

Applicability of Process Viewing Patterns in Business Process Management

14th IEEE International Enterprise Distributed Object Conference
Workshop on Models and Model-driven Methods for Service Engineering
October 25, 2010

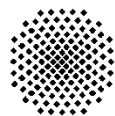
David Schumm



Frank Leymann



Institute of Architecture
of Application Systems



Universität Stuttgart
Germany

Tobias Anstett,
Daniel Schleicher



Overview

- Process View Fundamentals
 - Process view metamodel constructs
 - Process viewing patterns (excerpt)
- Process Viewing Scenarios
 - Process design
 - Process deployment
 - Process monitoring
 - Visual process analysis
 - General purpose
- Conclusion and Outlook

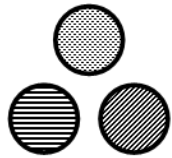
Process View Fundamentals

Process Views

- What is a process view...?
 - The presentation of the result from specific *transformations* applied to a process model
- What is the purpose of a process view...?
 - Reduction of complexity
 - Separation of concerns
 - Providing a problem-specific perspective
- What are process viewing patterns?
 - Elementary forms of transformations to alter processes

Process View Metamodel Constructs

■ Nodes



Typed nodes



Abstract node

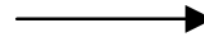


Aggregate node

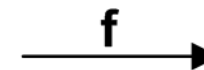


Inserted node

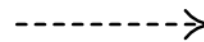
■ Edges



Control edge



Conditional control edge

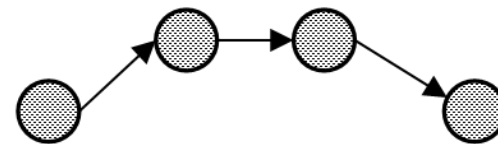
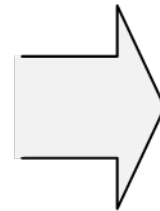
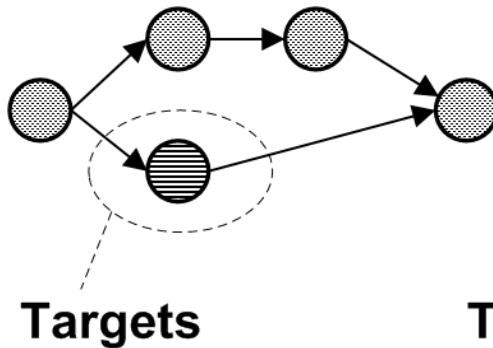


Inserted edge



Data edge

■ View Transformation

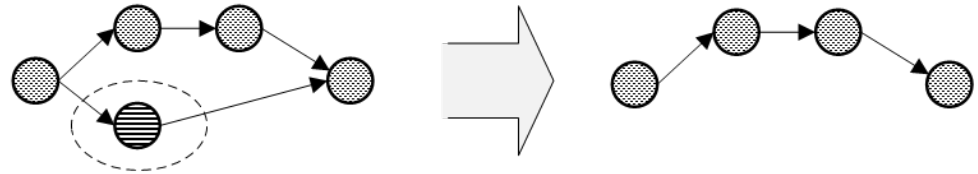


Transformation

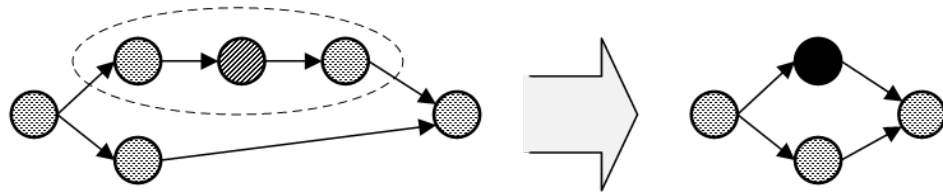
View

Structure Transformation Patterns

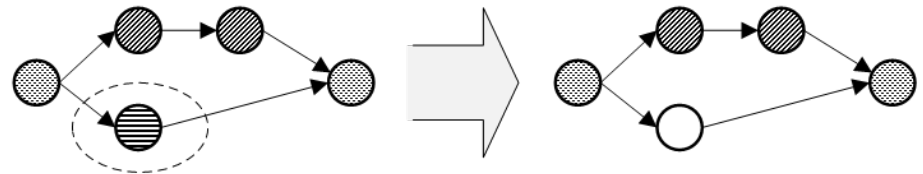
■ Omission



■ Aggregation

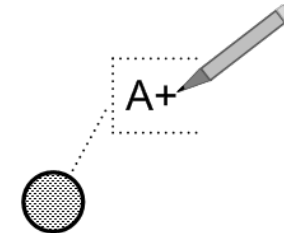
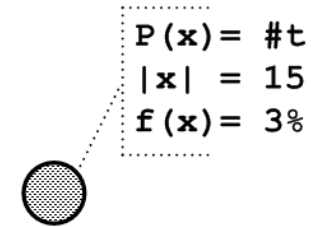
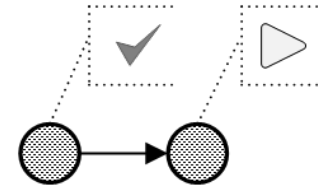


