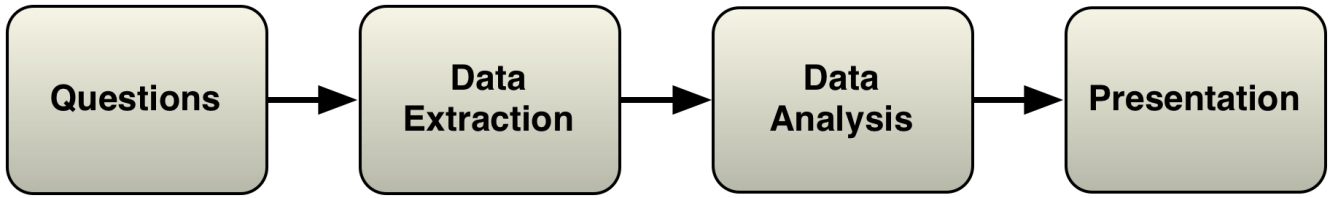
Outline

1. Introduction & Motivation
2. Analyzing and capturing business processes
3. Modeling Business Processes
4. Business Process Management
5. Case Management & Decision Management
6. Business Process Mining
7. Business Process Automation

Outline

1. Introduction & Motivation
2. Analyzing and capturing business processes
3. Modeling Business Processes
4. Business Process Management
5. Case Management & Decision Management
6. Business Process Mining
7. Business Process Automation

Roadmap



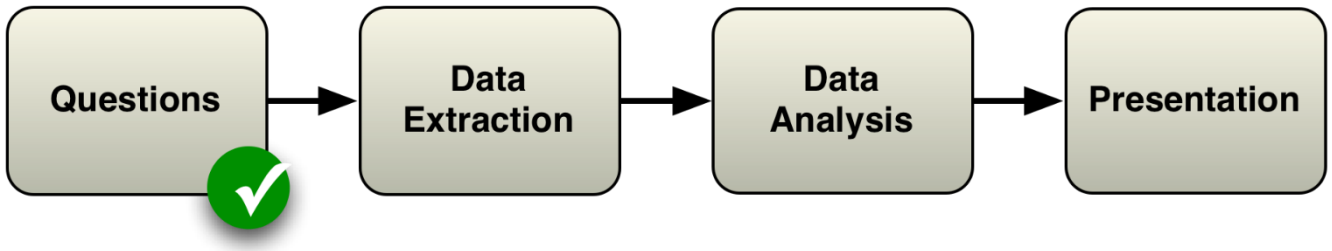
- Determine questions
- Process scope
- Which IT systems

- Via DB administrator
- CSV file or database extract

- Extract ‘As-is’ process
- Answer questions

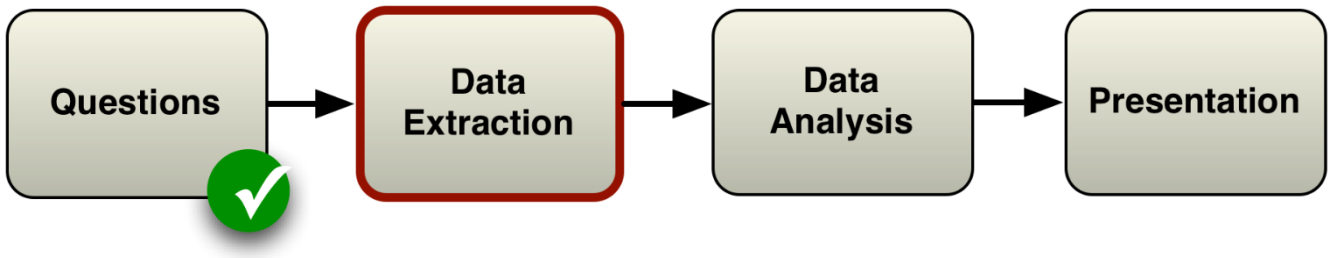
- Present results (e.g., report, presentation, workshop etc.)

Roadmap



1. How does the process actually look like?
2. Are there deviations from the prescribed process?
3. Do we meet the performance targets?

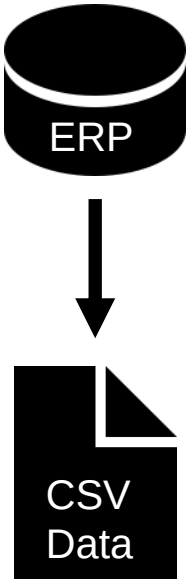
Roadmap



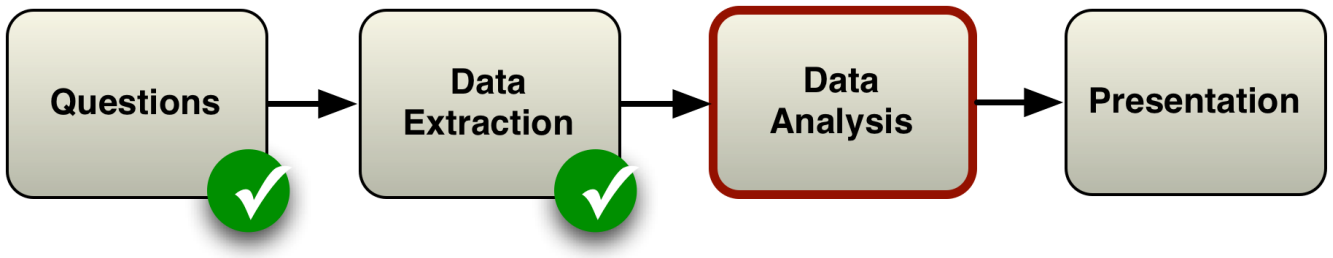
Data Extraction

fluxicon

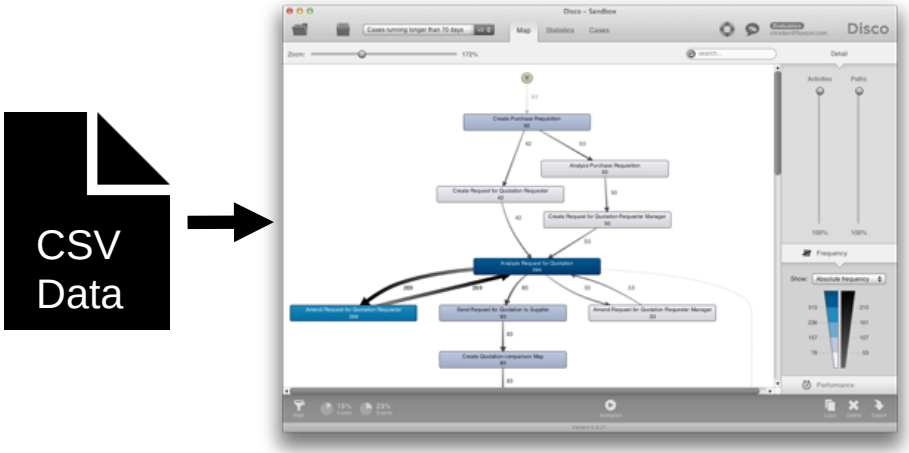
- IT staff extracts history logs from the ERP system
- CSV file is starting point for our session



