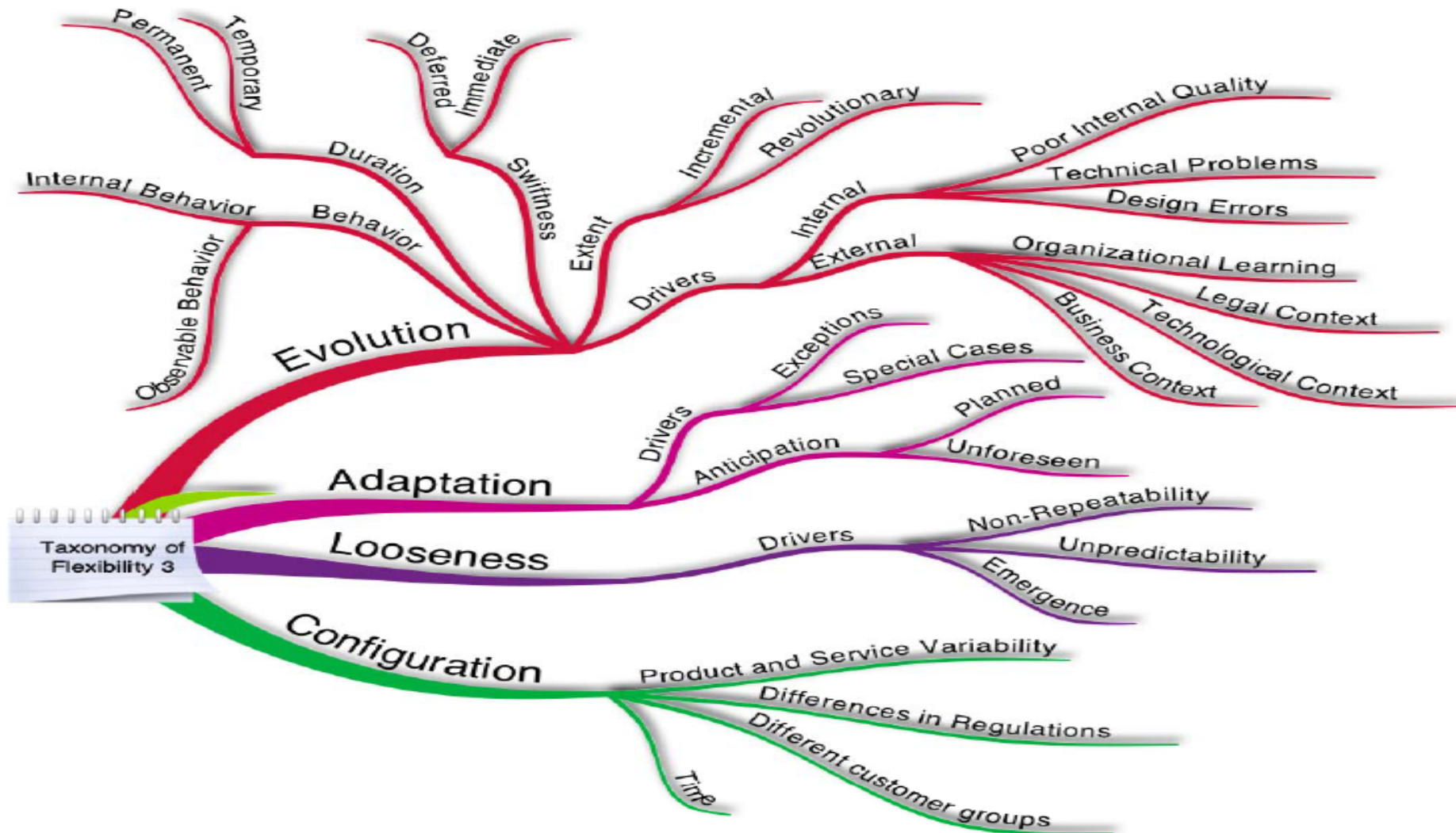




Flexibility Need

Business Process Management and Flexibility
Barbara Re, Phd

Taxonomy of Flexibility Needs



Variability

- Variability is typical for many domains and requires that processes are handled differently depending on the particular context
- Drivers
 - Product and service variability
 - Differences in regulations
 - Different customer groups
 - Temporal differences

Example: Vehicle
Repair



Looseness

- Knowledge-intensive processes cannot be fully pre-specified, but require loose specifications
- Drivers
 - Unpredictability
 - Non-Repeatability
 - Emergence

Example: Treatment
Processes in a Hospital



Adaptation

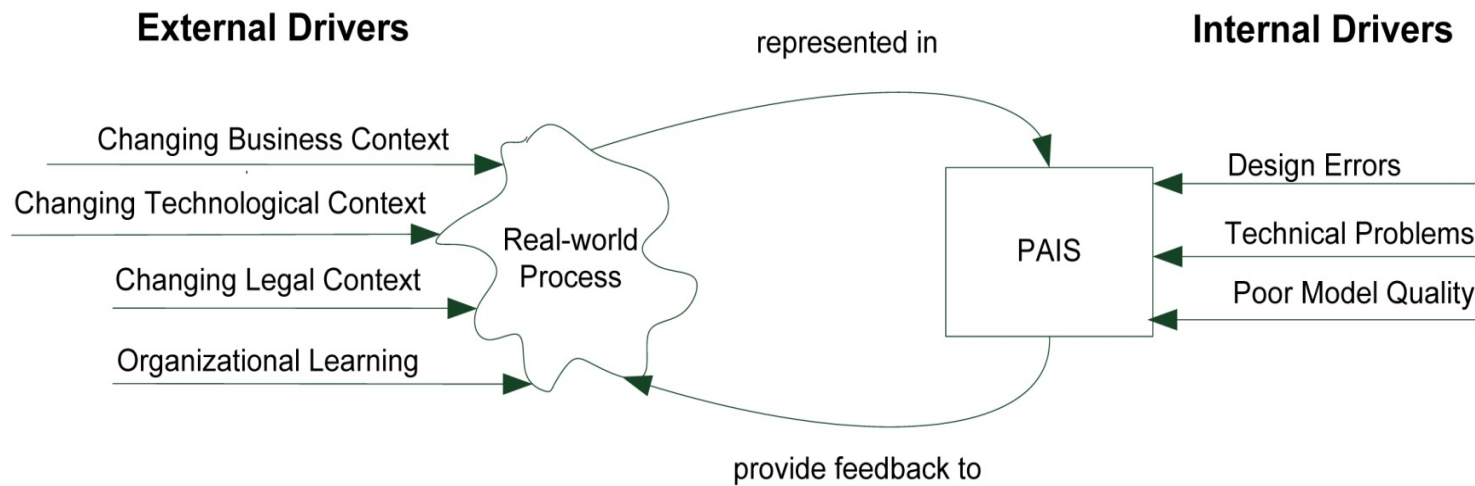
- Ability to adapt the process and its structure to temporary events
- Drivers
 - Special Situations
 - Exceptions
- Anticipation of Adaptation
 - Planned
 - Unanticipated

Example: Examination
Procedures in a Hospital



Evolution

- Ability of the implemented process to change when the business process evolves



Evolution

- Extent of Evolution
 - Incremental
 - Continuous Process Improvement
 - Revolutionary
 - Business Process Reengineering
- Duration
 - Temporary
 - Permanent



Evolution

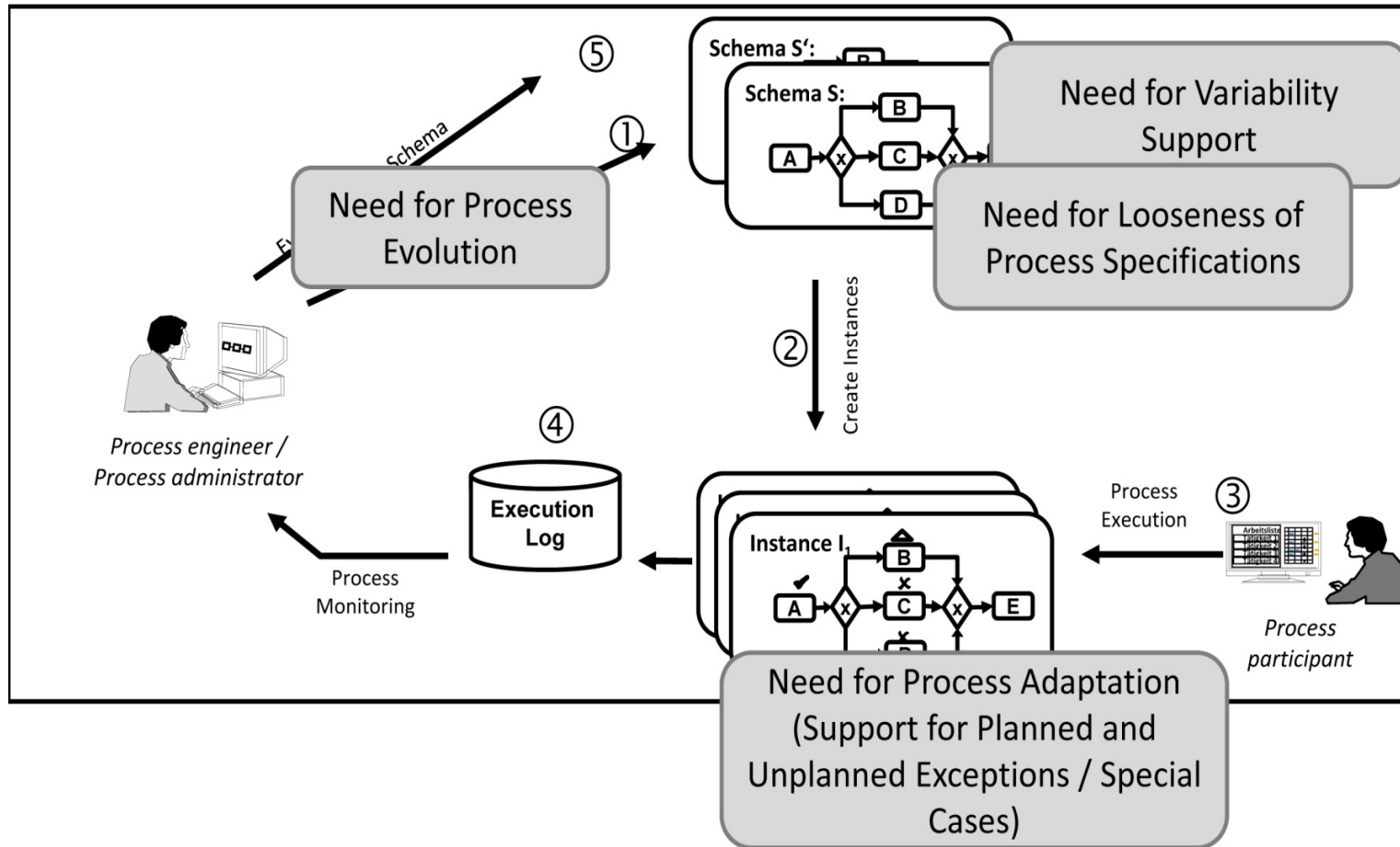
- Swiftiness
 - Deferred
 - Ongoing instances are not affected
 - Immediate
 - Ongoing instances are affected
- Visibility
 - Observable Behavior
 - Internal Structure