■ Abstraction



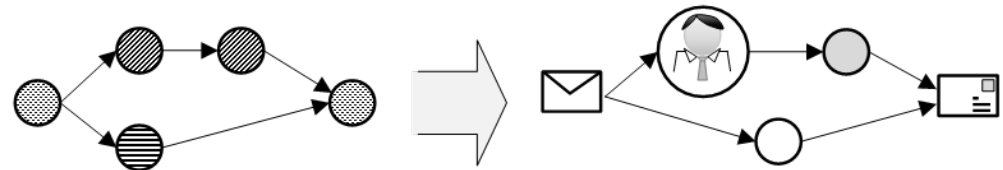
Augmentation Transformation Patterns

- Runtime Augmentation
- Calculated Augmentation
- Human-assisted Augmentation

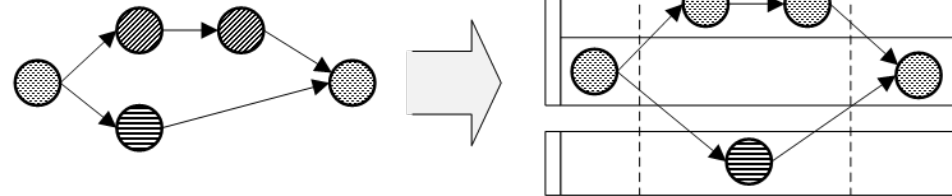


Presentation Patterns

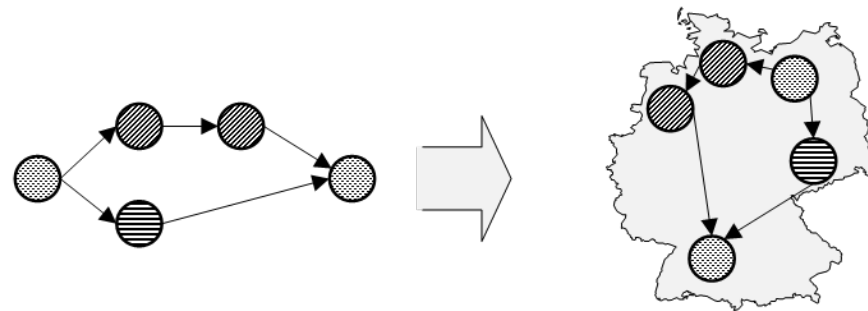
- Appearance



- Organizational Information



- Custom Categories



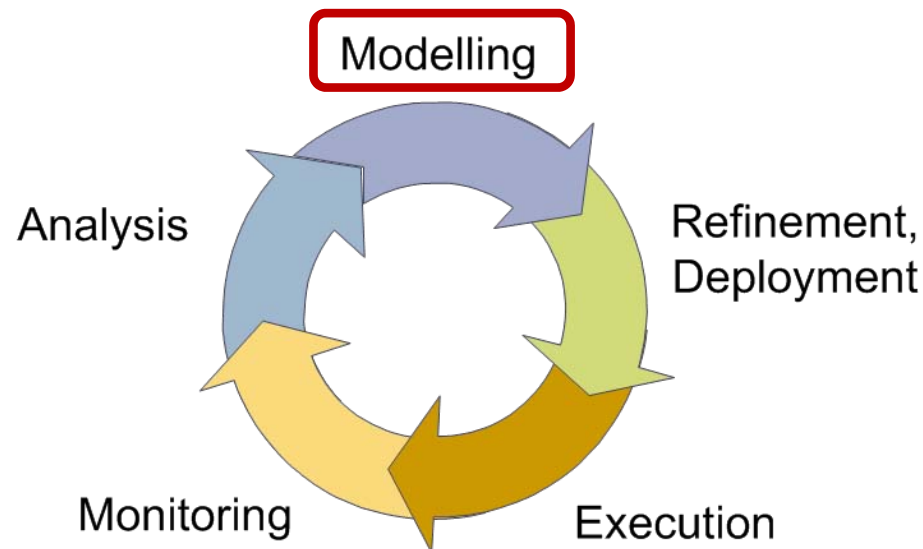
Process Viewing Scenarios

Process Viewing Scenarios

- Predefined views (discussed in the following)
 - Good solutions to *common* problems
 - View transformation can be predefined
- Custom views
 - Solutions tailored to *specific* problems
 - View transformation specified ad hoc

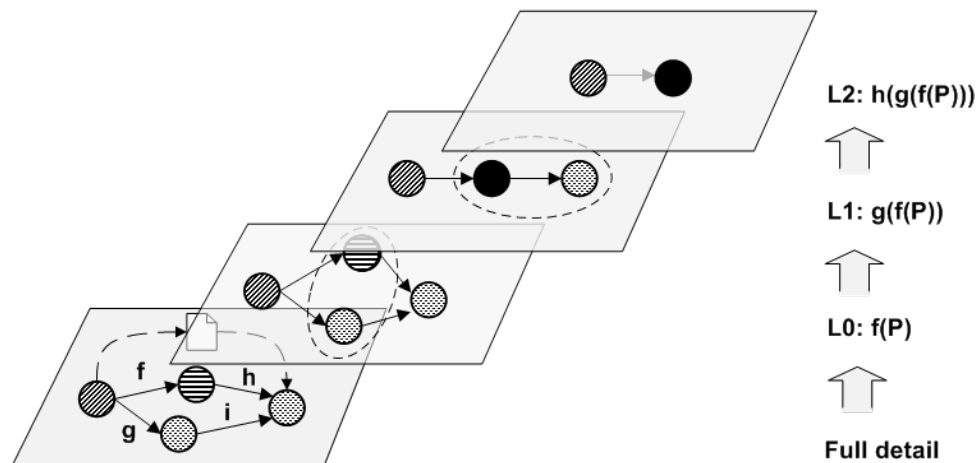
Part I: Process Design

- Process modeling on multiple levels of abstraction
- Extraction of process logic for reuse
- Process outsourcing



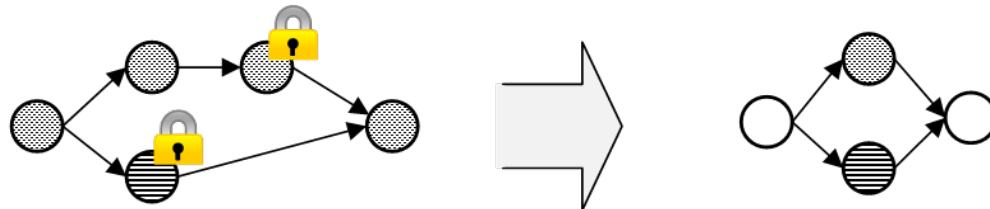
Process Modeling on Multiple Levels of Abstraction

