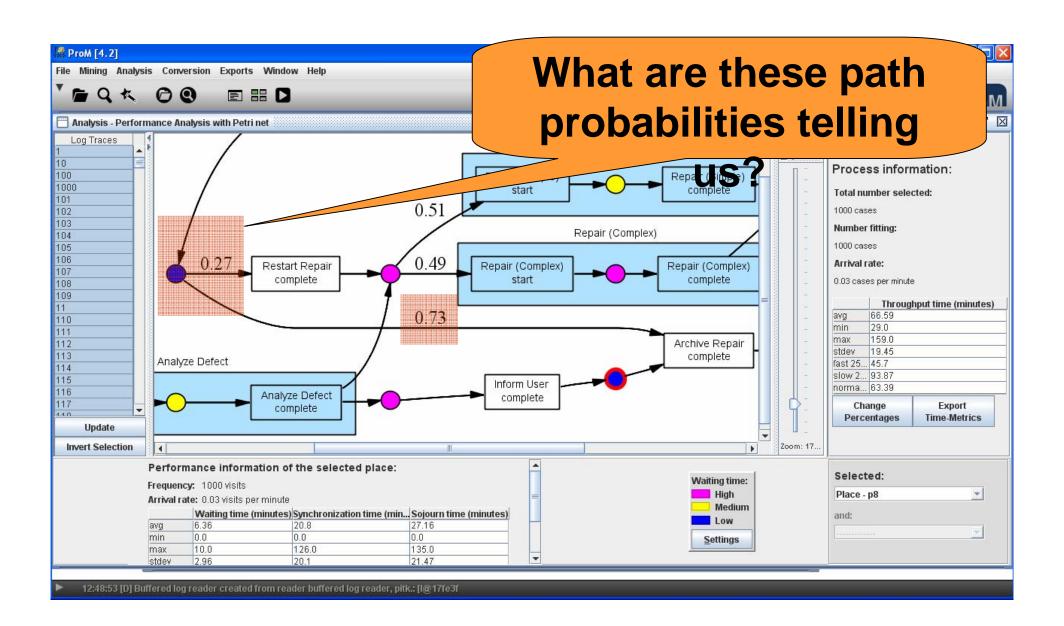
Bottlenecks – Synchronization Times







Bottlenecks – Path Probabilities





Performance Analysis with Petri Nets