Example: Tender
Preparation

Example: Inconsistent
Naming of Process Models



Flexibility Issues along the Process Lifecycle



Flexibility Needs and Technological Requirements

Flexibility Need	Dimension	Technological Requirement
Variability		Configuration
Looseness		Loosely-specified Processes
Adaptation	Planned	Exception Handling
	Unplanned	Ad-hoc Changes
Evolution	Deferred Evolution,	Versioning
	Immediate Evolution,	Process Instance Migration
	Poor Internal Quality,	Refactoring
	Organizational Learning	Monitoring, Analysis and Mining





Fundamental Features for Flexible PAIS

- ▶ **Accountability and traceability:** organizations are required to comply with a wide range of regulations
 - ▶ **Business compliance:** compliance with existing rules and regulations is another fundamental issue
 - ▶ **Access control:** the application of changes at the process type as well as the process instance level must be restricted to authorized users
 - ▶ **Correctness of change:** When adapting or evolving business processes—potentially in the midst of their execution—it has to be ensured that changes are performed in a controlled manner and do not lead to run-time errors
 - ▶ **User support:** While traditional PAISs provide little maneuvering room for their users, loosely specified processes require many decisions to be made along the way and therefore require significantly more user experience
 - ▶ **Need for learning from instance executions:** reuse of previously conducted process instances or the discovery of frequently occurring similar process fragments should be supported
 - ▶ **Concurrency of changes:** PAISs need to handle situations in which instance-specific adaptations (i.e., ad hoc changes) and evolutionary changes overlap, this is especially important when evolution has to be immediate and not deferred
-





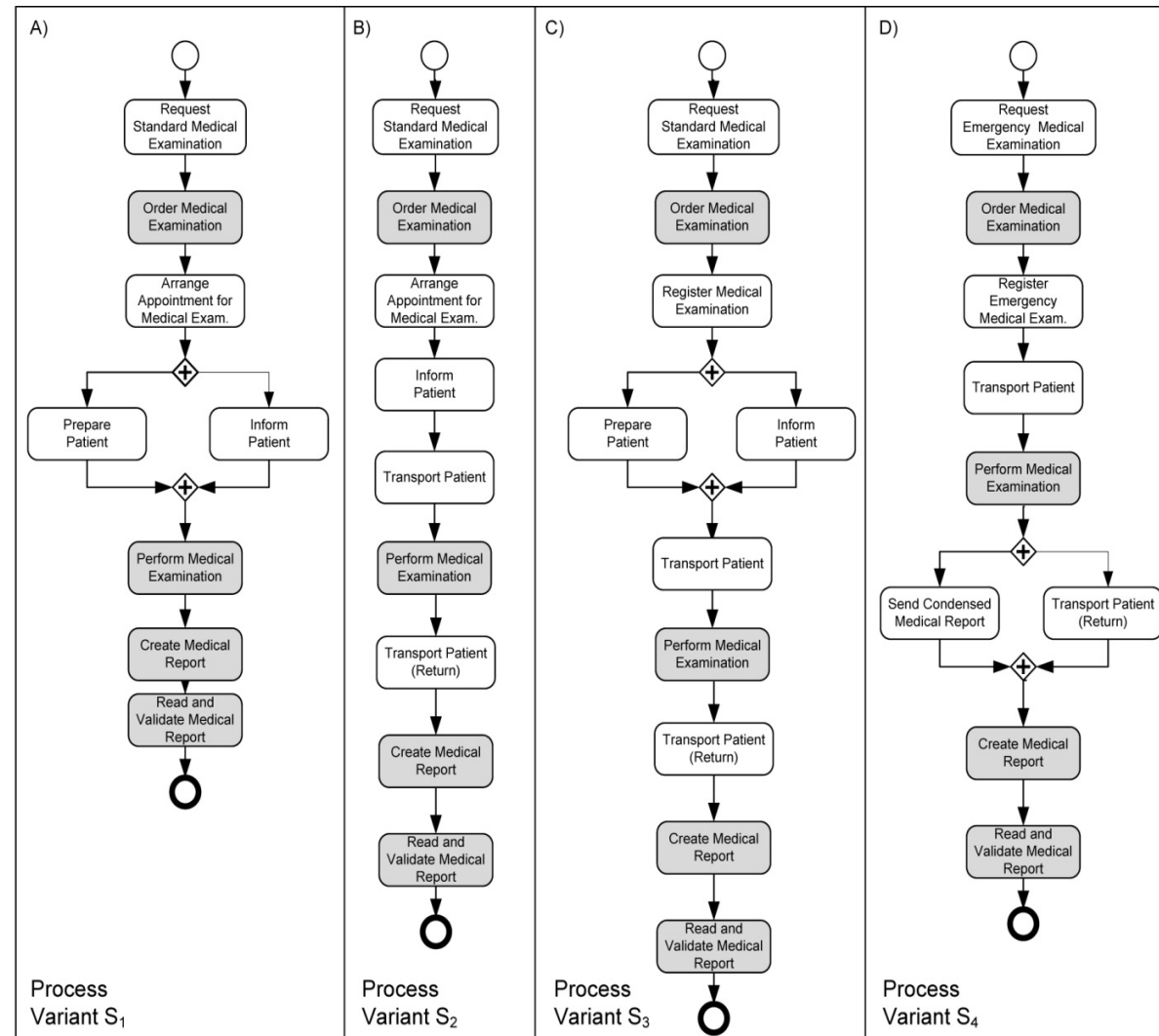
Capturing Variability in Pre-specified Process Models

Advanced Topic in Business Process Management
Barbara Re, Phd

Motivation – Handling Medical Examinations

Variety of related variants

- Same business objective
- Commonalities
- Differences due to varying application context





Two Main Approaches

- ▶ Behaviour-based Approaches for Capturing Process Variability
- ▶ Structural Approaches for Capturing Process Variability



Two Main Approaches

- ▶ Behaviour-based Approaches for Capturing Process Variability
- ▶ Structural Approaches for Capturing Process Variability



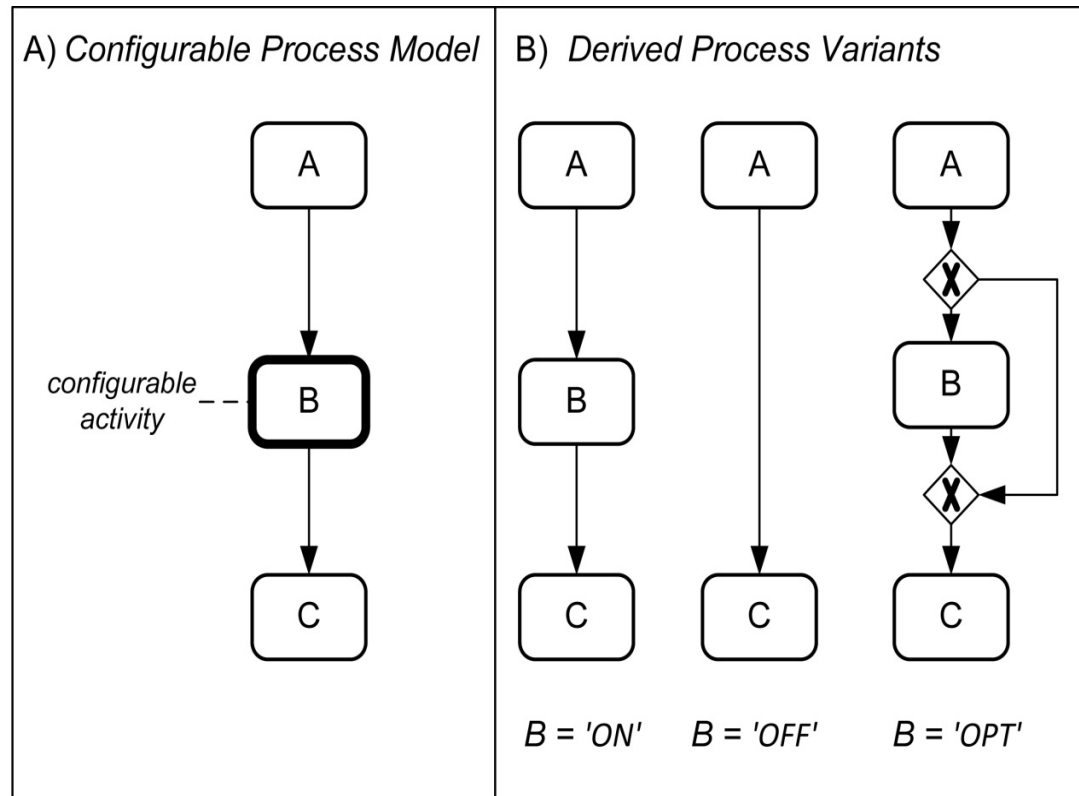
Configurable Nodes

- ▶ Main idea: configurable nodes
 - ▶ Extension of an existing process modeling language by adding configurable elements
 - ▶ E.g., C-EPC, E-YAWL
- ▶ Configurable nodes represent variation points and can be associated with configuration alternatives
- ▶ Possible combinations of configuration alternatives can be restricted through constraints