- Abstract process modelling
 - Modelling one and the same process on *multiple levels of abstraction*
- Solution
 - Layering concepts can be applied to provide multiple levels of abstraction
 - Aggregation and omission of nodes and artifacts
 - Layers are generated dynamically by executing transformations
 - A layer can be seen as a view on a view
 - Problem: updating the models in lower layers



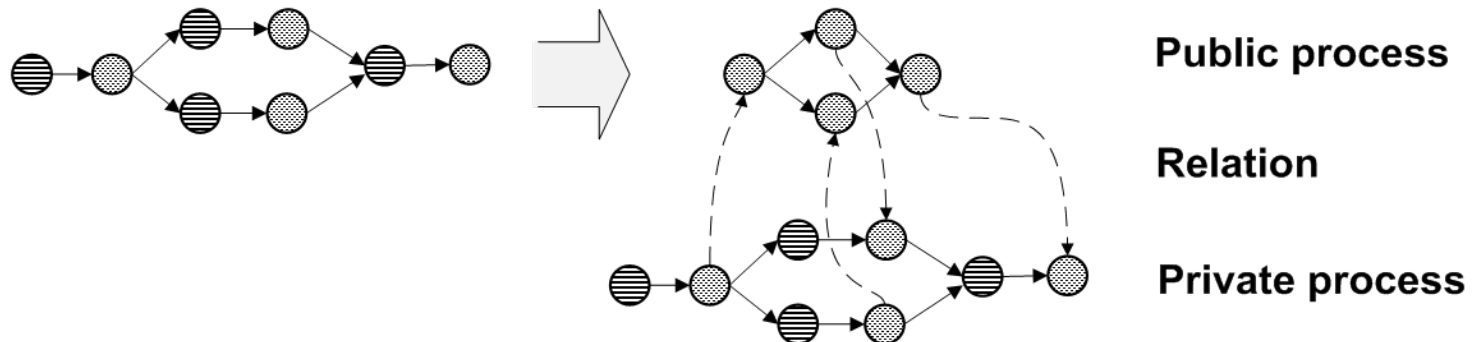
Extraction of Process Logic for Reuse

- Extraction of process logic for reuse
 - Some parts of a process might be *reusable* in other processes
 - Therefore, they shall be extracted
- Solution
 - Creation of a view which only contains the functionality that shall be reused
 - At first, these nodes need to be specified which shall be extracted for later reuse
 - In a next step all other activities are omitted
 - Control dependencies are preserved
 - But: dealing with ambiguities - automatic, manual, mixture?



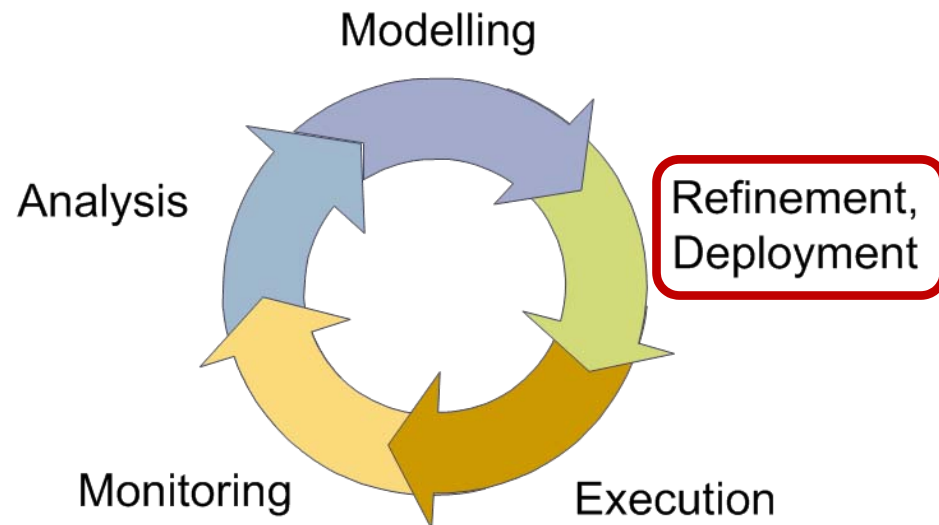
Public Views for Process Outsourcing

- Public views for process outsourcing
 - No disclosure of internal parts of the business process
 - Show the *visible behavior* of a process
- Solution
 - Classification of “confidential” and “internal” structures
 - Omission of these structures
 - Preserving required constructs
 - For each partner a different view can be generated



