A4M36AOS – Architektury orientované na služby

9. Business Process Modeling

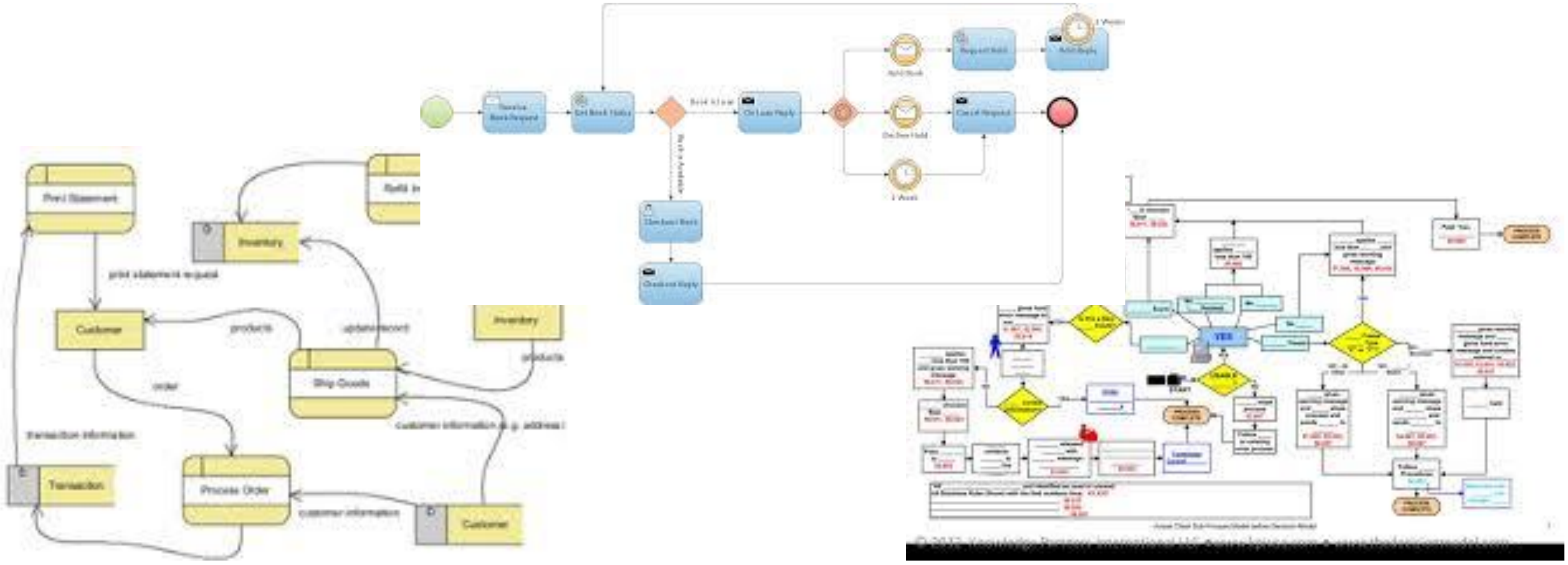
Jiří Vokřínek

Agent Technology Center
Department of Computer Science and Engineering
Faculty of Electrical Engineering, Czech Technical University in Prague