Roadmap

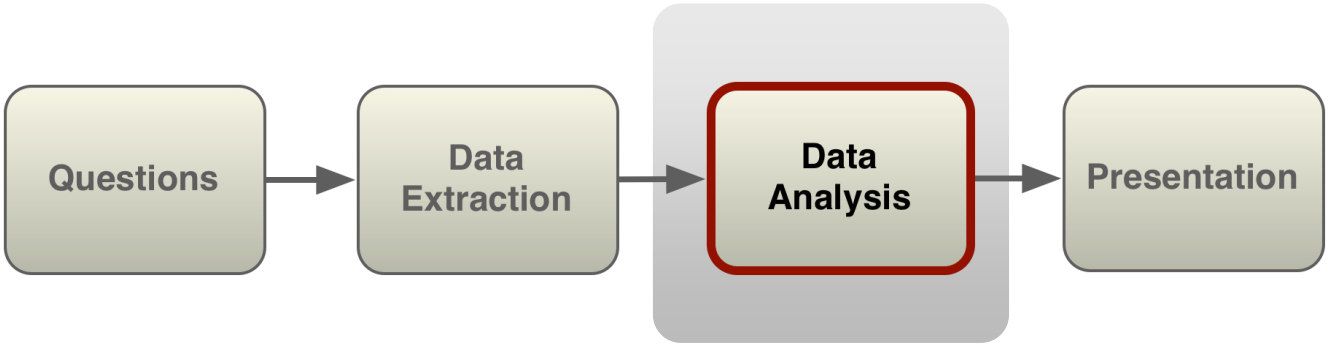


Data Analysis



Download :fluxicon.com/disco

Roadmap



Process Mining am Beispiel – Ereignis Log

Case	Prozess
1	<a, b, g, h>
2	<a, c, d, e, f, g, h>
3	<a, c, d, e, f, g, h>
4	<a, c, e, d, f, g, h>
5	<a, c, d, e, f, g, h>
6	<a, b, g, h>
7	<a, b, g, h>
8	<a, c, d, e, f, g, h>
9	<a, b, g, h>
10	<a, c, e, d, f, g, h>
:	:

Process Mining am Beispiel – Ereignis Log zum Footprint

Die Beziehungen der Aktivitäten zueinander werden in einer Matrix, einem sogenannten Footprint, dargestellt. Dabei gilt, dass $a, b \in \mathcal{A}$ wobei:

(1) $a >_L b$, wobei es einen Pfad $\sigma = \langle t_1, t_2, t_3, \dots, t_n \rangle$ und $i \in \{1, 2, \dots, n-1\}$ mit $\sigma \in L$ *und* $t_i = a$ und $t_{i+1} = b$ geben muss

(2) $a \rightarrow_L b$, wobei $a >_L b$ und $b \not> _L a$ gelten muss

(3) $a \#_L b$, wobei $a \not> _L b$ und $b \not> _L a$ gelten muss

(4) $a \|_L b$, wobei $a >_L b$ und $b >_L a$ gelten muss

Process Mining am Beispiel – Ereignis Log zum Footprint (2)

- $\#L_1$: zwischen den Aktivitäten existiert keine Beziehung zueinander
- $\rightarrow L_1$: auf die Aktivität a folgt direkt die Aktivität b
- $\leftarrow L_1$: die Aktivität b ist ein Vorgänger von a
- $||L_1$: die Aktivität a und b verlaufen parallel

von \ nach	a	b	c	d	e	f	g	h
a		$\rightarrow L_1$	$\rightarrow L_1$					
b	$\leftarrow L_1$						$\rightarrow L_1$	
c	$\leftarrow L_1$			$\rightarrow L_1$	$\rightarrow L_1$			
d			$\leftarrow L_1$			$\rightarrow L_1$		
e			$\leftarrow L_1$			$\rightarrow L_1$		
f					$\leftarrow L_1$		$\rightarrow L_1$	
g		$\leftarrow L_1$				$\leftarrow L_1$		$\rightarrow L_1$
h							$\leftarrow L_1$	

Process Mining am Beispiel – Ereignis Log zum Footprint (3)

$$\#L_1 = \{(a,a),(a,d),(a,e),(a,f),(a,g),(a,h),(b,b),(b,c),(b,d),(b,e),(b,f),(b,h),(c,b)(c,c),(c,f),(c,g),(c,h),(d,a),(d,b),(d,d),(d,e),(d,g),(d,h),(e,a),(e,b),(e,d),(e,e),(e,g),(e,h),(f,a),(f,b),(f,c),(f,d),(f,f),(f,h),(g,a),(g,c),(g,d),(g,e),(g,g),(h,a),(h,b),(h,c),(h,d),(h,e),(h,f),(h,h)\}$$

von \ nach	a	b	c	d	e	f	g	h
a	$\#L_1$	$\rightarrow L_1$	$\rightarrow L_1$	$\#L_1$	$\#L_1$	$\#L_1$	$\#L_1$	$\#L_1$
b	$\leftarrow L_1$	$\#L_1$	$\#L_1$	$\#L_1$	$\#L_1$	$\#L_1$	$\rightarrow L_1$	$\#L_1$
c	$\leftarrow L_1$	$\#L_1$	$\#L_1$	$\rightarrow L_1$	$\rightarrow L_1$	$\#L_1$	$\#L_1$	$\#L_1$
d	$\#L_1$	$\#L_1$	$\leftarrow L_1$	$\#L_1$		$\rightarrow L_1$	$\#L_1$	$\#L_1$
e	$\#L_1$	$\#L_1$	$\leftarrow L_1$		$\#L_1$	$\rightarrow L_1$	$\#L_1$	$\#L_1$
f	$\#L_1$	$\#L_1$	$\#L_1$	$\leftarrow L_1$	$\leftarrow L_1$	$\#L_1$	$\rightarrow L_1$	$\#L_1$
g	$\#L_1$	$\leftarrow L_1$	$\#L_1$	$\#L_1$	$\#L_1$	$\leftarrow L_1$	$\#L_1$	$\rightarrow L_1$
h	$\#L_1$	$\#L_1$	$\#L_1$	$\#L_1$	$\#L_1$	$\#L_1$	$\leftarrow L_1$	$\#L_1$

Process Mining am Beispiel – Ereignis Log zum Footprint (4)