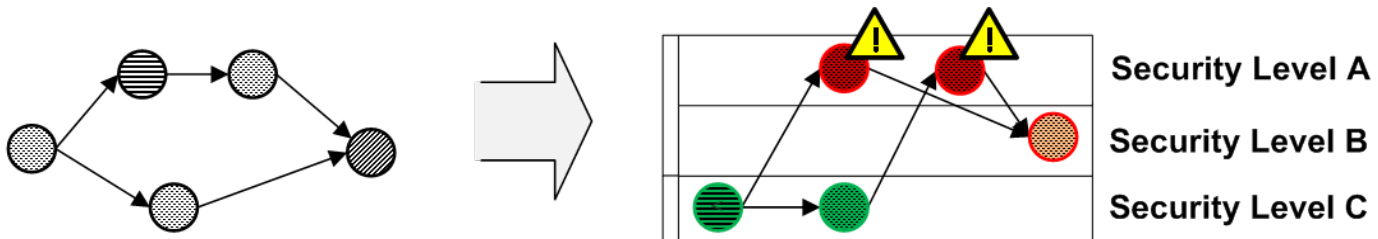
Part II: Process Deployment

- View on security
- Process distribution



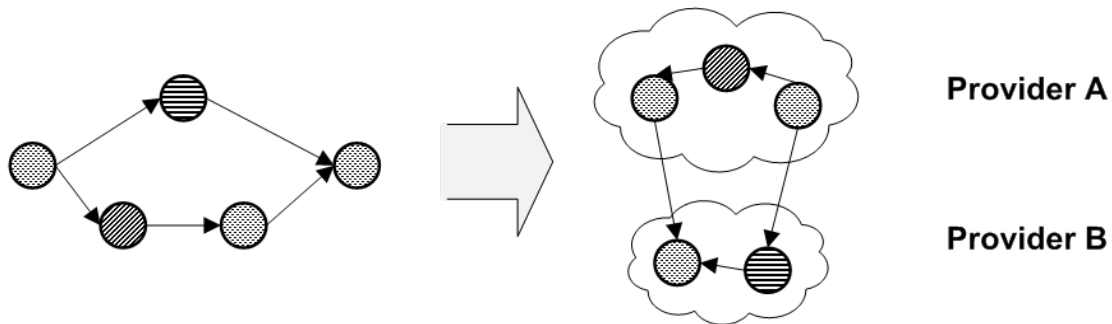
View on Security

- View on security
 - Security-related aspects are typically defined externally
- Solution
 - *Augment a process* and emphasize security-related aspects
 - Annotations about the required security levels
 - Visualize the required security level
 - Use swimlanes for different security levels
 - Border colors
 - Decorators for particular security-related aspects



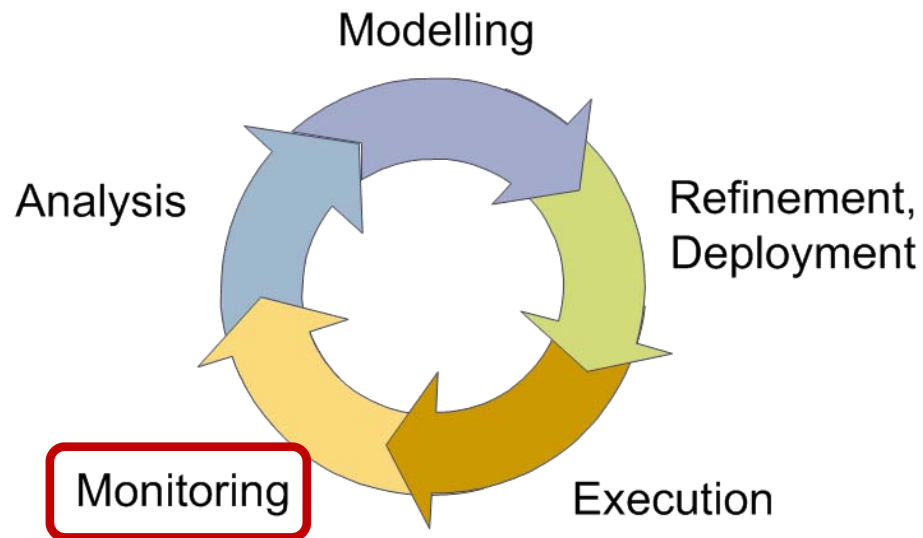
Process Distribution

- Process distribution
 - Manage the *location* of the actors and programs that execute tasks
- Solution
 - Runtime augmentation needs to provide information about the location of actors of the process
 - Those “coordinates” are used by the layouting function
 - Example: analyze and modify a process whose services are distributed over different cloud providers



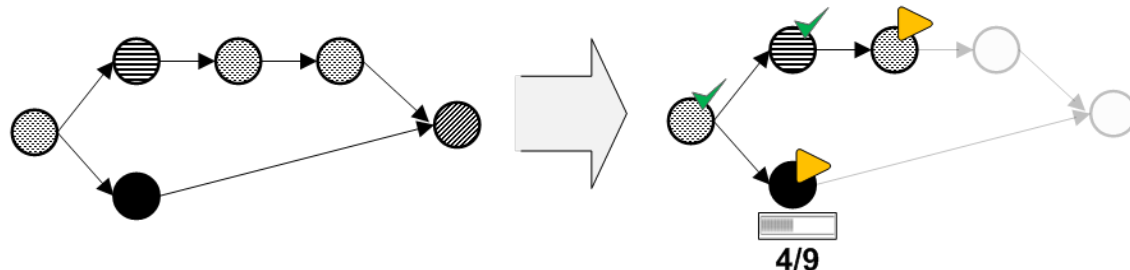
Part III: Process Monitoring

- Status of an instance
- Custom business process monitoring



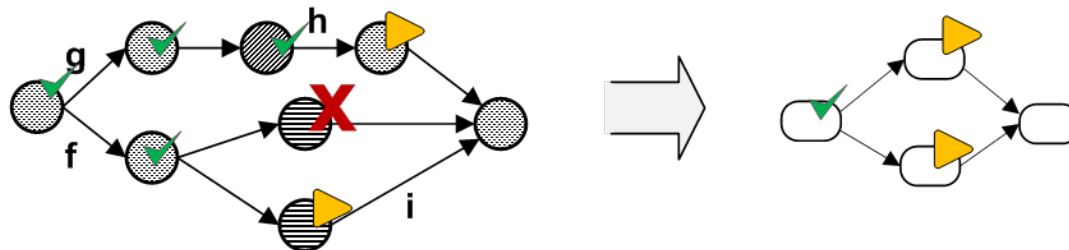
Status of an Instance

- Status of an instance
 - Process mining provides runtime information about a process
 - Having a clear view on the *status* of an instance
- Solution
 - Runtime augmentation of the process with the current status of an instance is a basic prerequisite
 - Presentation can use decorators to show the current state, while lowering the contrast for activities that are not ready yet
 - This solution can also be used for visualizing a simulation