jiri.vokrinek@fel.cvut.cz http://agents.fel.cvut.cz

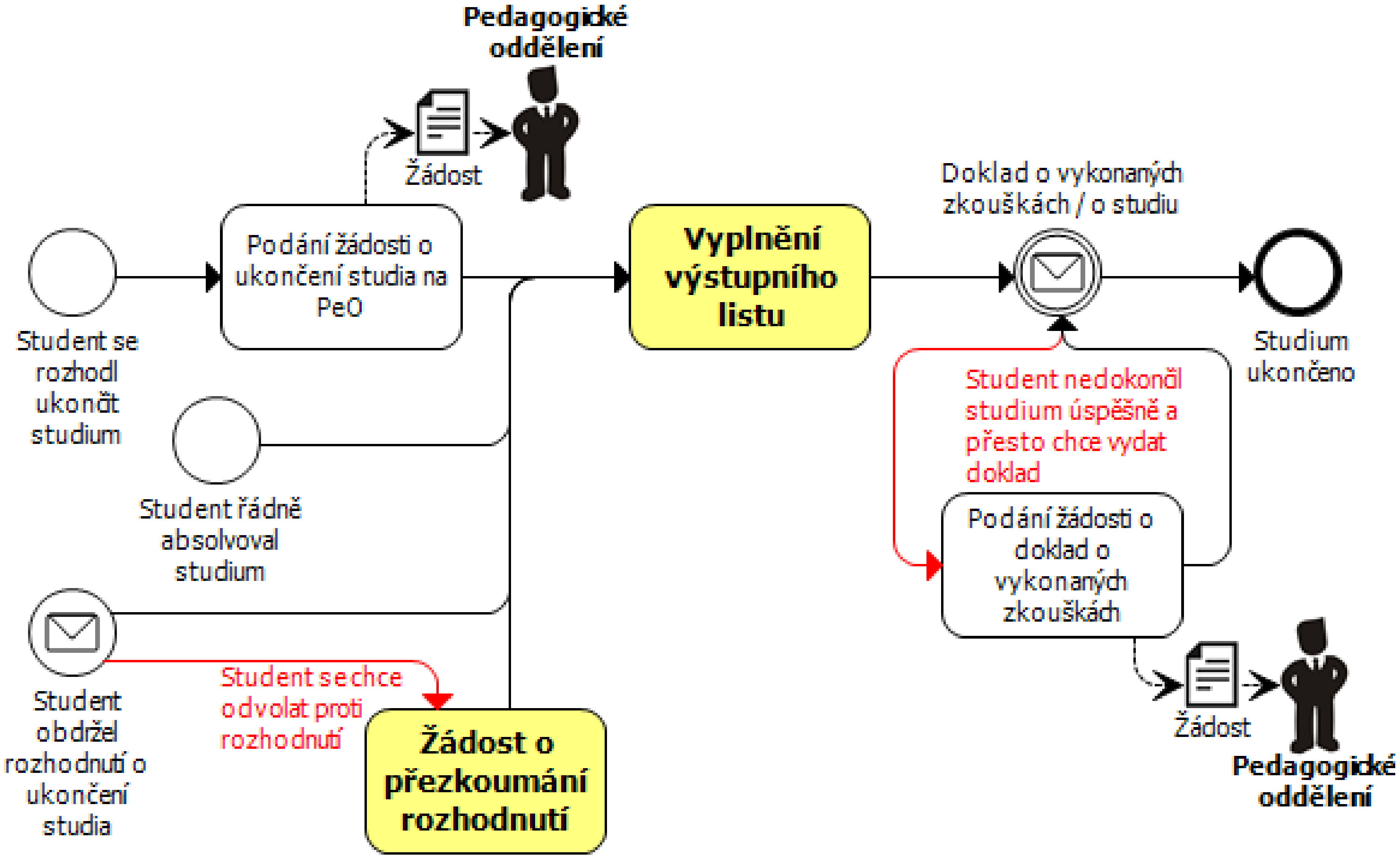
Business Process Modeling

- Processes capturing/representation
- Business analysis for processes improvements
- Change management



Business Process Modeling

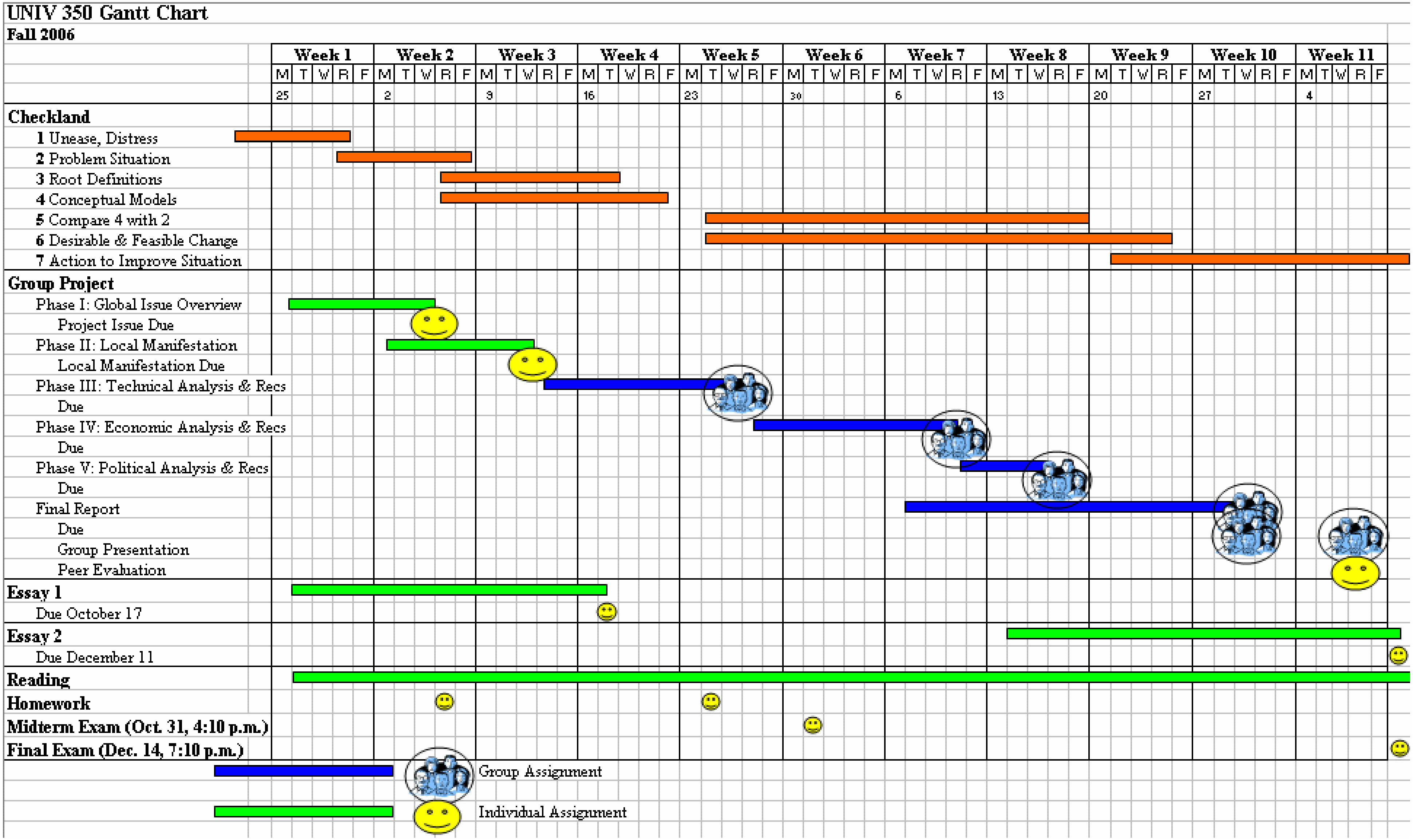
- Example: https://dekanat.fel.cvut.cz/procesy