- $||L1 = \{(d, e), (e, d)\}$

von \ nach	a	b	c	d	e	f	g	h
a	$\#L_1$	$\rightarrow L_1$	$\rightarrow L_1$	$\#L_1$	$\#L_1$	$\#L_1$	$\#L_1$	$\#L_1$
b	$\leftarrow L_1$	$\#L_1$	$\#L_1$	$\#L_1$	$\#L_1$	$\#L_1$	$\rightarrow L_1$	$\#L_1$
c	$\leftarrow L_1$	$\#L_1$	$\#L_1$	$\rightarrow L_1$	$\rightarrow L_1$	$\#L_1$	$\#L_1$	$\#L_1$
d	$\#L_1$	$\#L_1$	$\leftarrow L_1$	$\#L_1$	$\|\|L_1$	$\rightarrow L_1$	$\#L_1$	$\#L_1$
e	$\#L_1$	$\#L_1$	$\leftarrow L_1$	$\|\|L_1$	$\#L_1$	$\rightarrow L_1$	$\#L_1$	$\#L_1$
f	$\#L_1$	$\#L_1$	$\#L_1$	$\leftarrow L_1$	$\leftarrow L_1$	$\#L_1$	$\rightarrow L_1$	$\#L_1$
g	$\#L_1$	$\leftarrow L_1$	$\#L_1$	$\#L_1$	$\#L_1$	$\leftarrow L_1$	$\#L_1$	$\rightarrow L_1$
h	$\#L_1$	$\#L_1$	$\#L_1$	$\#L_1$	$\#L_1$	$\#L_1$	$\leftarrow L_1$	$\#L_1$

Process Mining am Beispiel - α Algorithmus

Sei L ein Ereignislog über eine Teilmenge T von Aktivitäten, gilt $T \subseteq \mathcal{A}$.

α(L) wird wie folgt ermittelt.

(1) $T_L = \{t \in T | \exists_{\sigma \in L} t \in \sigma\}$

(2) $T_I = \{t \in T | \exists_{\sigma \in L} t = first(\sigma)\}$

(3) $T_O = \{t \in T | \exists_{\sigma \in L} t = last(\sigma)\}$

(4) $X_L = \left\{ \begin{array}{c} (A,B) | A \subseteq T_L \wedge A \neq 0 \wedge B \subseteq T_L \wedge B \neq 0 \wedge \\ \forall_{a \in A} \forall_{b \in B} a \rightarrow {}_L b \; \forall_{a_1, a_2 \in A} a_1 \#_L a_2 \wedge \\ \forall_{b_1, b_2 \in B} b_1 \#_L b_2 \end{array} \right\}$

(5) $Y_L = \left\{ \begin{array}{c} (A,B) \in X_L | \forall_{(A',B') \in X_L} A \subseteq A' \wedge B \subseteq B' \Longrightarrow \\ (A,B) = (A',B') \end{array} \right\}$

(6) $P_L = \{p_{(A,B)} | (A,B) \in Y_L\} \cup \{i_L, o_L\}$

(7) $F_L = \{a, p_{(A,B)} | (A,B) \in Y_L \wedge a \in A\} \cup \{p_{(A,B)}, b | (A,B) \in Y_L \wedge b \in B\} \cup \{(i_L, t) | t \in T_I\} \cup \{(t, o_L) | t \in T_O\}$

(8) $\alpha(L) = (P_L, T_L, F_L)$

Process Mining am Beispiel - α Algorithmus (2)

$$T_L = \{a, b, c, d, e, f, g, h\}$$

$$T_I = \{a\}.$$

$$T_O = \{h\}$$

Process Mining am Beispiel - α Algorithmus (3)

von \ nach	a_1	a_2	...	a_m	b_1	b_2	...	b_n
a_1	#	#	...	#	→	→	...	→
a_2	#	#	...	#	→	→	...	→
...	...	...	...	...	...	...	...	...
a_m	#	#	...	#	→	→	...	→
b_1	←	←	...	←	#	#	...	#
b_2	←	←	...	←	#	#	...	#
...	...	...	...	....	...	...	...	...
b_n	←	←	...	←	#	#	...	#

Hilfsmatrix

Process Mining am Beispiel - α Algorithmus (4)

$$X_{L_1} = \{(\{a\},\{b\}),(\{a\},\{c\}),(\{a\},\{b,c\}),(\{b\},\{g\}),(\{c\},\{d\}),(\{c\},\{e\}),(\{d\},\{f\}),(\{e\},\{f\}),(\{f\},\{g\}),(\{g\},\{h\}),(\{b,f\},\{g\})\}$$

Schritt (4)

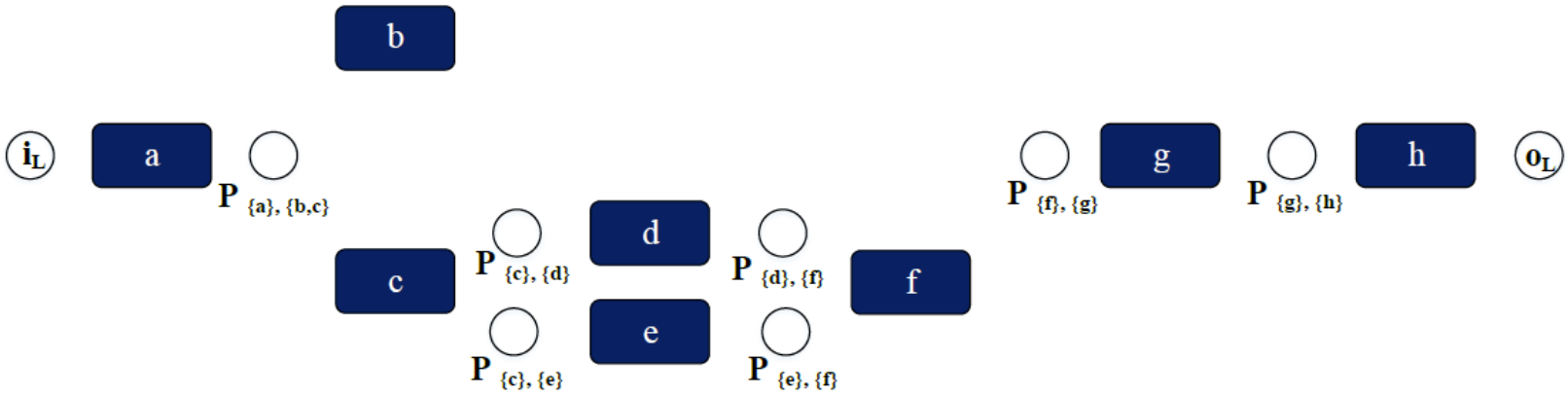
$$Y'_{L_1} = \{(\{a\},\{b\}),(\{a\},\{c\}),(\{a\},\{b,c\}),(\{b\},\{g\}),(\{c\},\{d\}),(\{c\},\{e\}),(\{d\},\{f\}),(\{e\},\{f\}),(\{f\},\{g\}),(\{g\},\{h\}),(\{b,f\},\{g\})\}$$

$$Y_{L_1} = \{(\{a\},\{b,c\}),(\{c\},\{d\}),(\{c\},\{e\}),(\{d\},\{f\}),(\{e\},\{f\}),(\{g\},\{h\}),(\{b,f\},\{g\})\}$$