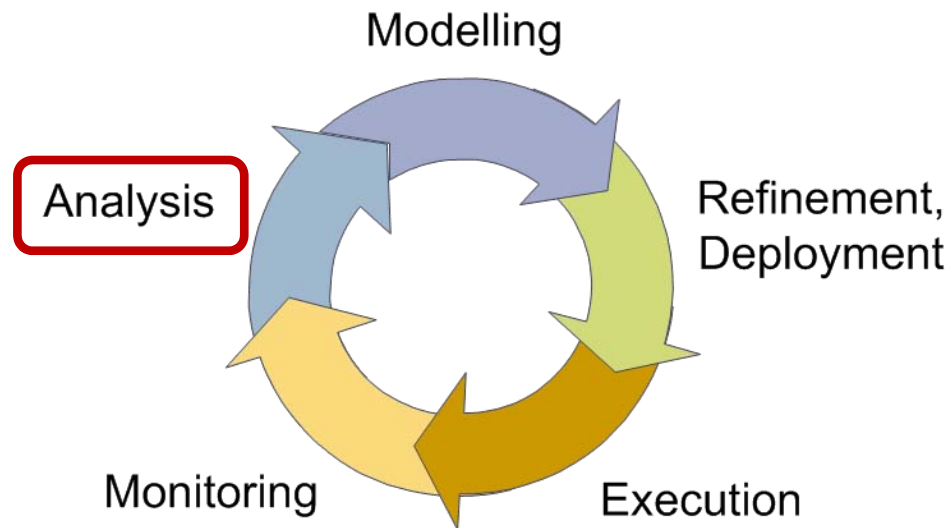
Custom Business Process Monitoring

- Custom business process monitoring
 - Modelling and execution of processes is often performed at various different levels of abstraction
 - Users near to business use more high level notations
 - Technical staff is more familiar with programming languages
- Solution
 - Define *custom views for monitoring* for different stakeholders
 - Define projections of activities and states



Part IV: Visual Process Analysis

- Bottlenecks
- Probable execution path
- Path analysis
- Process automation



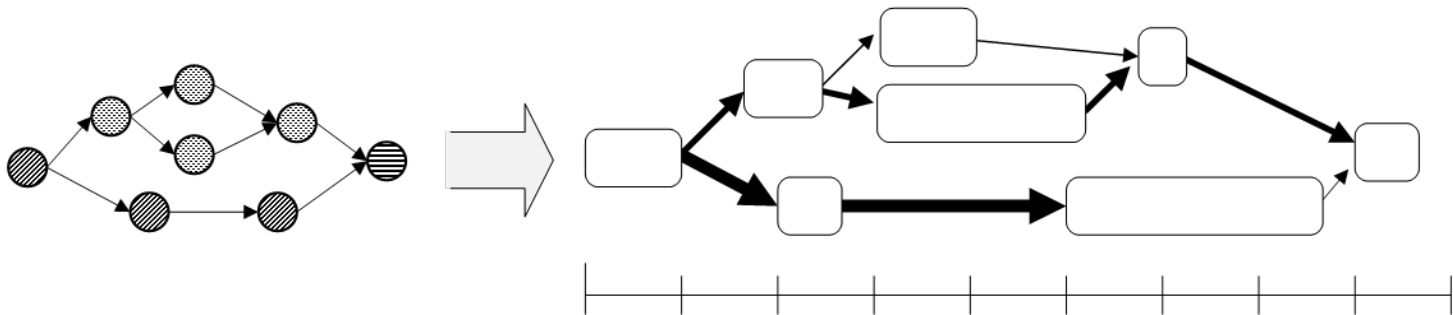
Bottlenecks

■ Bottlenecks

- Bottlenecks are *hot spots* in a process model
- They require more resources than currently are available
- A bottleneck can be rooted in the services or humans performing activities, or in the size of the transferred data

■ Solution

- Augment the process with runtime information (“cost”)
- The width of the nodes can be bound to the cost of nodes
- Data transfer cost can be bound to the thickness of edges
- As custom theme a linear time bar can be shown



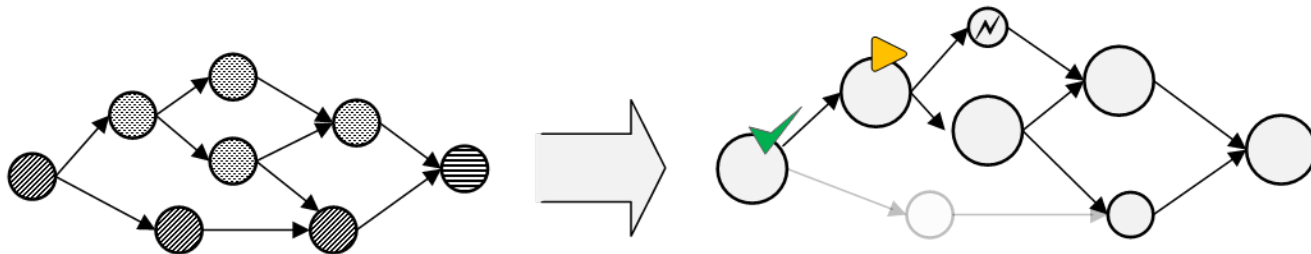
Probable Execution Path

- Probable execution path

- In some cases it is useful to *predict the future* execution path
- Intelligent proposals like “people who chose A, also liked B”

- Solution

- Runtime augmentation enriches the process with probabilities
- Presentation shows the current status using decorators
- Size of nodes can be bound to the calculated probability
- Possible extension: Omit improbable steps