Pedagogické oddělení
Žádost
Doklad o vykonaných zkouškách / o studiu
Podání žádosti o ukončení studia na PeO
Vyplnění výstupního listu
Student se rozhodl ukončit studium
Studium ukončeno
Student nedokončil studium úspěšně a přesto chce vydat doklad
Student řádně absolvoval studium
Podání žádosti o doklad o vykonaných zkouškách
Student obdržel rozhodnutí o ukončení studia
Student se chce odvolat proti rozhodnutí
Žádost o přezkoumání rozhodnutí
Žádost
Pedagogické oddělení

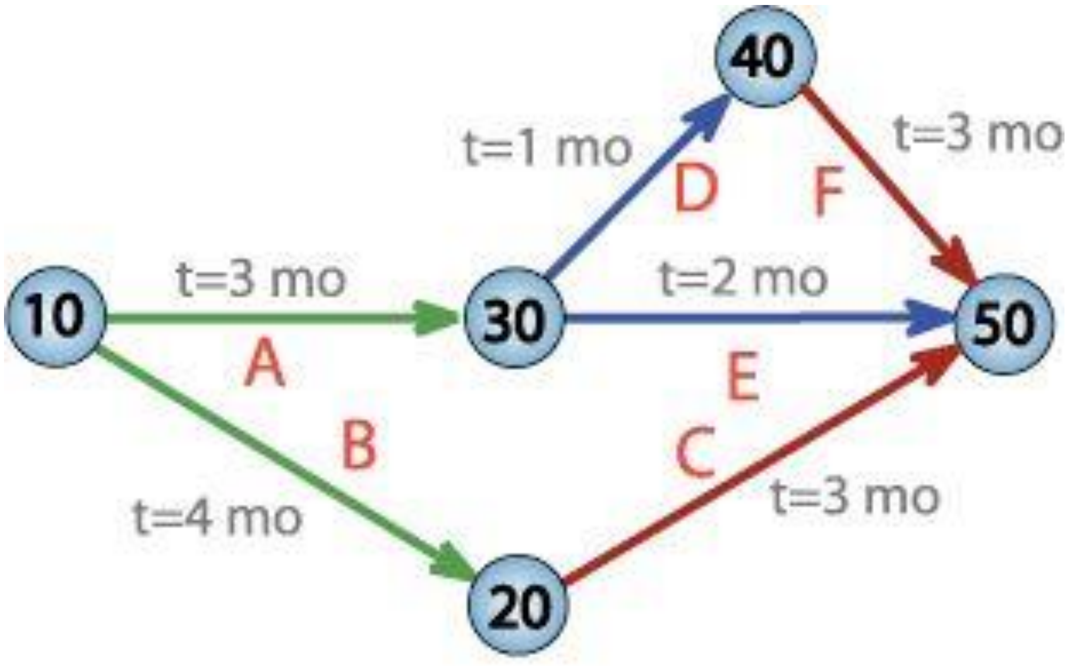
History of BPM

- Gantt chart (~1899)
- PERT charts (1950s)
- Data Flow Diagram (1970s)
- UML (mid 1990s)
- BPMN (2000+)