Schritt (5)

Process Mining am Beispiel - α Algorithmus (5)

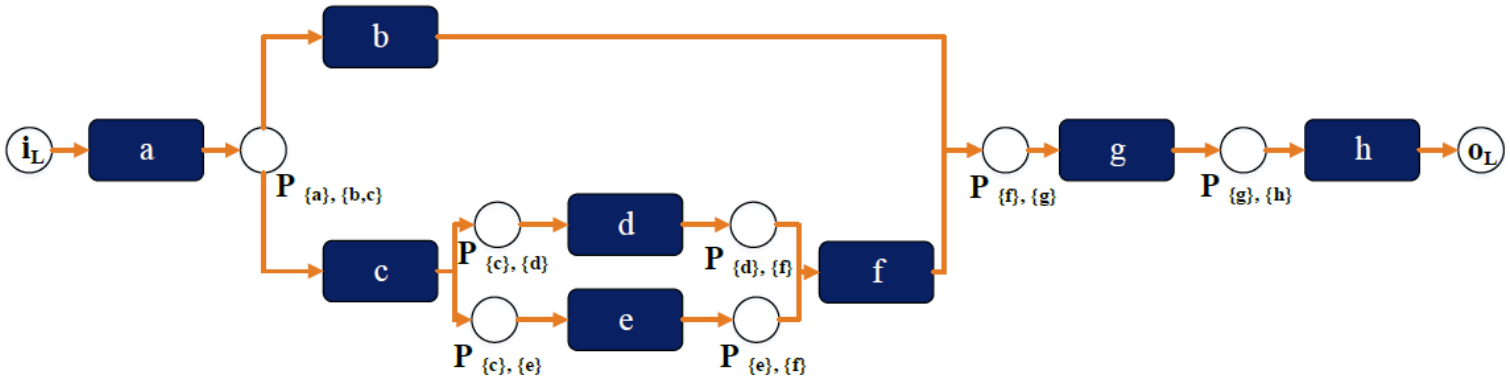
$$P_{L_1} = \{p_{(\{a\},\{b,c\})}, p_{(\{c\},\{d\})}, p_{(\{c\},\{e\})}, p_{(\{d\},\{f\})}, p_{(\{e\},\{f\})}, p_{(\{g\},\{h\})}, p_{(\{b,f\},\{g\})}, i_{L_1}, o_{L_1}\}$$



Schritt (6)

Process Mining am Beispiel - α Algorithmus (6)

$$F_{L_1} = \left\{ \begin{array}{c} (i_L, a), (a, p_{(\{a\},\{b,c\})}), (p_{(\{a\},\{b,c\})}, b), (p_{(\{a\},\{b,c\})}, c), \\ (c, p_{(\{c\},\{d\})}), (p_{(\{c\},\{d\})}, d), (c, p_{(\{c\},\{e\})}), (p_{(\{c\},\{e\})}, e), (d, p_{(\{d\},\{f\})}), \\ (p_{(\{d\},\{f\})}, f), (e, p_{(\{e\},\{f\})}), (p_{(\{e\},\{f\})}, f), (g, p_{(\{g\},\{h\})}), (p_{(\{g\},\{h\})}, h), \\ (b, p_{(\{b,f\},\{g\})}), (f, p_{(\{b,f\},\{g\})}), (p_{(\{b,f\},\{g\})}, g), (g, o_L) \end{array} \right\}$$



Schritt (7)

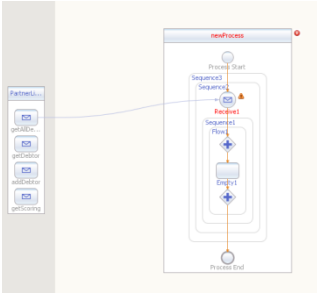
Outline

1. Introduction & Motivation
2. Analyzing and capturing business processes
3. Modeling Business Processes
4. Business Process Management
5. Case Management & Decision Management
6. Business Process Mining
7. Business Process Automation

Implementing a business process workflow

Hard coded services execution:
- Implementation via programming language (e.g. Java, C#,..)
- Allows process specific handling/implementation
- Hardcoded sequences
- Requires specific execution runtime
- Hard to maintain

Business process/workflow engines:
- Abstract (implementation independent) process model
- Execution via process execution runtime
- Flexible process implementation
- Easy to maintain and to enhance

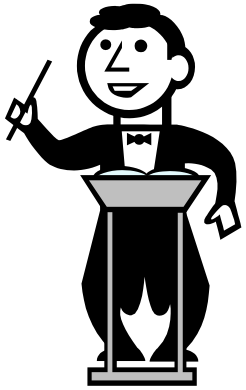


Business Process Management System

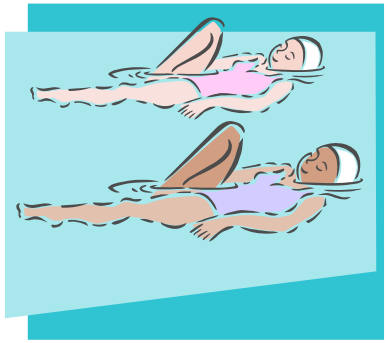
Business Process Management Systems are the technology to enable the business process automation. They provide the information technology to model and run processes, setup integrated business processes, monitor the process performance and introduce collaboration within the organizations.