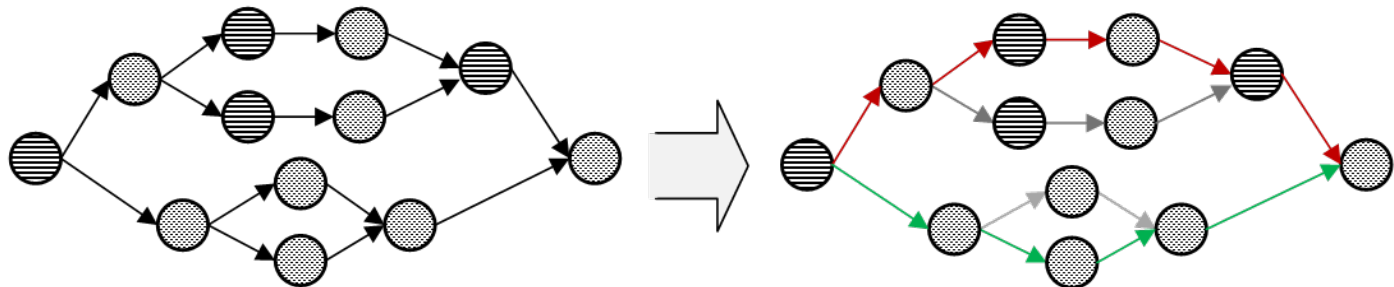
Path Analysis

■ Path analysis

- Having a clear view on the longest or *most expensive path* in a process is essential in business reengineering
- Immediate feedback if a changed design will probably lead to a cheaper execution is desirable

■ Solution

- Augmentation with calculated information about the
 - Cheapest path
 - Most expensive path
 - Shortest path (i.e., cost = time)
 - ...
- The different paths can be shown with different colors for edges



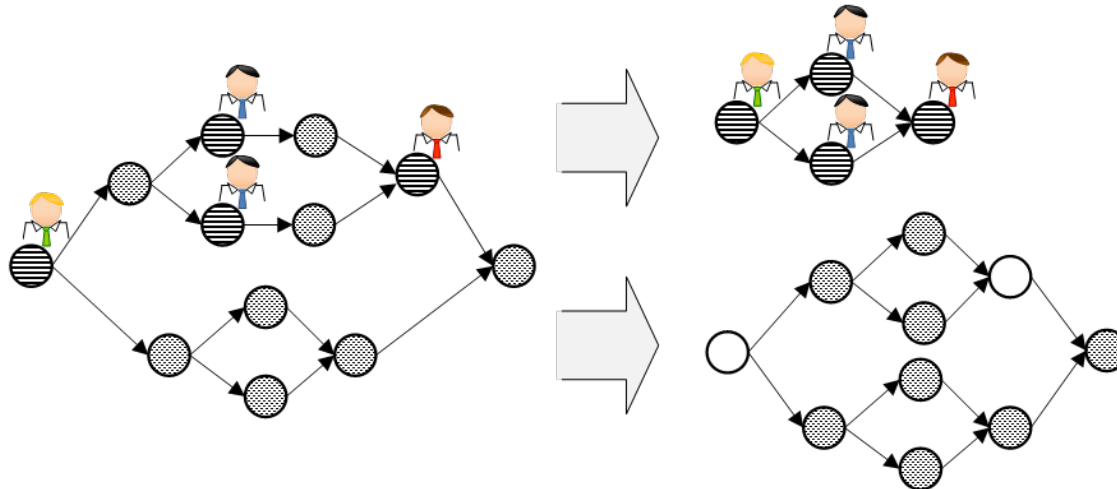
Process Automation

■ Process automation

- Processes are partially or completely executed by humans or by software
- Getting a clear insight which parts of a process are automated and which are not, is useful in many scenarios of process analysis

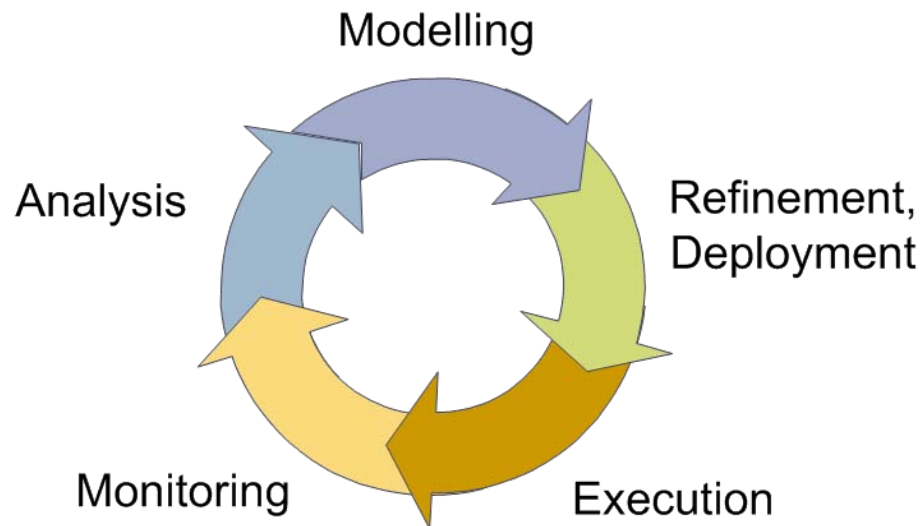
■ Solution

- Node properties allow to distinguish which nodes are executed by programs and which by humans
- Otherwise, runtime augmentation can provide this information
- This information can be displayed using decorators
- When using omission it is also possible to show just the human process graph as well as the automated process graph



Part V: General Purpose

- Custom appearance
- Abstraction from details
- Set a focus
- Access control
- Applicable in all phases