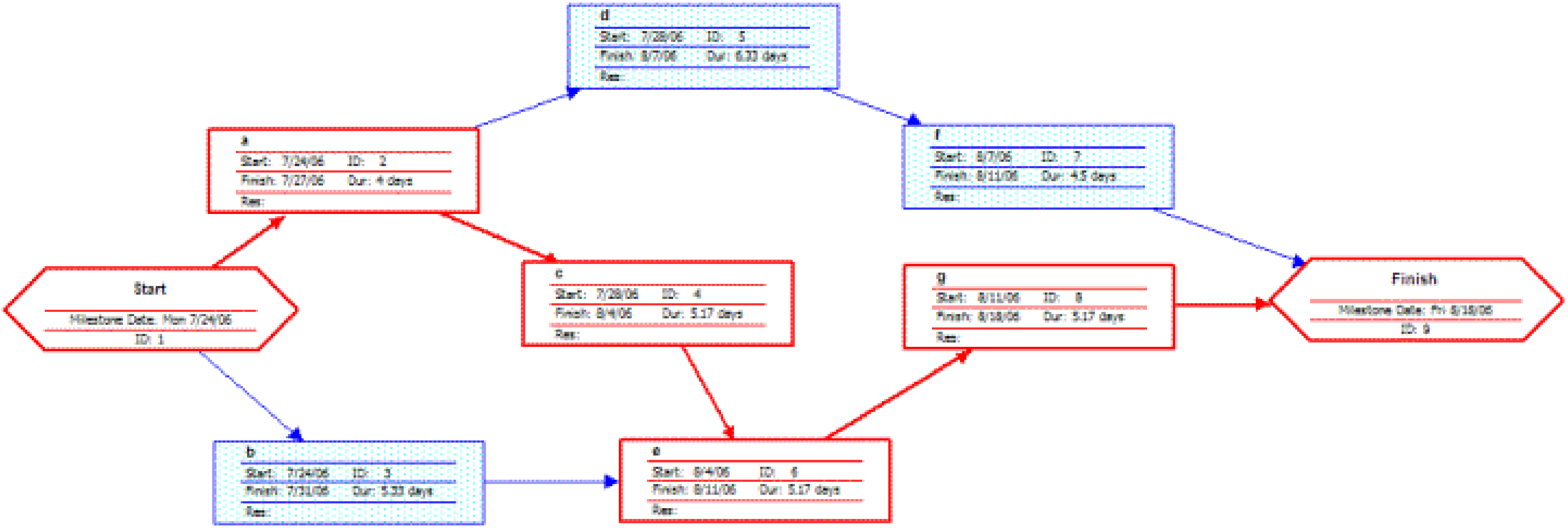
Gantt Chart


UNIV 350 Gantt Chart
Fall 2006
Week 1
Week 2
Week 3
Week 4
Week 5
Week 6
Week 7
Week 8
Week 9
Week 10
Week 11
M T W R F
25
2
9
16
23
30
6
13
20
27
4
Checkland
1 Unease, Distress
2 Problem Situation
3 Root Definitions
4 Conceptual Models
5 Compare 4 with 2
6 Desirable & Feasible Change
7 Action to Improve Situation
Group Project
Phase I: Global Issue Overview
Project Issue Due
Phase II: Local Manifestation
Local Manifestation Due
Phase III: Technical Analysis & Recs
Due
Phase IV: Economic Analysis & Recs
Due
Phase V: Political Analysis & Recs
Due
Final Report
Due
Group Presentation
Peer Evaluation
Essay 1
Due October 17
Essay 2
Due December 11
Reading
Homework
Midterm Exam (Oct. 31, 4:10 p.m.)
Final Exam (Dec. 14, 7:10 p.m.)
Group Assignment
Individual Assignment

PERT



PERT network chart for a seven-month project with five milestones (10 through 50) and six activities (A through F)



d
Start: 7/28/06 ID: 5
Finish: 8/7/06 Dur: 6.33 days
Res:

a
Start: 7/24/06 ID: 2
Finish: 7/27/06 Dur: 4 days
Res:

f
Start: 8/7/06 ID: 7
Finish: 8/11/06 Dur: 4.5 days
Res:

Start
Milestone Date: Mon 7/24/06
ID: 1

c
Start: 7/28/06 ID: 4
Finish: 8/4/06 Dur: 5.17 days
Res:

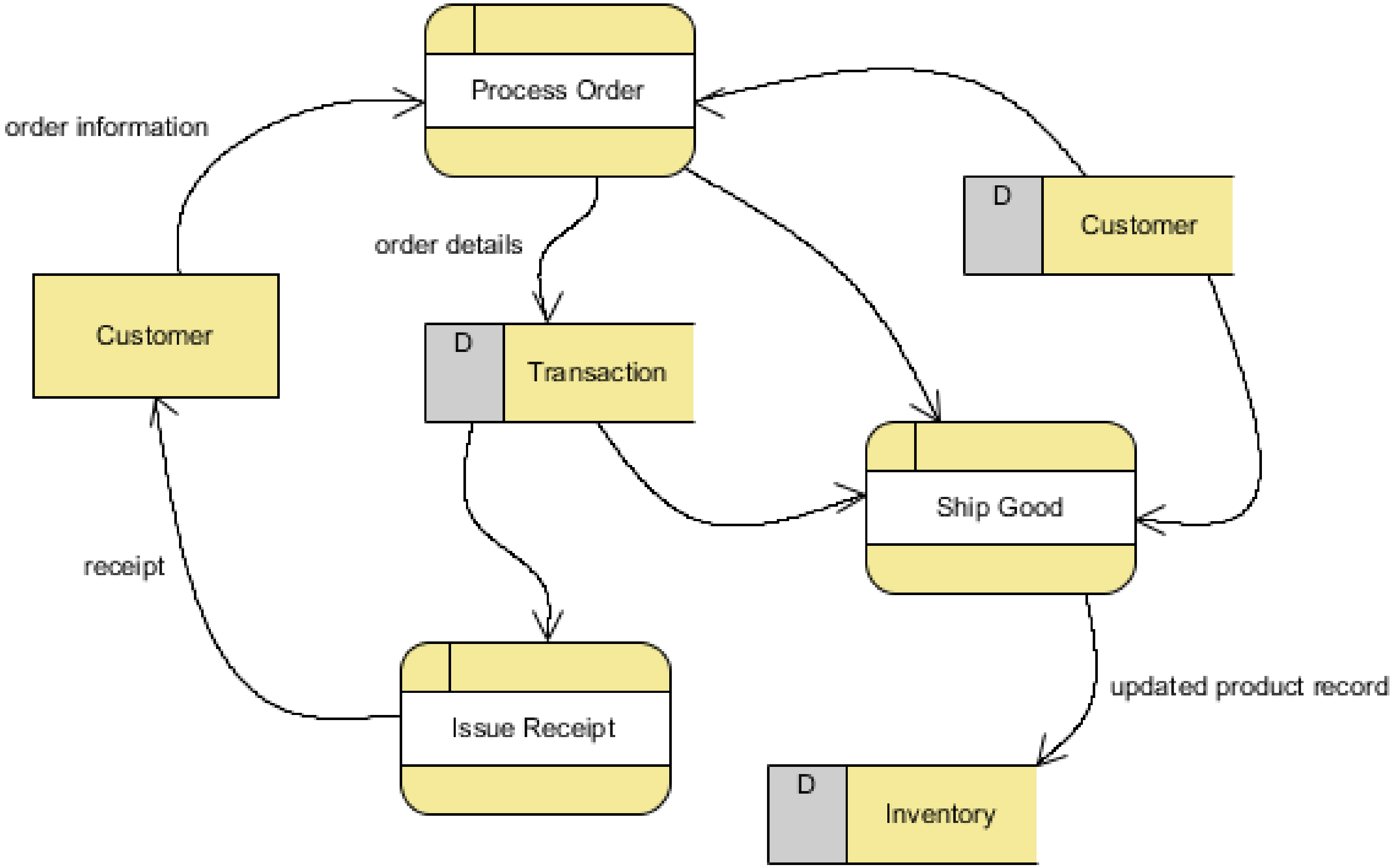
g
Start: 8/11/06 ID: 8
Finish: 8/18/06 Dur: 5.17 days
Res:

Finish
Milestone Date: Fri 8/18/06
ID: 9

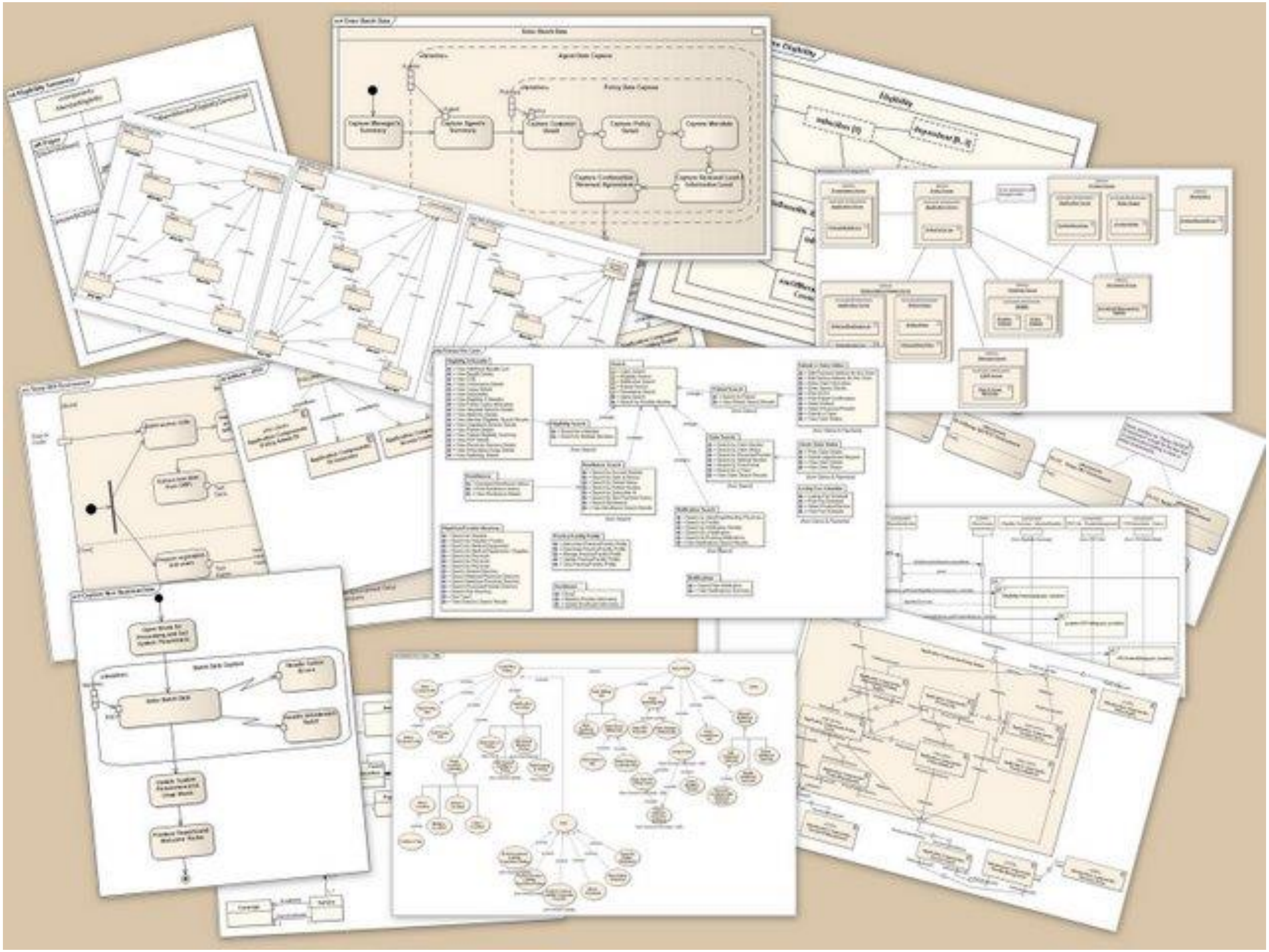
b
Start: 7/24/06 ID: 3
Finish: 7/31/06 Dur: 5.33 days
Res:

e
Start: 8/4/06 ID: 6
Finish: 8/11/06 Dur: 5.17 days
Res:

Data Flow Diagram



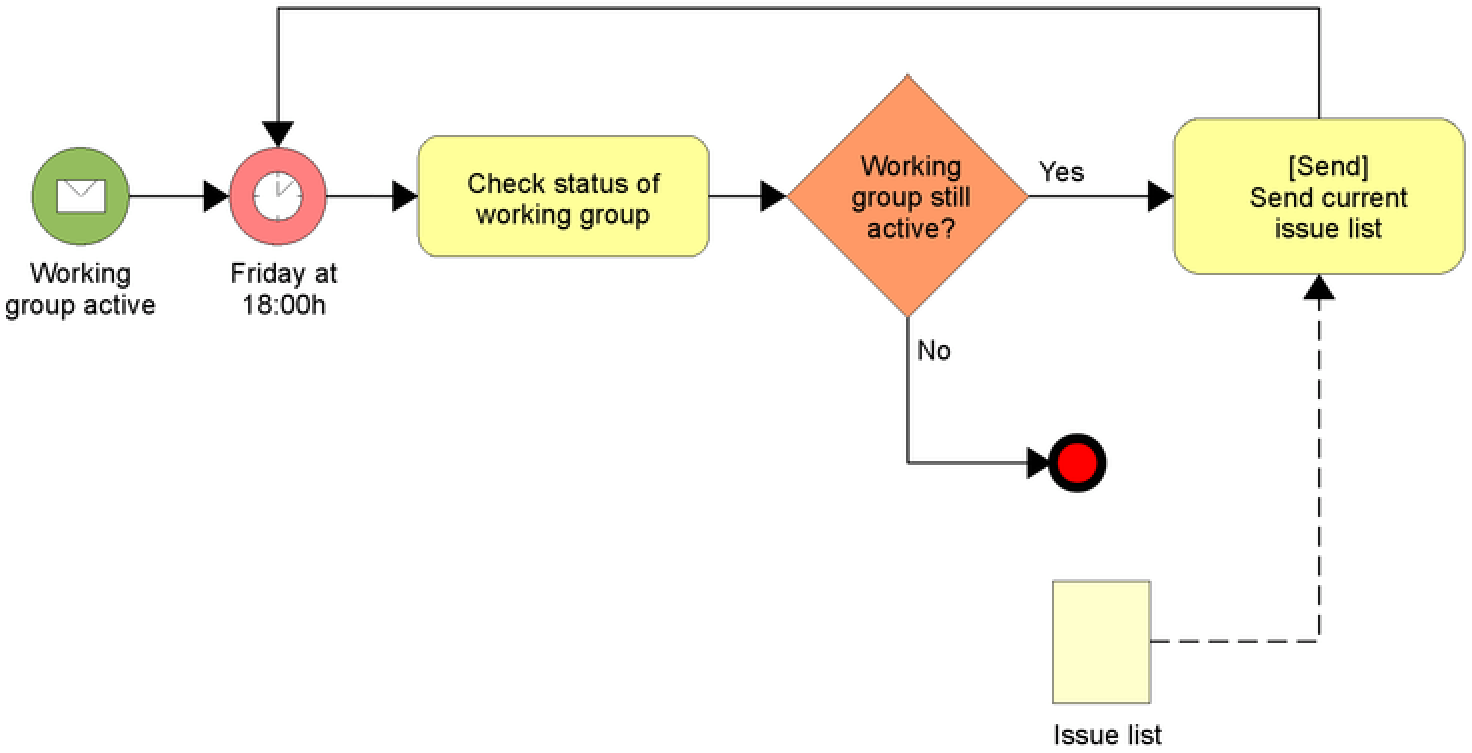
UML


Enter Batch Data
Agent Data Capture
Policy Data Capture
Capture Manager's Summary
Capture Agent's Summary
Capture Customer Detail
Capture Policy Detail
Capture Mandate
Capture Confirmation Renewal Agreement
Capture Renewal Lead Substitution Level
Eligibility
Batch Data Capture
Enter Batch Data

Business Process Modeling Notation (BPMN)

BPMN

- Graphical representation for specifying business processes in a business process (only) modeling


Working group active
Friday at 18:00h
Check status of working group
Working group still active?
Yes
No
[Send] Send current issue list
Issue list

BPMN

- Based on a flowcharting technique very similar to activity diagrams from UML
- Intuitive notation to business users yet able to represent complex process semantics
- Provides a mapping between the graphics of the notation to the underlying constructs of execution languages (BPEL)

BPMN

- Examples

Introduction to BPMN 1.1

Sylvain ASTIER – 2008: sastier@diveintobpm.org

http://diveintobpm.org

Business Process Execution Language (BPEL)

BPEL

- Web Service composition language
- Used for web service orchestration
- BPEL was originally developed by BEA, IBM, and Microsoft. Version 1.1 also includes input from SAP and Siebel.
- The OASIS TC “Web Services Business Process Execution Language” now continues the standardization of BPEL

BPEL

- BPEL4WS 1.0 (7/2002)
  - Original proposal from BEA, IBM, Microsoft
  - Combined ideas from IBM’s WSFL and Microsoft’s XLANG
- BPEL4WS 1.1 (5/2003)
  - Revised proposal submitted to OASIS
  - With additional contributions from SAP and Siebel

BPEL

- WS-BPEL 2.0 (6/2007)
  - Formalization of 1.1 capabilities
  - OASIS formally adopted standard
- WS-BPEL 2.0 and beyond (10/2007)
  - Additional proposals on the table
  - Vendors beginning to ship products conforming to standards