Configurable Activities

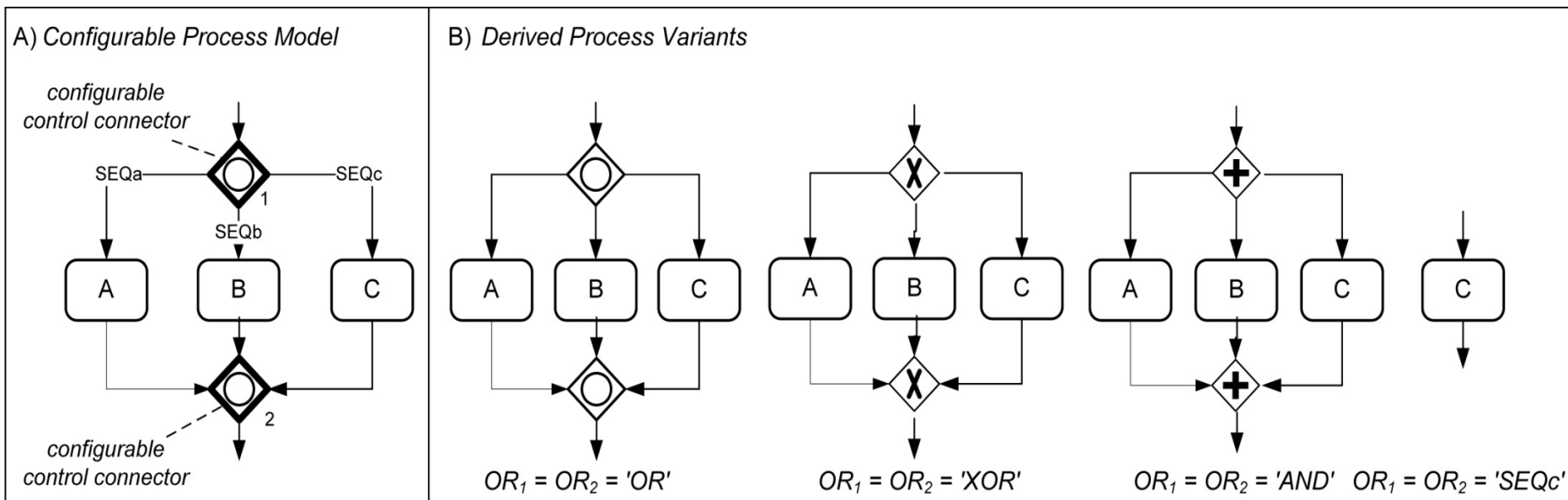
- ▶ Included (ON)
- ▶ Excluded (OFF)
- ▶ Conditional (OPT)



Configurable Control Connectors

- ▶ Configurable OR
- ▶ Configurable XOR
- ▶ Configurable AND

Can be configured to a connector equally restrictive or less restrictive





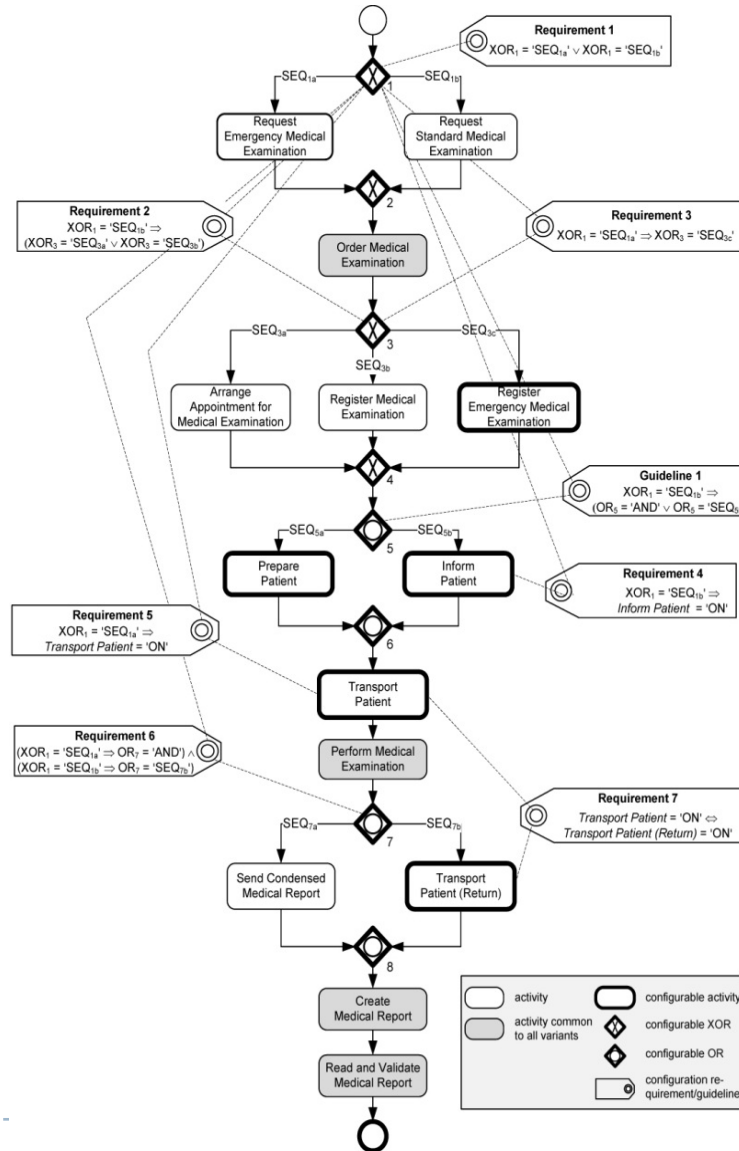
Configuration Requirements and Guidelines

- ▶ Requirements
 - ▶ Define constraints over the configuration alternatives that can be chosen

- ▶ Guidelines
 - ▶ Do not prescribe mandatory constraints, but serve as recommendations



Configurable Model: Handling Medical Examinations



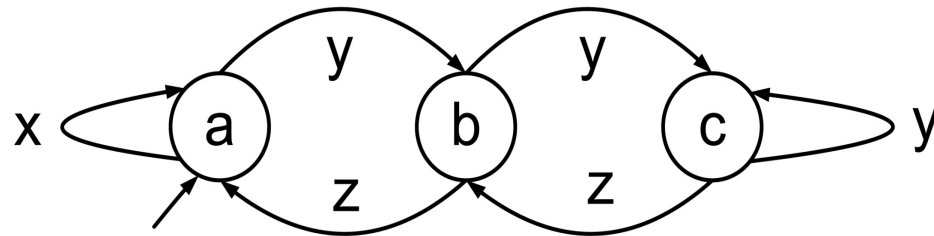
M. Reichert, B. Weber:
Enabling Flexibility in Process-Aware Information Systems,
© Springer-Verlag Berlin Heidelberg 2012

Configurable Model: Handling Medical Examinations

	Settings of Configurable Connectors								Settings of Configurable Activities				
	XOR1	XOR2	XOR3	XOR4	OR5	OR6	OR7	OR8	Register Emergency Medical Examination	Prepare Patient	Inform Patient	Transport Patient	Transport Patient (Return)
Process variant S1	SEQ1b	SEQ1b	SEQ3a	SEQ3a	AND	AND	SEQ7b	SEQ7b	OFF	ON	ON	OFF	OFF
Process variant S2	SEQ1b	SEQ1b	SEQ3a	SEQ3a	SEQ5b	SEQ5b	SEQ7b	SEQ7b	OFF	OFF	ON	ON	ON
Process variant S3	SEQ1b	SEQ1b	SEQ3b	SEQ3b	AND	AND	SEQ7b	SEQ7b	OFF	ON	ON	ON	ON
Process variant S4	SEQ1a	SEQ1a	SEQ3c	SEQ3c	SEQ5b	SEQ5b	AND	AND	ON	OFF	OFF	ON	ON

Hiding and Blocking

- ▶ Capturing variability in a language-independent way
 - ▶ (1) Represent process model as a labeled transition system (LTS)



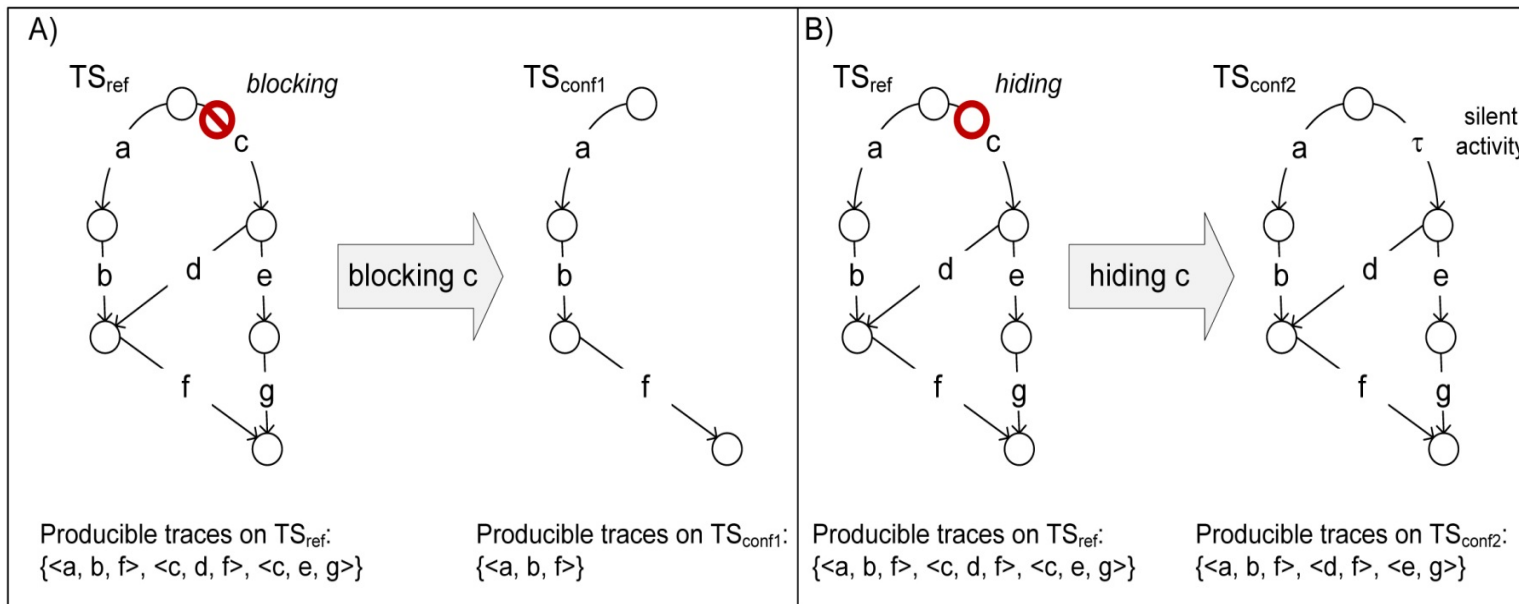
A. Kretschmer, A. Maier:
Evaluating Feasibility in Process Algebras: Information Systems,
© Springer-Verlag Berlin Heidelberg 2012