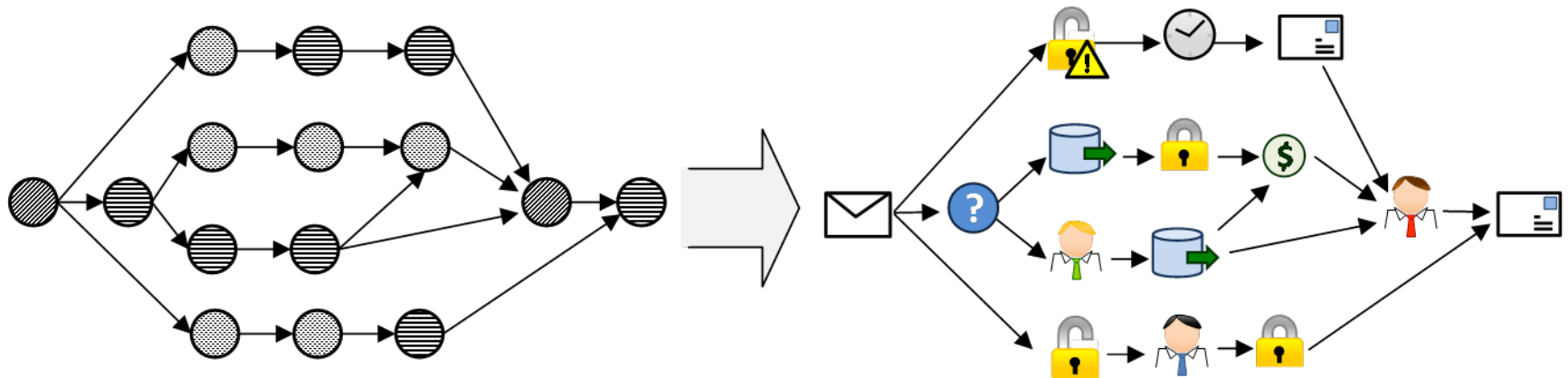
Custom Appearance

- Custom appearance

- Activities may have speaking labels
- But: no *obvious shapes* that show their meaning

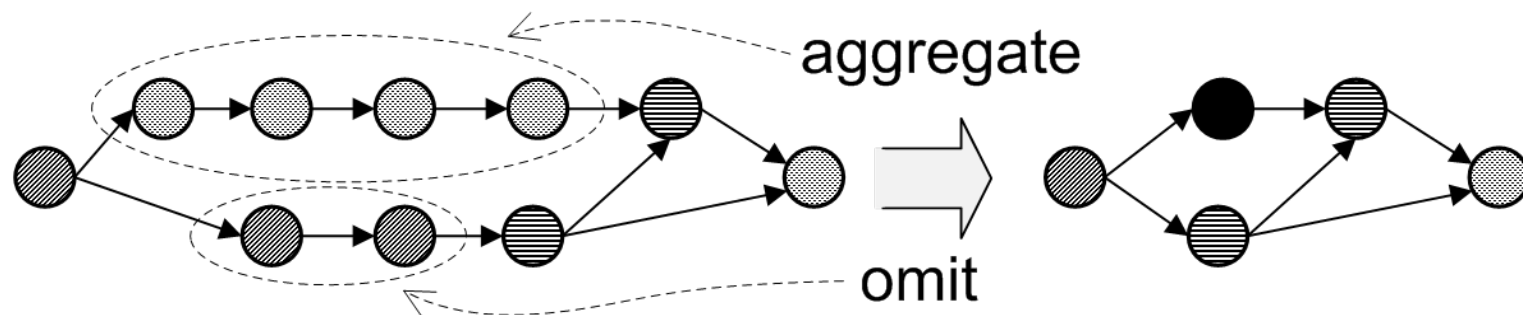
- Solution

- Presentation of nodes and artifacts with particular appearance
- Use “speaking” shapes that denote their meaning
- Human-assisted augmentation can provide a classification of activities, possibly based on an ontology



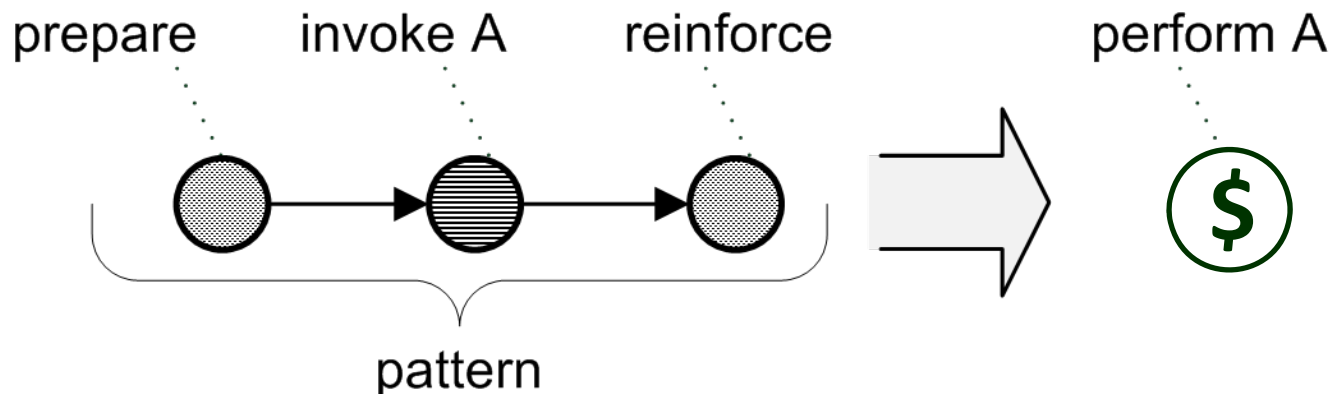
Abstraction from Details

- Abstraction from details
 - Some nodes and artifacts are of *minor importance*
- Solution
 - Multiple abstraction levels, or user profiles
 - Complexity can be reduced by removing details
 - Omission of nodes and artifacts