BPMS: Orchestration vs. Choreography

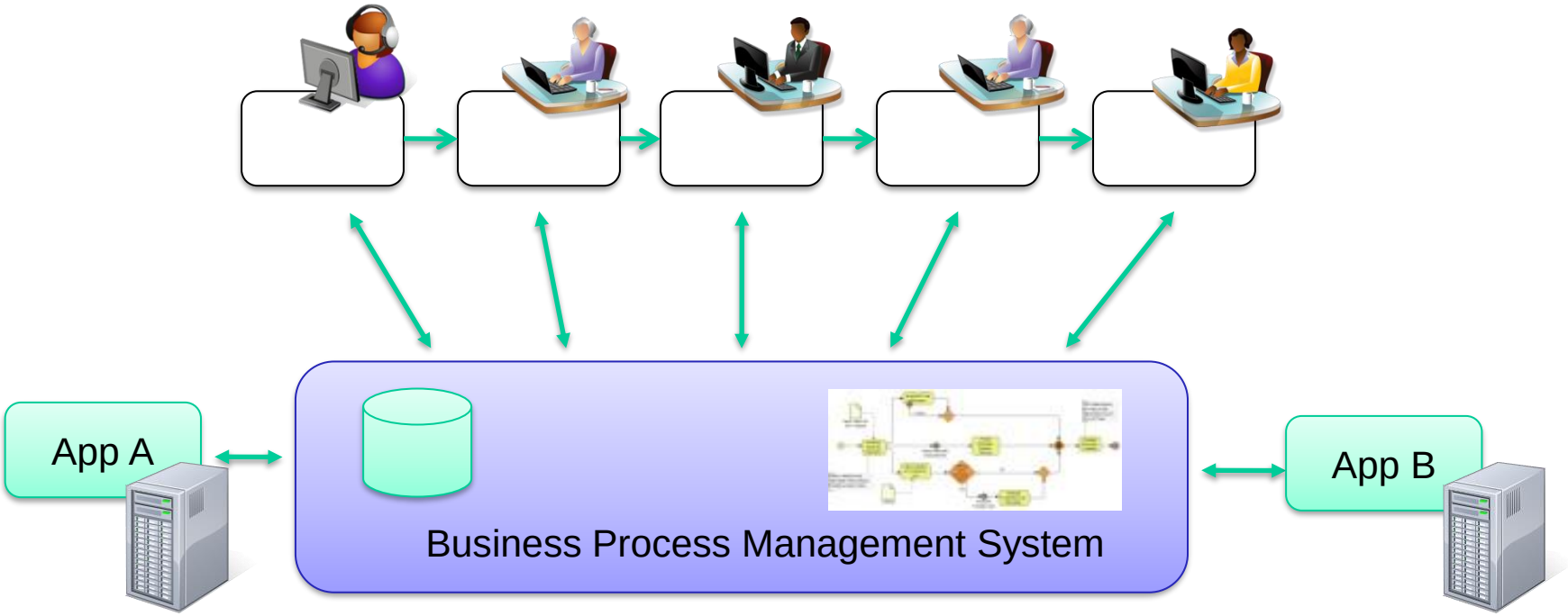
Orchestration:
- Centralized coordination
- Controlled process steps sequence
- Defined messages types being exchanged by the web services
- Hides process (web service) internals



Choreography:
- Describes the collaboration of the business process partners
- Focus on peer-to-peer communication
- Doesn’t have a central coordination