- ▶ (2) Apply hiding and blocking operators for configuring LTL-based reference process models
-

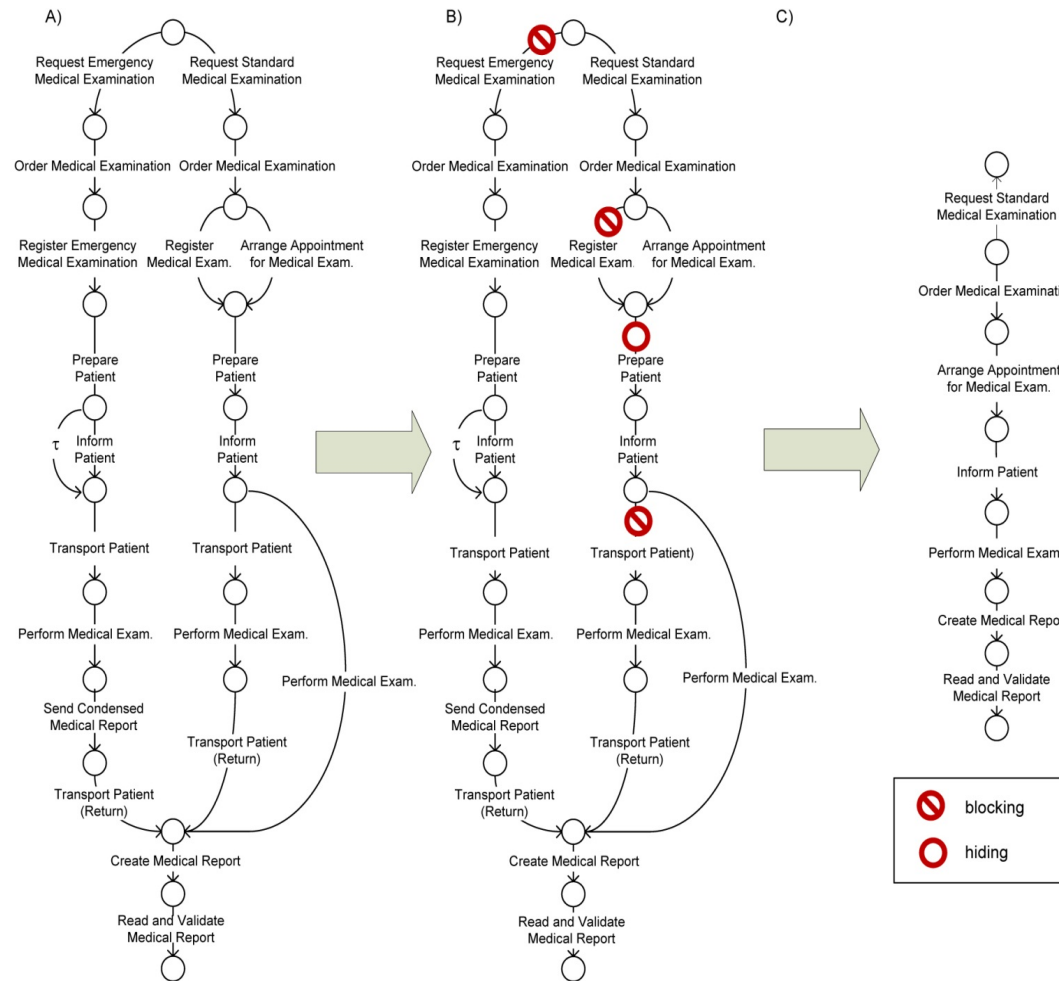


Hiding and Blocking

- ▶ **Blocking:** enables encapsulation – execution of an atomic activity (event is disabled)
- ▶ **Hiding:** enables abstraction – execution of an event becomes non-observable (activity is replaced by silent activity)



Hiding and Blocking Example



M. Reichert, B. Weber:
 Enabling Flexibility in Process-Aware Information Systems.
 © Springer-Verlag Berlin Heidelberg 2012

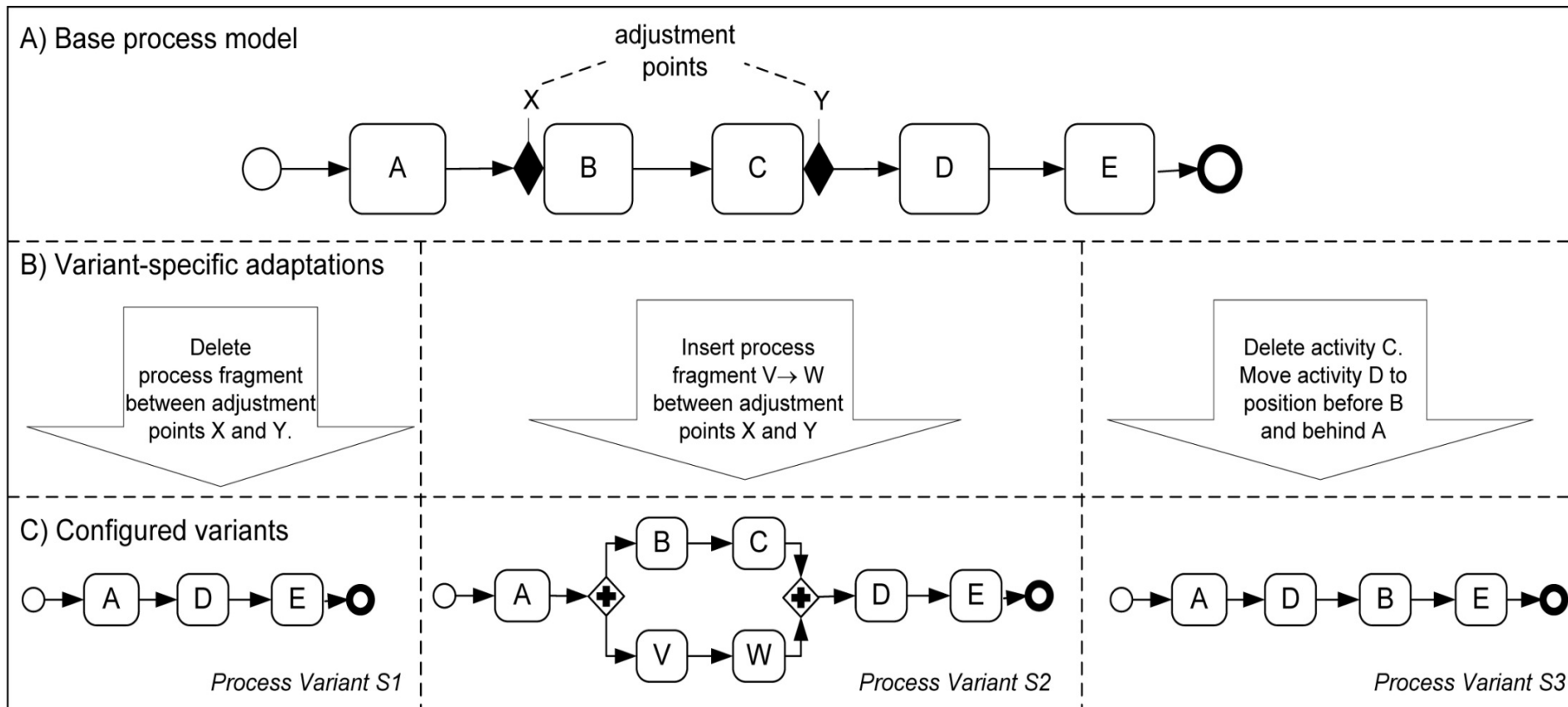


Two Main Approaches

- ▶ Behavior-based Approaches for Capturing Process Variability
- ▶ Structural Approaches for Capturing Process Variability



Deriving Variants through Structurally Changing a Base Process Model






Representing a Process Family

- ▶ Through a configurable base process model
 - ▶ Policy 1: Standard Process
 - ▶ Policy 2: Most frequently used process
 - ▶ Policy 3: Superset of all process variants
 - ▶ Policy 4: Intersection of all process variants
- ▶ and a related set of pre-specified changes
 - ▶ Adjustment points
 - ▶ Change options (i.e., a grouping of change operations)