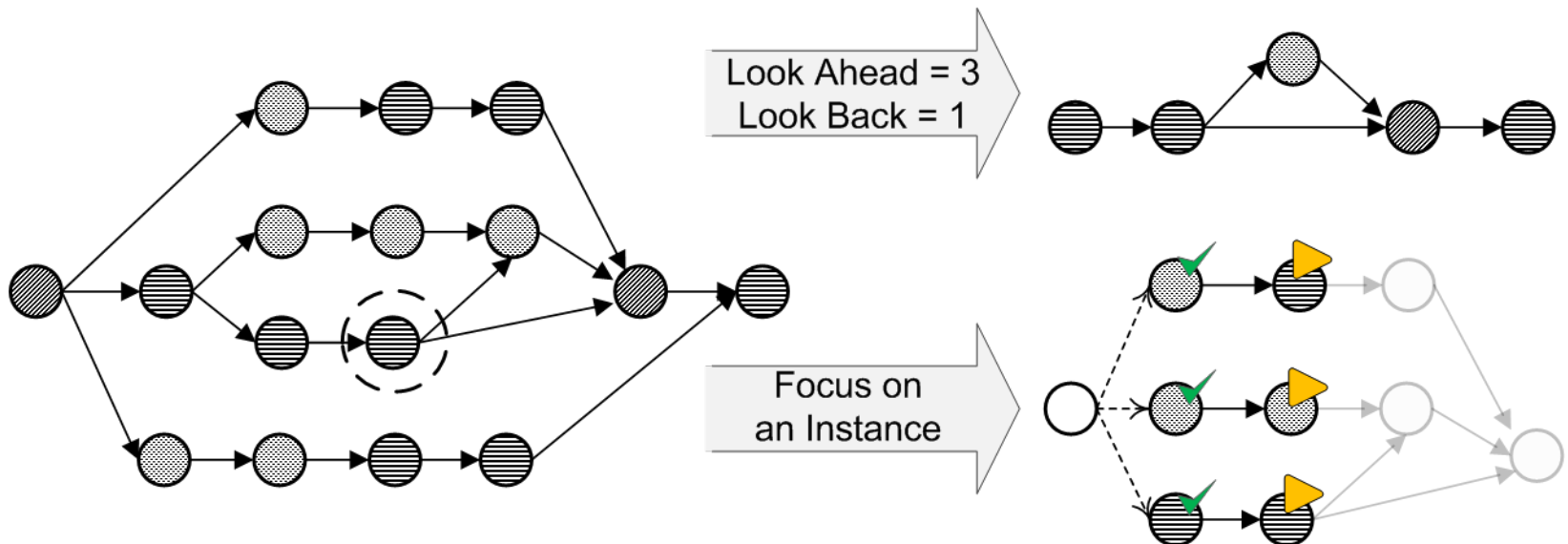
Abstraction from Details

- Multiple activities are serving only one single purpose
- Their aggregation can be ...
 - User-driven, e.g. specifying which structures to aggregate
 - Patterns can be defined to describe higher level structures
 - Precise views can thereby be created automatically
- Abstracting known structures into meaningful aggregates
- Shapes for such aggregates can be automatically assigned



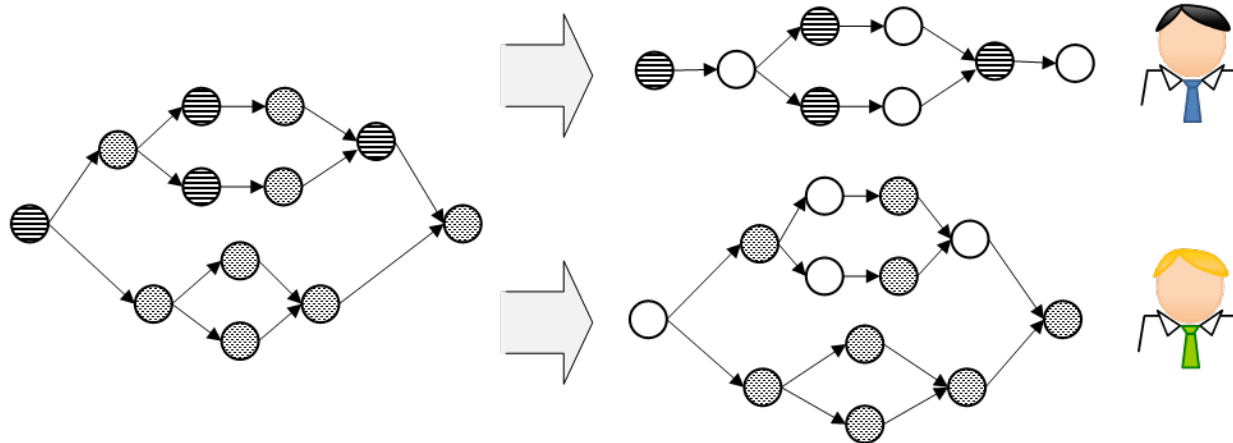
Set a Focus

- Set a focus
 - Process models quickly become complex and “crowded”
 - It is helpful to focus on one or more *points of interest*
- Solution
 - Omission of all nodes that are out of the focus
 - On a process model
 - On a process instance



Access Control on the Process Model

- Access control on the process model
 - People with particular roles are only allowed to see certain parts of a process
 - The rest shall be invisible or obfuscated (and locked)
- Solution
 - Human-assisted augmentation with access control information about roles and access rights
 - Disambiguation has to be applied for cases, where updates of the originating process shall be performed (problem: visibility)
 - Straightforward solution
 - Abstraction of all “invisible/locked” artifacts
 - Show that something happens while hiding what exactly happens



Conclusion and Outlook

Conclusion

■ General conclusion

- Process views have the potential to increase the efficiency of BPM dramatically
- Views transformations can be concatenated
- Analysis support versus modelling support

■ Outlook

- Traditional role of modelling tool is going to change into a powerful and flexible platform
- Augmentation with semantics has manifold benefits

End of Document