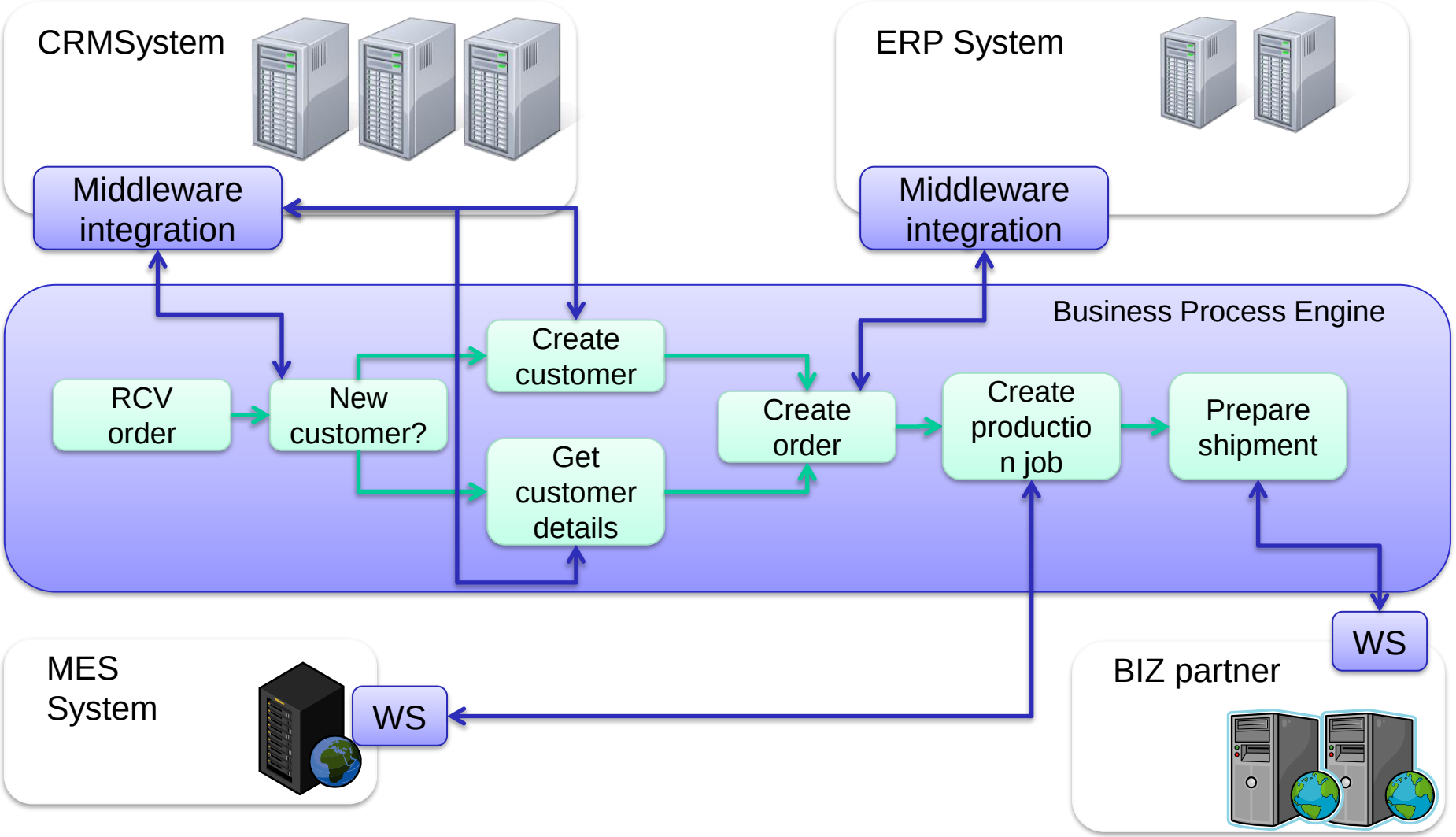


Business Process Management System - Overview

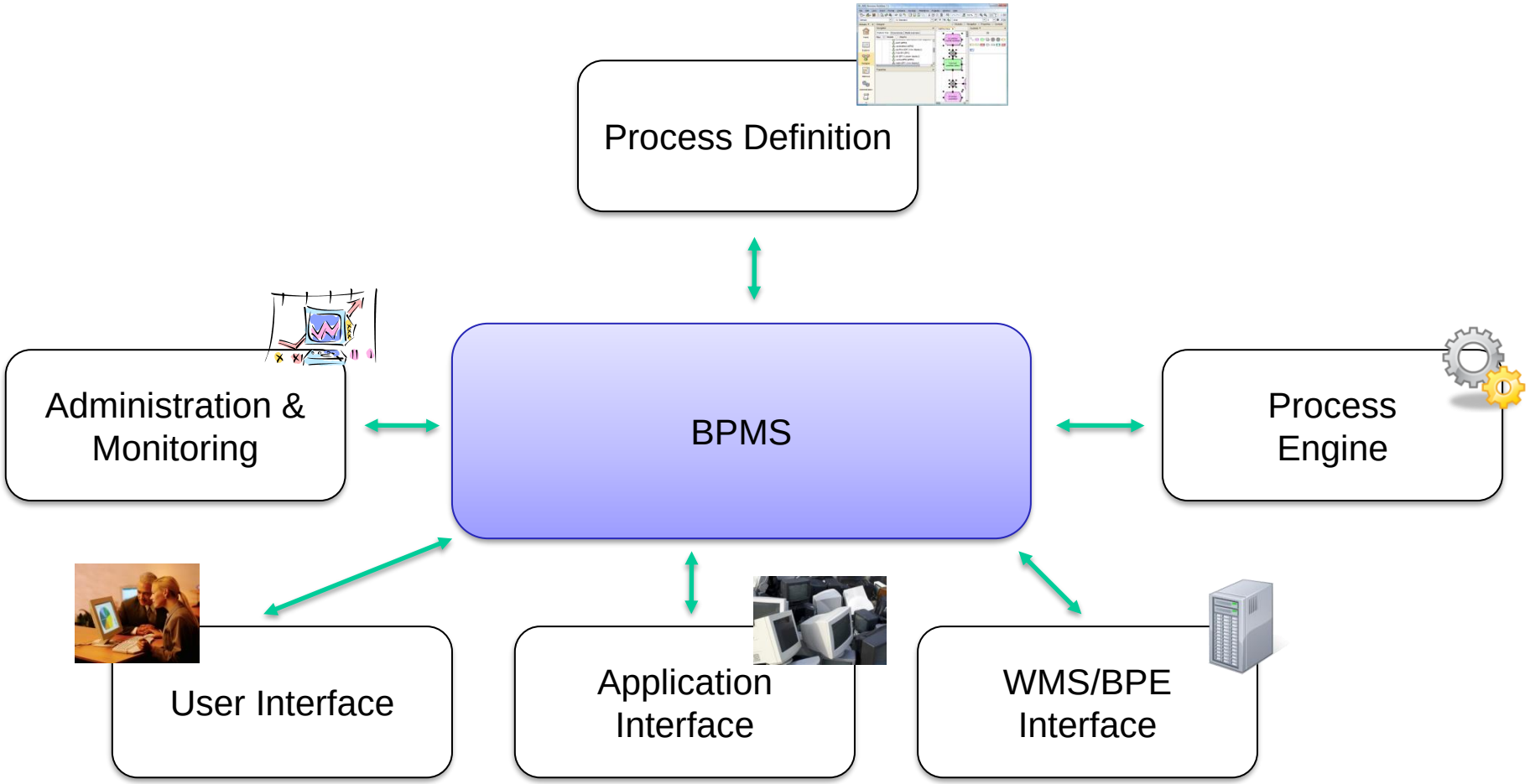


- Control the process execution (task assignments, task sequences, application interfaces, ..)
- Requires detailed process model

Executing a business process w/ a BPE

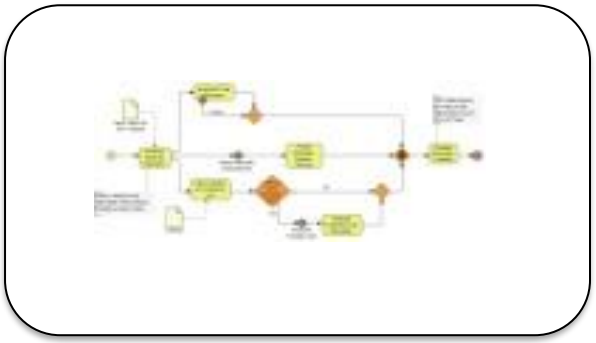


Business Process Management Systems - Components



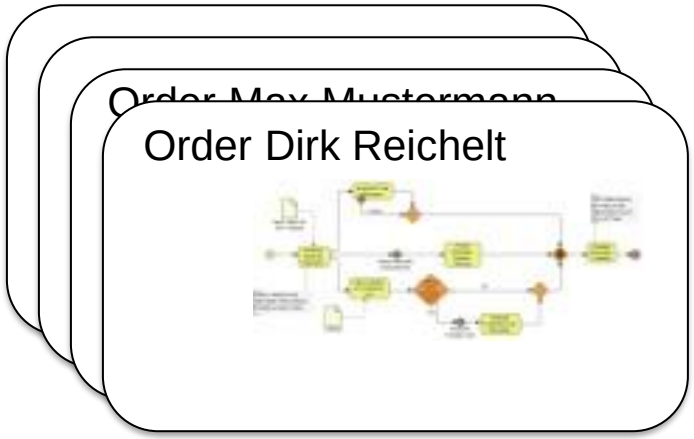
Process Model vs. Process Instance

Process Model:



Execution

Process Instances:



- The BPE creates a new process instance for each model execution
- The process instances run in parallel
- Each instance holds ist own state and data
- The BPE controls the execution of the process instances

BPMS: Design Time vs. Run time

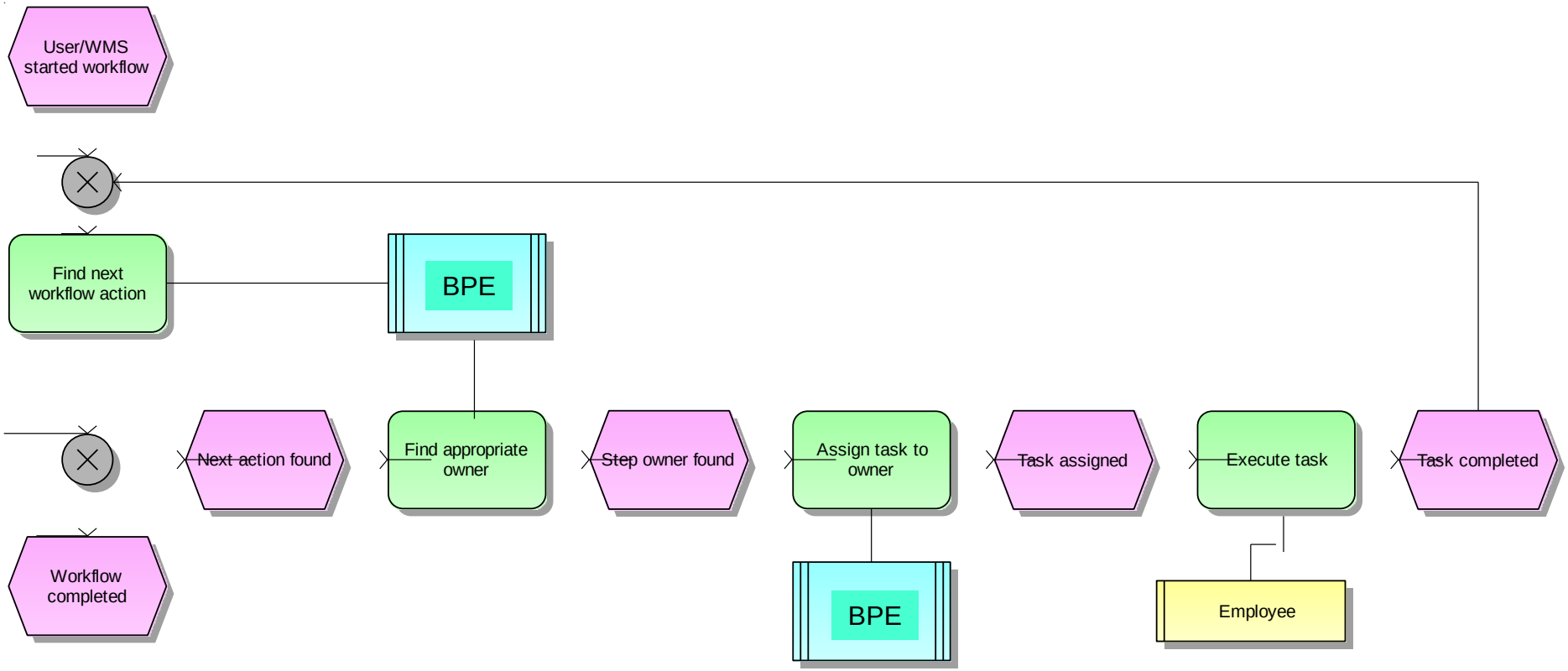
- Process definition
  step sequence, rules, events, …
- Organizational setup
  departments, positions, resposibilities
- Data model
  input and output data
- Application systems
  Interfaces to other systems being used by the process