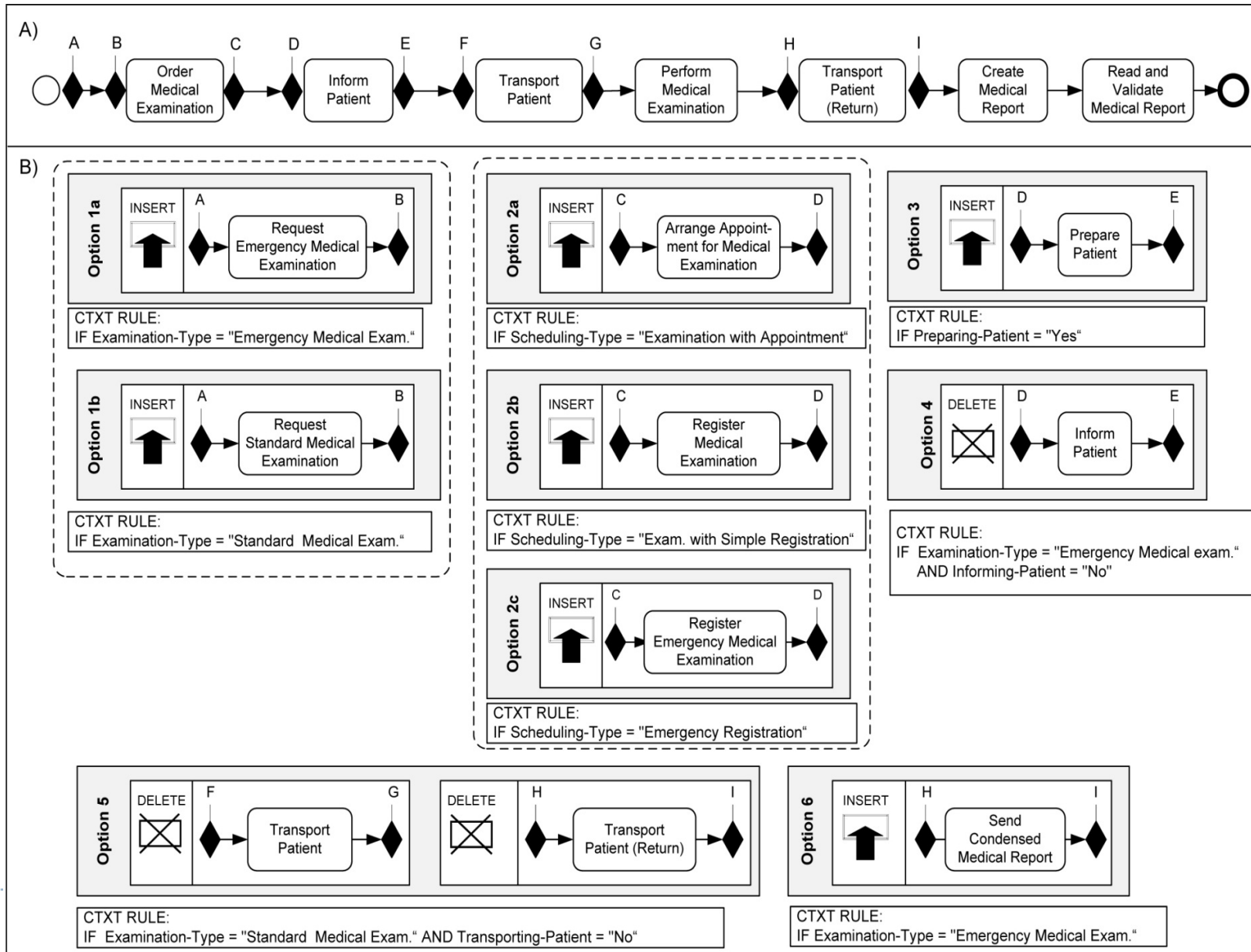


Examples of Change Operations

1. INSERT-Operation	
Symbol	
Purpose	Adding a <i>process fragment</i> (e.g., a single activity or an activity sequence).
Parameters	<ul style="list-style-type: none"> • Process fragment to be added • Target position of the process fragment to be added in the base process, specified in terms of adjustment points
2. DELETE-Operation	
Symbol	
Purpose	Removing a <i>process fragment</i>
Parameters	<ul style="list-style-type: none"> • Process fragment to be deleted with entries and exits being marked by adjustment points • Alternatively: deleting single activities by referring to their ID
3. MOVE-Operation	
Symbol	
Purpose	Changing the execution order of activities
Parameters	<ul style="list-style-type: none"> • Process fragment to be moved with entries and exits being marked by adjustment points • Target position of the process fragment to be moved, specified in terms of adjustment points

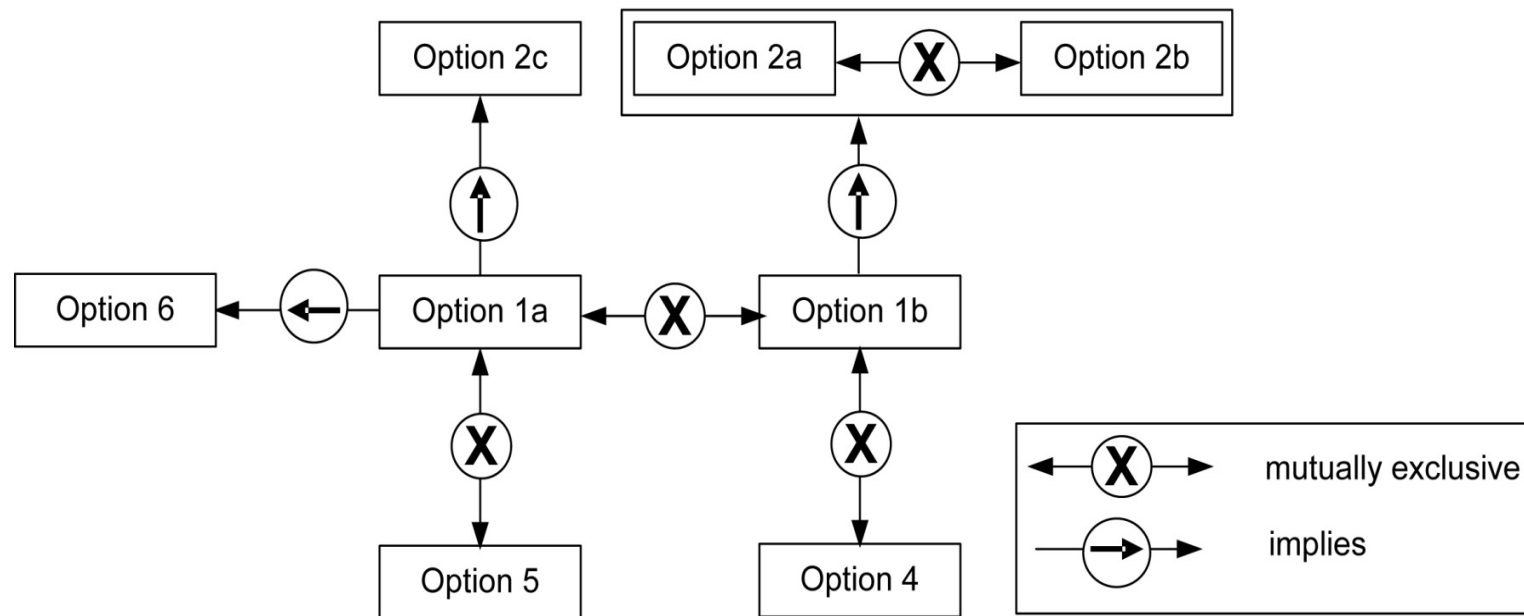


Example of Base Process + Options



Constraining Allowed Combinations of Change Operations

- ▶ Implications
- ▶ Mutual exclusion
- ▶ Hierarchy



Context Model

Context-specific selection of change operations

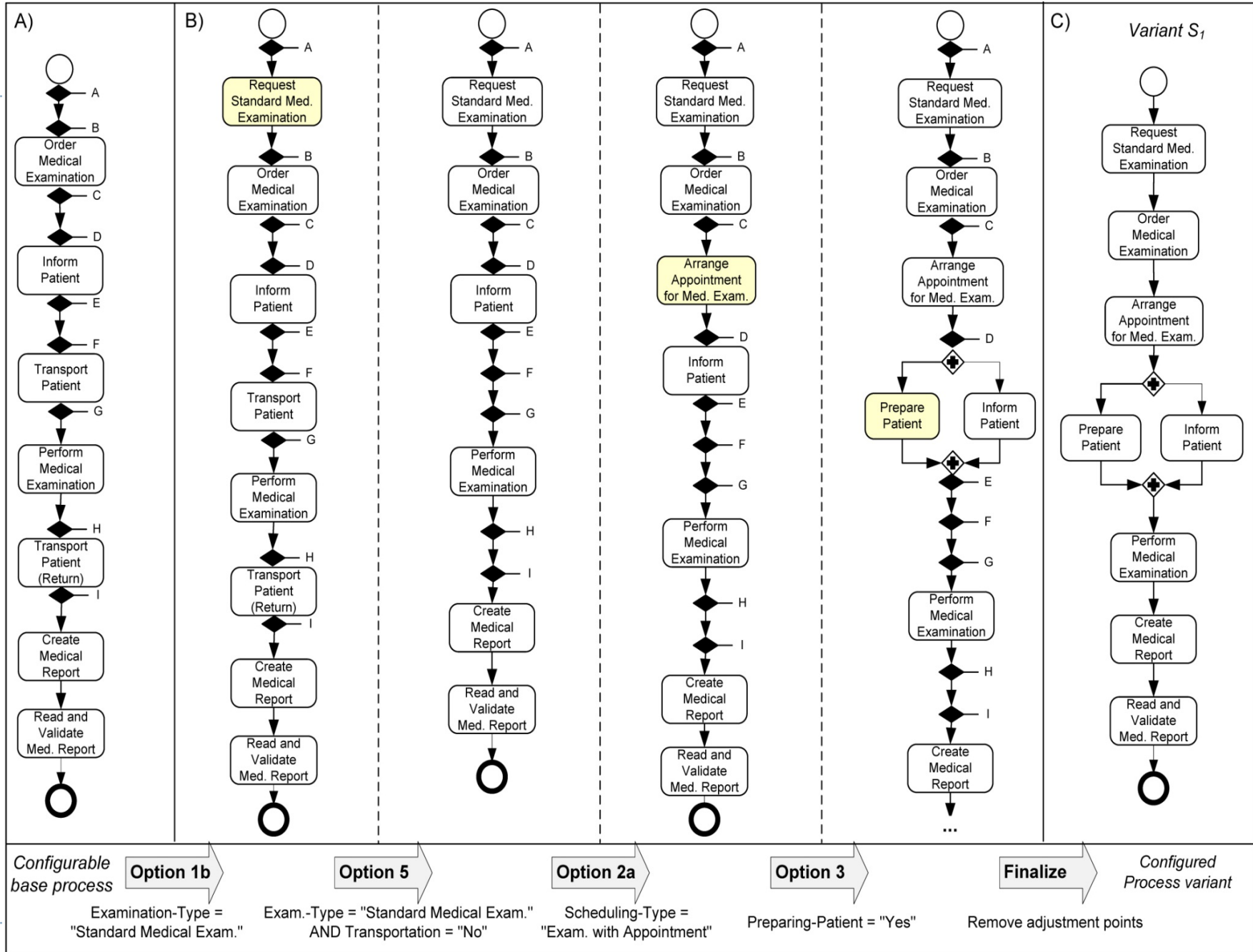
Context Variable	Value Range
Examination-Type	Standard Medical Exam., Emergency Medical Exam
Scheduling-Type	Examination with Appointment, Examination with Simple Registration, Emergency Registration
Preparing-Patient	Yes, No
Informing-Patient	Yes, No
Transporting-Patient	Yes, No