BPEL

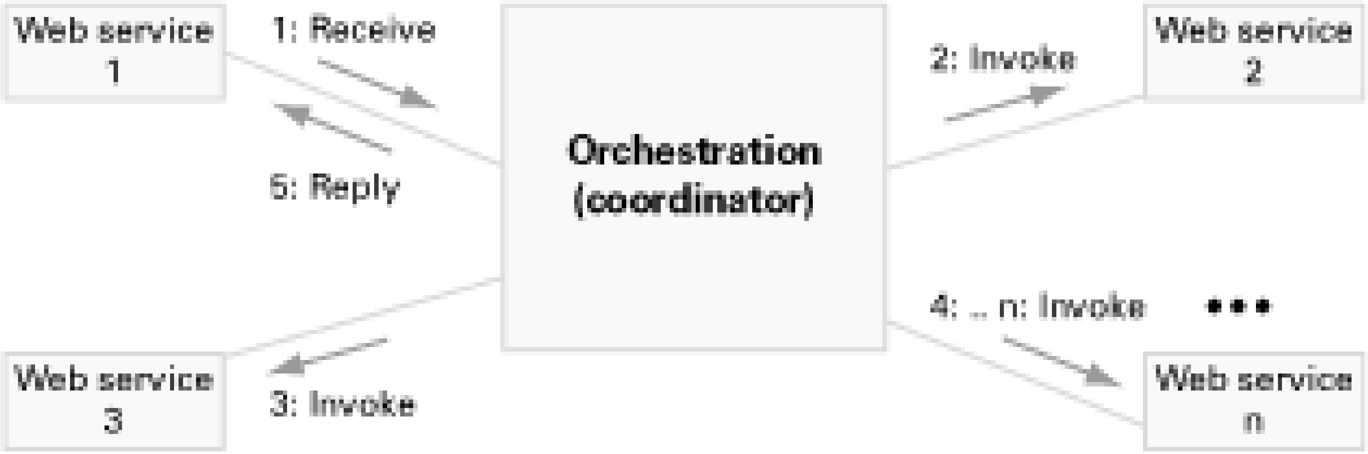
Is this your "new age" IT organization?
BPEL
business process modeling

BPEL

Is this your "new age" IT organization?
BPEL
business process modeling
?

BPEL

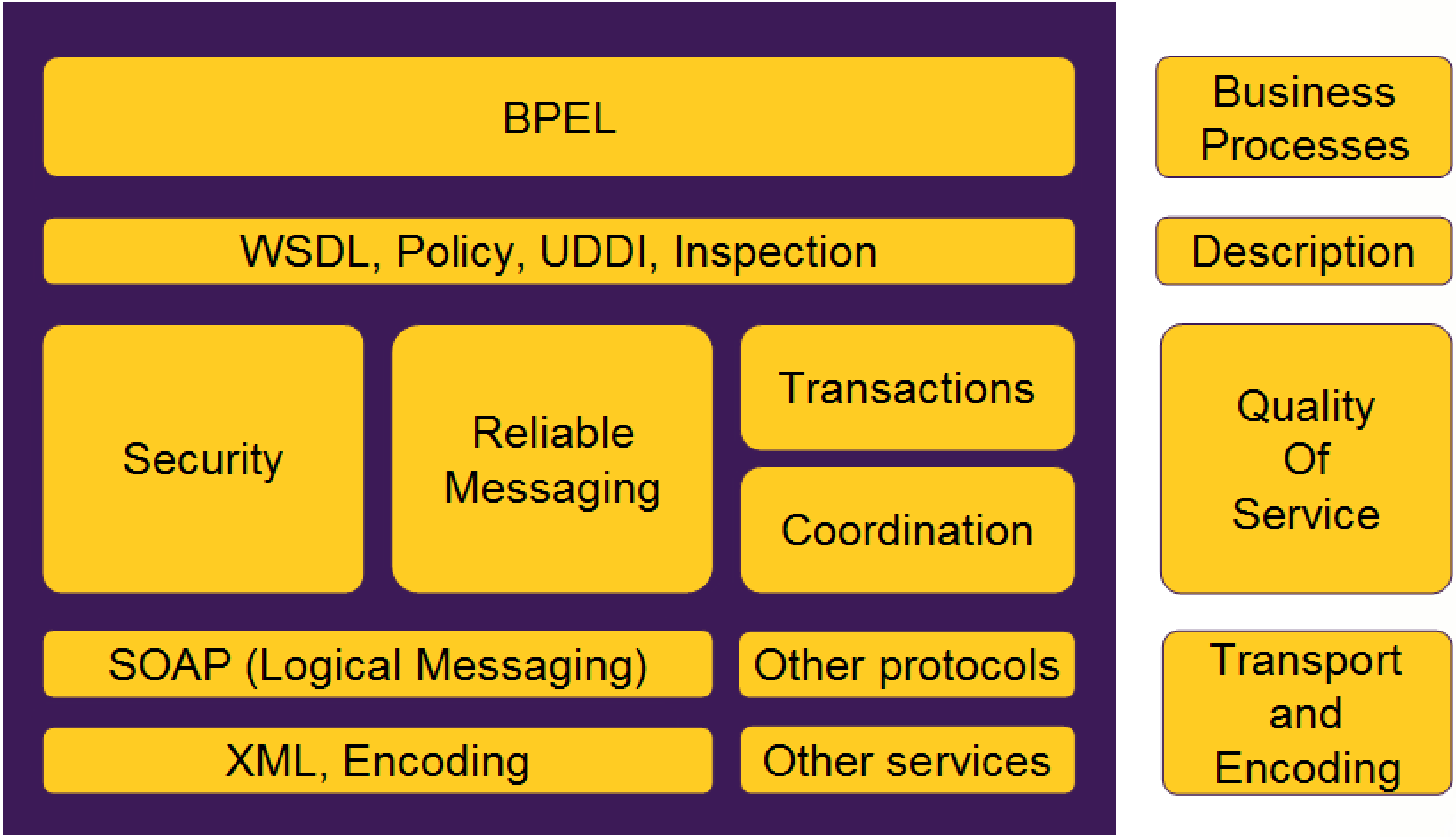
- Defines business processes that interact with external entities through Web services
- The definitions use XML and are not concerned with the graphical representation of processes
- Defines a set of Web service orchestration concepts