Design time

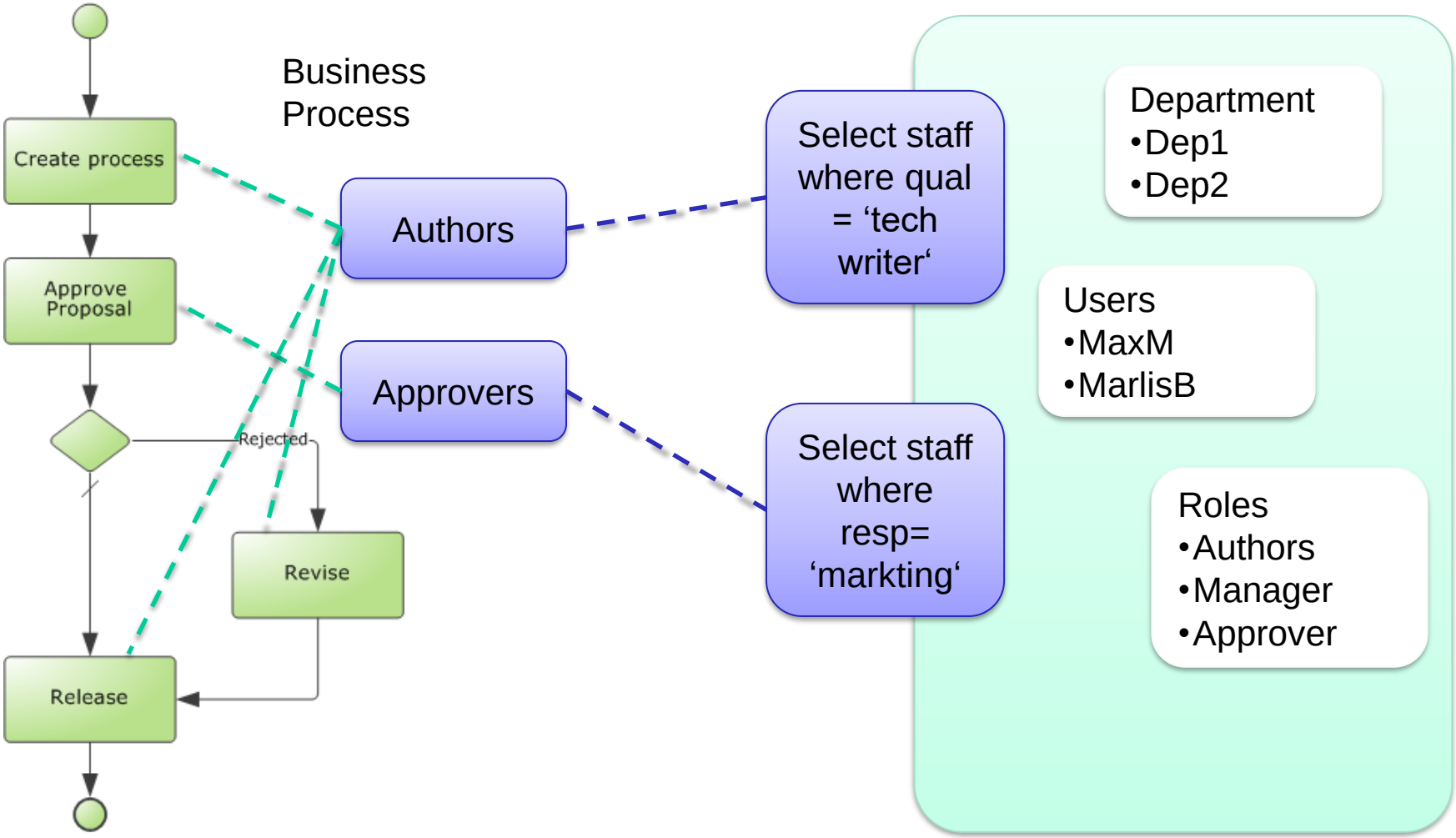
- Execute the Process
  Create process instances, control step sequence and assignments
  Manage process instance data and state
  Link to other IT systems