The Whole Approach

- ▶ (1) Select relevant options
 - ▶ All change options whose context rules evaluate to true are selected
- ▶ (2) Ensure compliance of the selected options with option constraints
 - ▶ Compliance with option constraints has to be checked
- ▶ (3) Determine the order in which options shall be applied
- ▶ (4) Configuring the base process by applying the selected options and their change operations to it





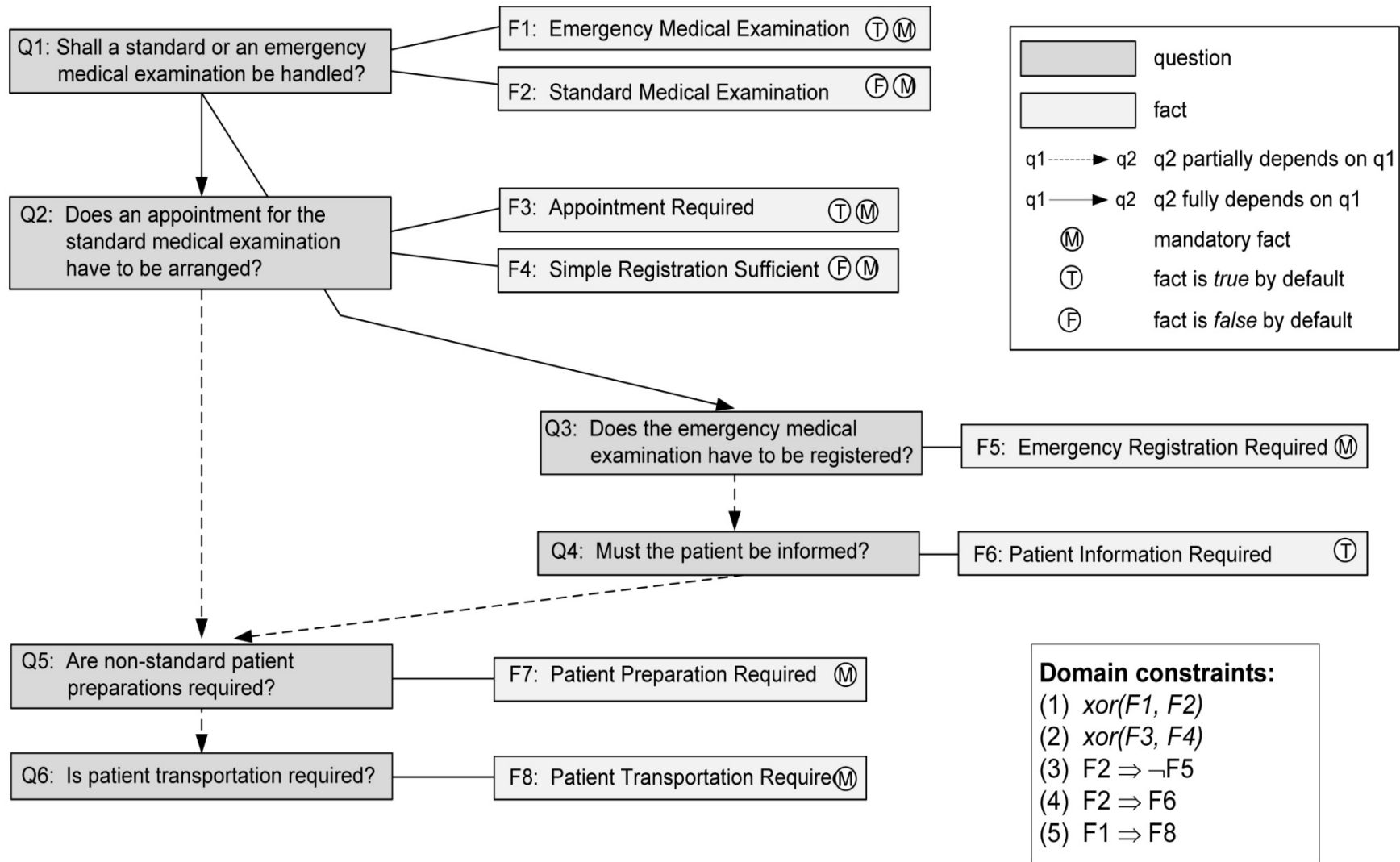


Questionnaire-driven Process Configuration

- ▶ (1) Questionnaire Model
- ▶ (2) Using Questionnaire Models for Configuring a Reference Process Model
 - ▶ (a) Linking Domain Facts and Configurable Activities
 - ▶ (b) Linking Domain Facts and Configurable Connectors



Questionnaire Model





Linking Domain Facts and Configurable Activities

Configurable Activity	Configuration Alternative	Boolean Expression Over Facts
<i>Register Emergency Medical Examination</i>	ON	$F1 \wedge F5$
<i>Prepare Patient</i>	ON	F7
<i>Inform Patient</i>	ON	F6
<i>Transport Patient (...)</i>	ON	F8

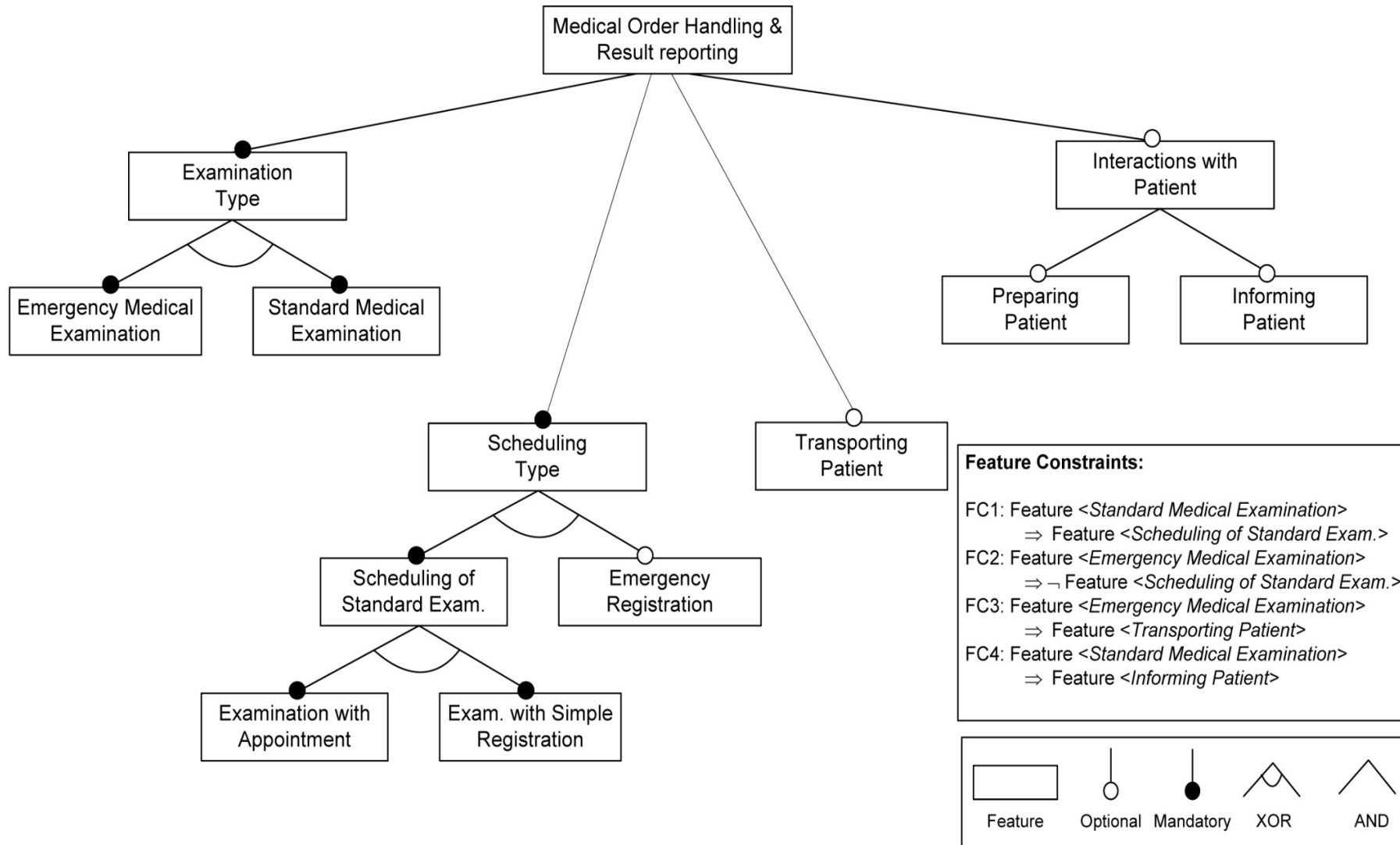




Linking Domain Facts and Configurable Connectors

Configurable Connector	Configuration Alternative	Boolean Expression over Facts
XOR ₁	SEQ _{1a}	F1
	SEQ _{1b}	F2
XOR ₃	SEQ _{3a}	F2 \wedge F3
	SEQ _{3b}	F2 \wedge F4
	SEQ _{3c}	F1 \wedge F5
OR ₅	AND	F6 \wedge F7
	SEQ _{5a}	F7 \wedge \neg F6
	SEQ _{5b}	F6 \wedge \neg F7
	OR	<i>false</i>
	XOR	<i>false</i>
OR ₇	AND	F1
	SEQ _{7a}	<i>false</i>
	SEQ _{7b}	F2
	OR	<i>false</i>
	XOR	<i>false</i>

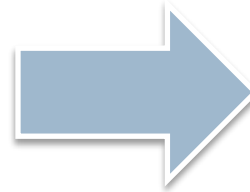
Feature Diagram



Modeling Variability in BP

- ▶ A number of approaches to support variability modeling have emerged over the past decade

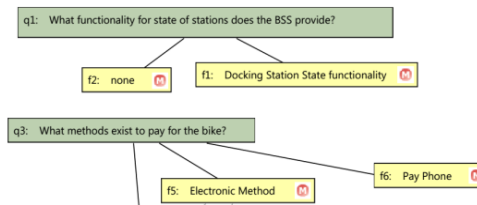
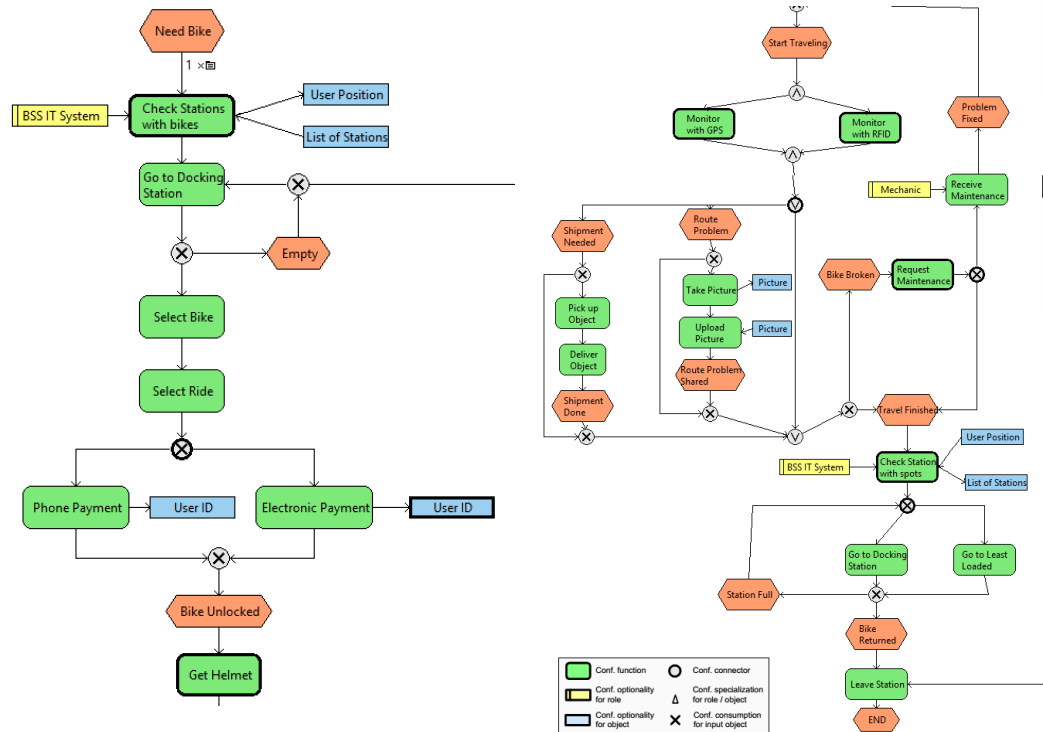
We selected five and used them to model a case extracted from the *smart city scenario* characterized for sustaining multiple variants.



- C-iEPC
- C-YAWL
- PESOA
- Provop
- vBPMN

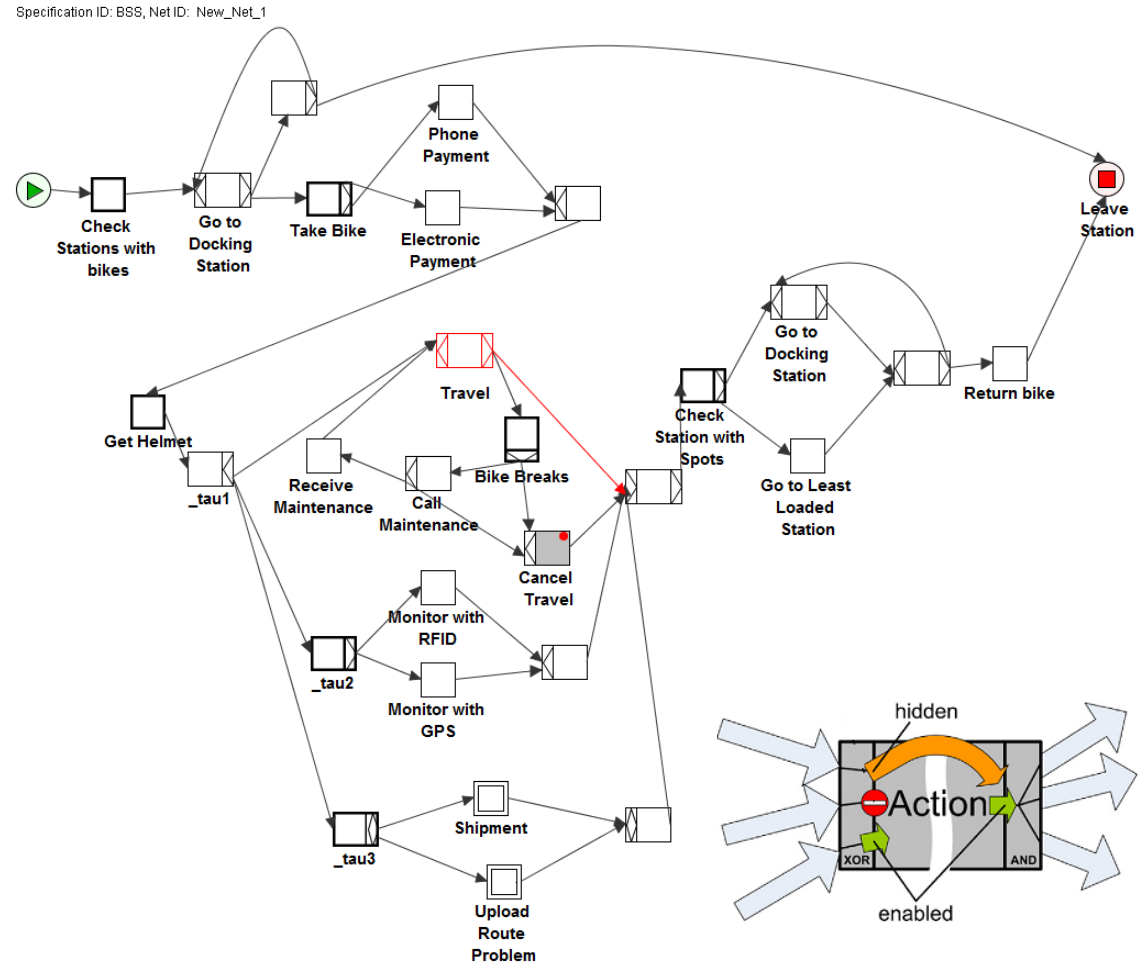
PART I: Modeling Variability in BP Configurable integrated EPC (C-iEPC)

- ▶ Customization is achieved by **restriction**
- ▶ Customizes control flow, resources and object perspective
- ▶ Uses an extension of **EPC**.
- ▶ **Configurable Nodes** include functions, data, roles and connectors
- ▶ Customization is done by means of **questionnaire models**



PART I: Modeling Variability in BP Configurable YAWL (C-YAWL)