Run time

BPMS: Process Execution



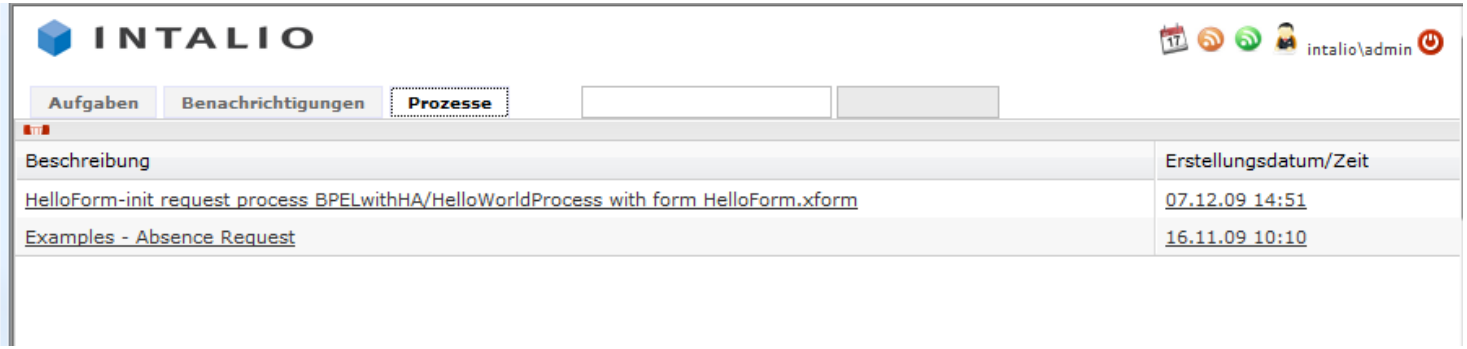
Human activties: An example

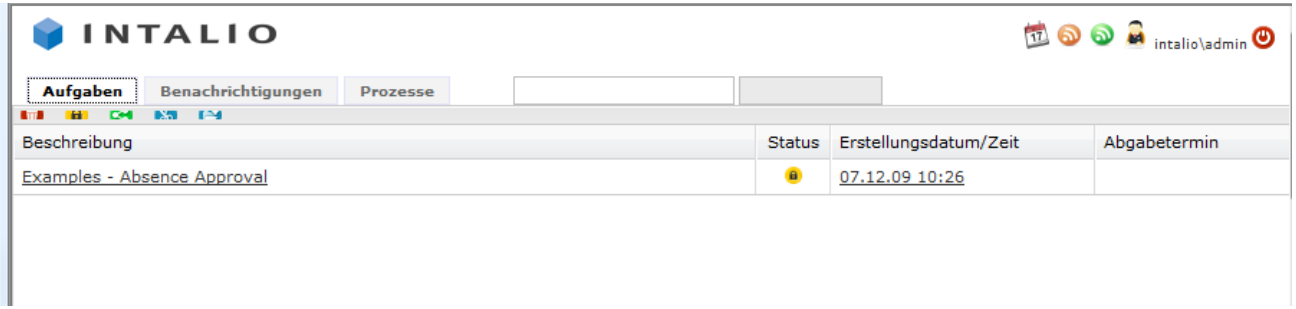


BPMS: User Interface

The key elements for a BPMS user interface are:

- A task list
- A process list
- A list of the notifications





BPMS - People and roles

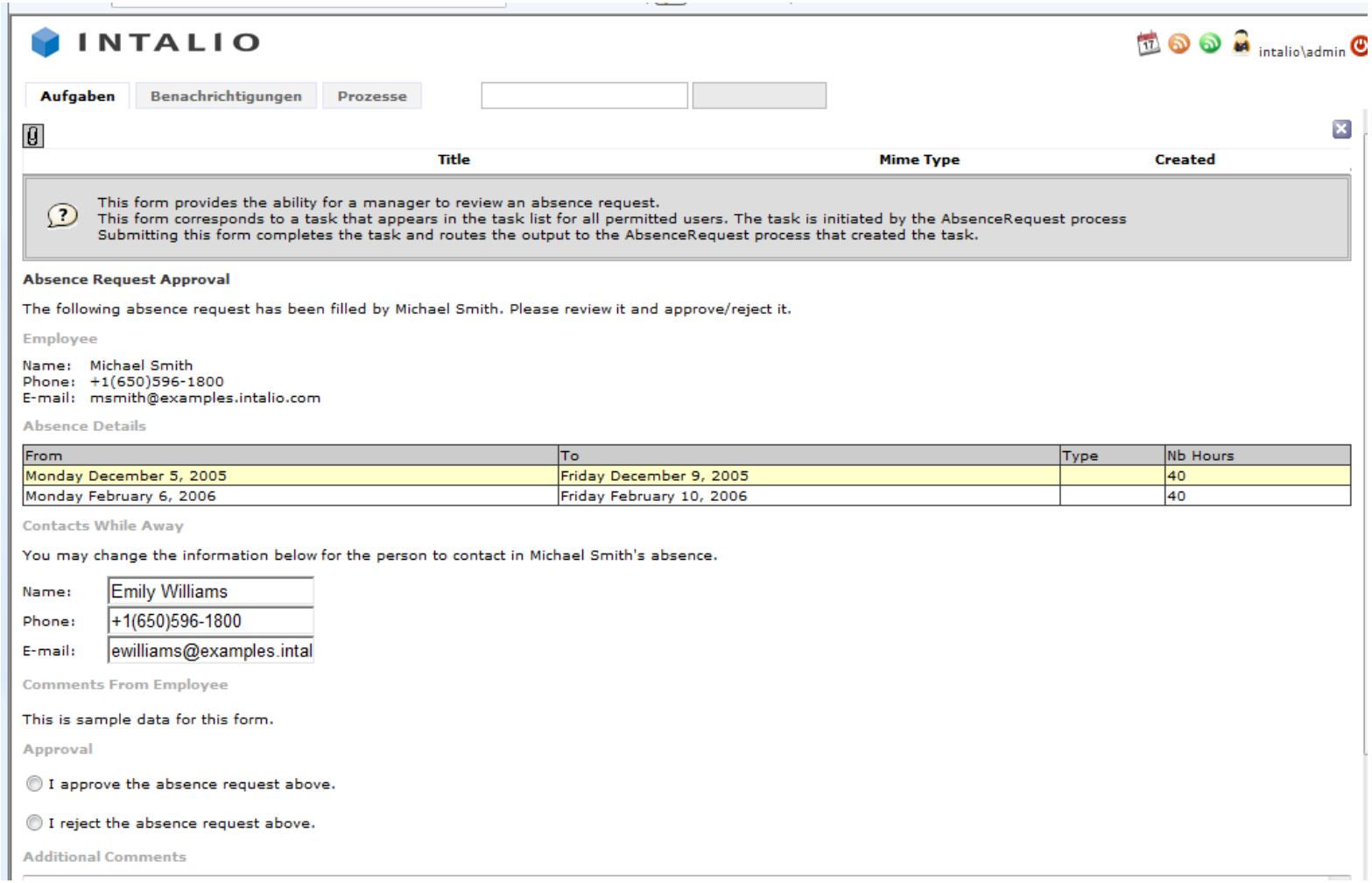
- **Task initiator** is the person associated with **triggering the process** instance at its creation time.
- **Task stakeholders** are people who can **influence the progress of a process instance**, for example, by adding ad-hoc attachments, forwarding a task, or simply observing the progress of the process instance.
- **Potential owners** of a task are persons who **receive the task so that they can claim** and **complete it**.
- **Excluded owners** may **not become an actual** or **potential owner** and thus they may not reserve or start the task.
- An **actual owner** of a task is the **person actually performing the task**. A task has exactly one actual owner.
- **Business administrators** are people allowed to **perform administrative actions** on the business process, such as resolving missed deadlines.
- **Notification recipients** are persons who **receive the notification**, such as happens when a deadline is missed or when a milestone is reached.

BPMS – Typical User Tasks

Human tasks:

- Start new workflow instance
- Claim an task
- Execute a task
- Delegate a task
- Assign task
- Release task
- Skip task
- Abort workflow
- Confirm notifications

BPMS – User Task Example



Get it working… Enterprise Integration Pattern