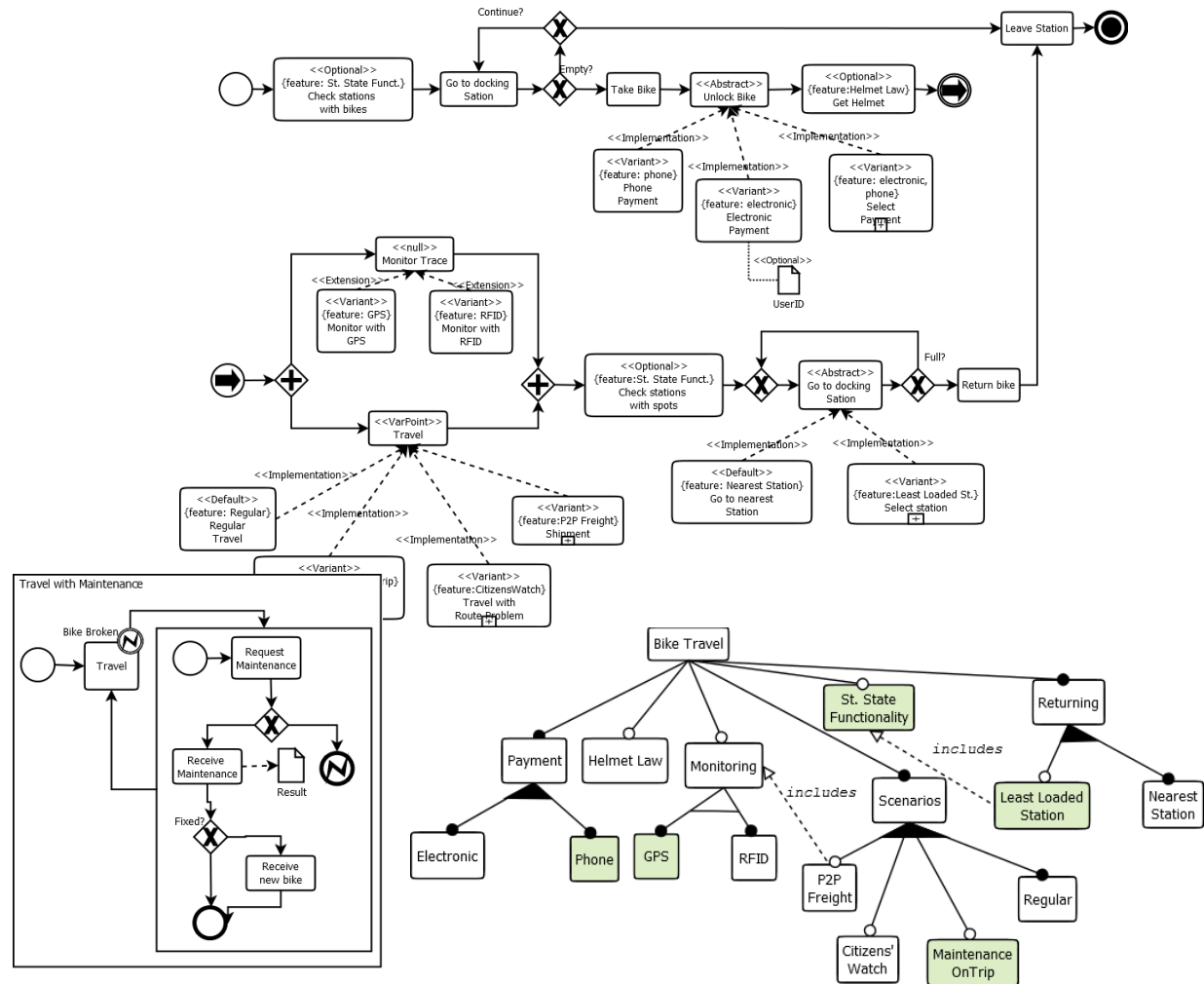
- ▶ Customization is achieved by **restriction**
- ▶ Customizes only **control flow** perspective
- ▶ Uses an extension of **YAWL**
- ▶ **Configurable Nodes** include a task and ports located at each task's split or join can be. Can be hidden or blocked.
- ▶ Customization is done by means of **questionnaire models**.



PART I: Modeling Variability in BP

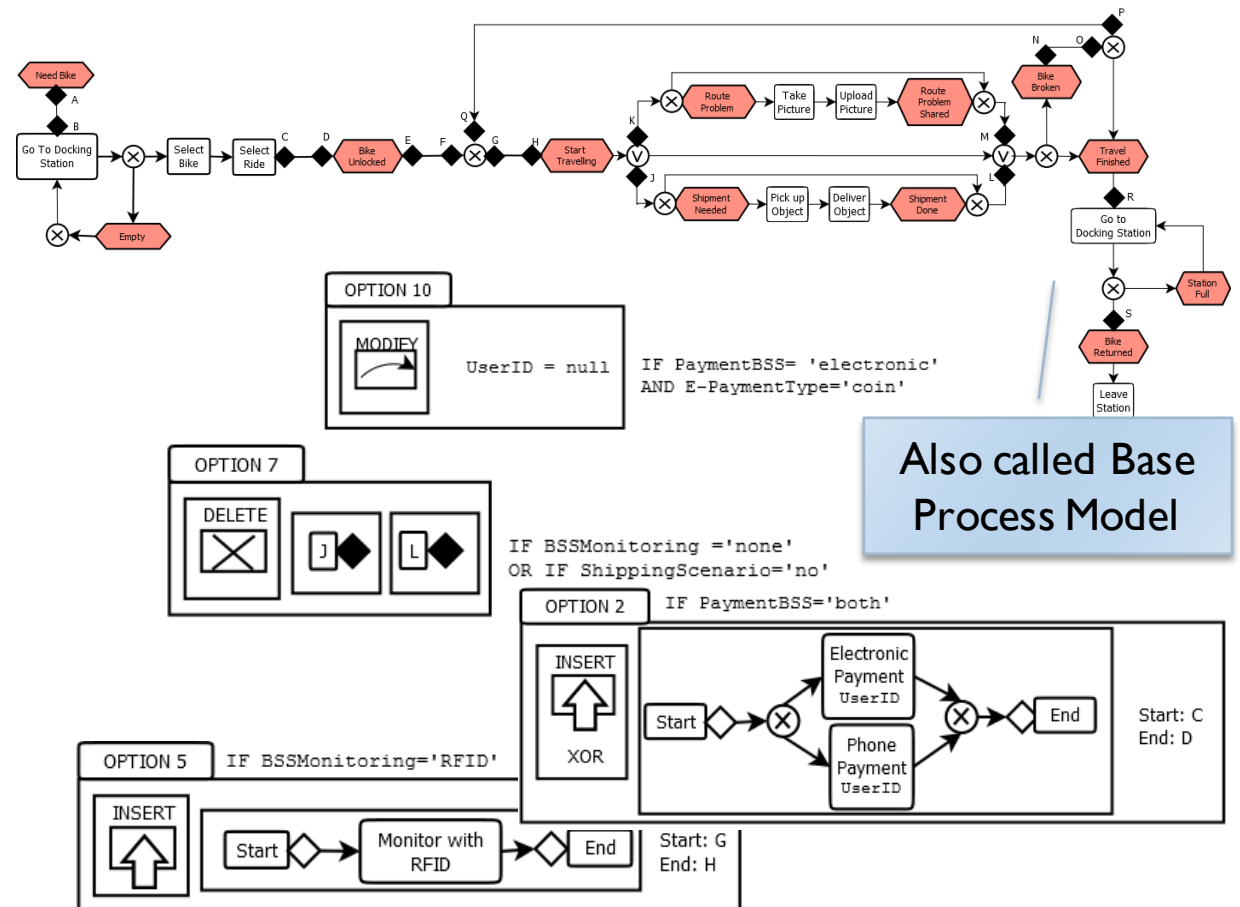
Process family Engineering in SOA (PESOA)

- ▶ Customization is achieved by **restriction**
- ▶ Customizes **control flow** and **object** perspective
- ▶ Uses an extension of **BPMN 2.0** or **UMLADs**
- ▶ Uses **stereotype annotations** applied on the process model elements
- ▶ Customization is done by means of **feature models**



PART I: Modeling Variability in BP Process Variants by Options (Provop)

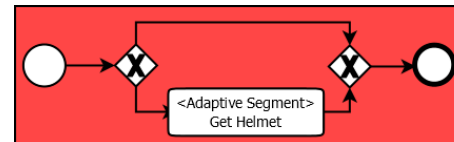
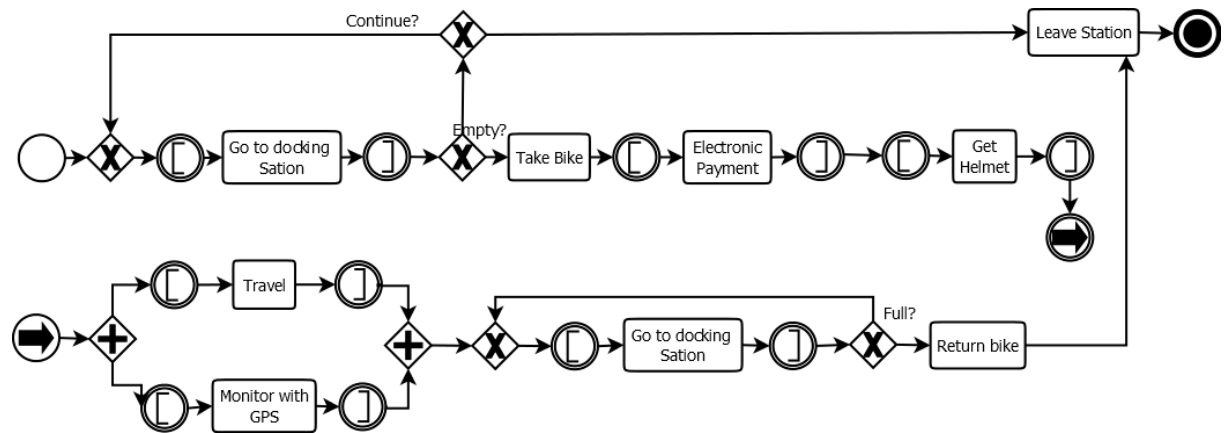
- ▶ Customization is achieved by **restriction and extension**
- ▶ Customizes **control flow, resource and object perspective**
- ▶ Language-independent
- ▶ Applies change operations (INSERT, DELETE, MOVE and MODIFY)
- ▶ Customization is done by means of **operations** applied according to **context rules** in segments bounded with **adjustment points**



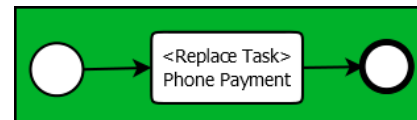
PART I: Modeling Variability in BP

Variant BPMN (vBPMN)

- ▶ Customization is achieved by **restriction and extension**
- ▶ Customizes only **control flow perspective**
- ▶ Extension of **BPMN 2.0**
- ▶ Applies **adaptation, exception handling and time patterns**
- ▶ Customization is done by means of **patterns applied according to adaptation rules in adaptive segments**



```
Adaptation Rule #4
ON GetHelmet_entry
IF HelmetLaw = 'no'
THEN APPLY skip (segment= 'GetHelmet')
```



```
Adaptation Rule #2
ON ElectronicPayment_entry
IF PaymentBSS = 'phone'
THEN APPLY replace (segment= 'ElectronicPayment
task= 'PhonePayment')
```