BPEL

- Supports the implicit creation and termination of process instances as the basic lifecycle mechanism
- Defines a long-running transaction model to support failure recovery
- Uses Web services as the model for process decomposition and assembly
- Builds on compatible Web services standards

BPEL


BPEL
WSDL, Policy, UDDI, Inspection
Security
Reliable Messaging
Transactions
Coordination
SOAP (Logical Messaging)
Other protocols
XML, Encoding
Other services
Business Processes
Description
Quality Of Service
Transport and Encoding

BPEL and WSDL

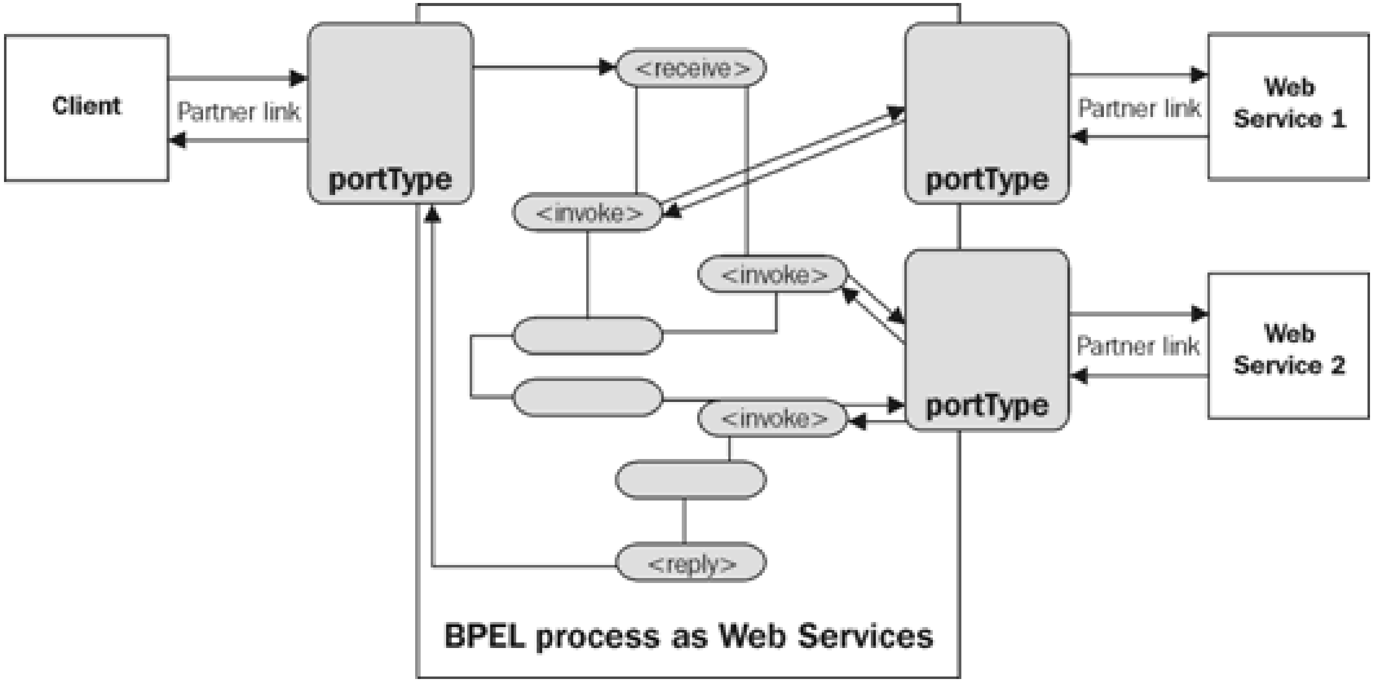
- BPEL processes exposed as WSDL services
- Message exchanges map to WSDL operations
- WSDL can be derived from partner definitions and the role played by the process in interactions with partners
- Interfaces exposed by the BPEL process
- Interfaces consumed by the BPEL process

BPEL and WSDL

- BPEL uses Web Services (BPEL is orchestrating these services)
- The BPEL process itself is a Web Service (it has interfaces) – when defining a BPEL process, it also is described by a WSDL (its interface)
- WSDL Port Types are named sets of abstract operations
- WSDL extensions are used to identify which port types are used to link services

BPEL as Process

- Most BPEL applications are executable processes
- Describes the interfaces to external data sources
- Describes the control flow for orchestrating these data sources



BPEL Partners

- BPEL supports different relationships with partners
  - Partners may invoke the BPEL process
  - BPEL process may invoke partners
  - Partners and the BPEL process play both roles
- BPEL processes will have at least one client (the partner activating the process)

BPEL as Language

- Business process modeling language that is executable

- Language for specifying business process behavior based on Web Services

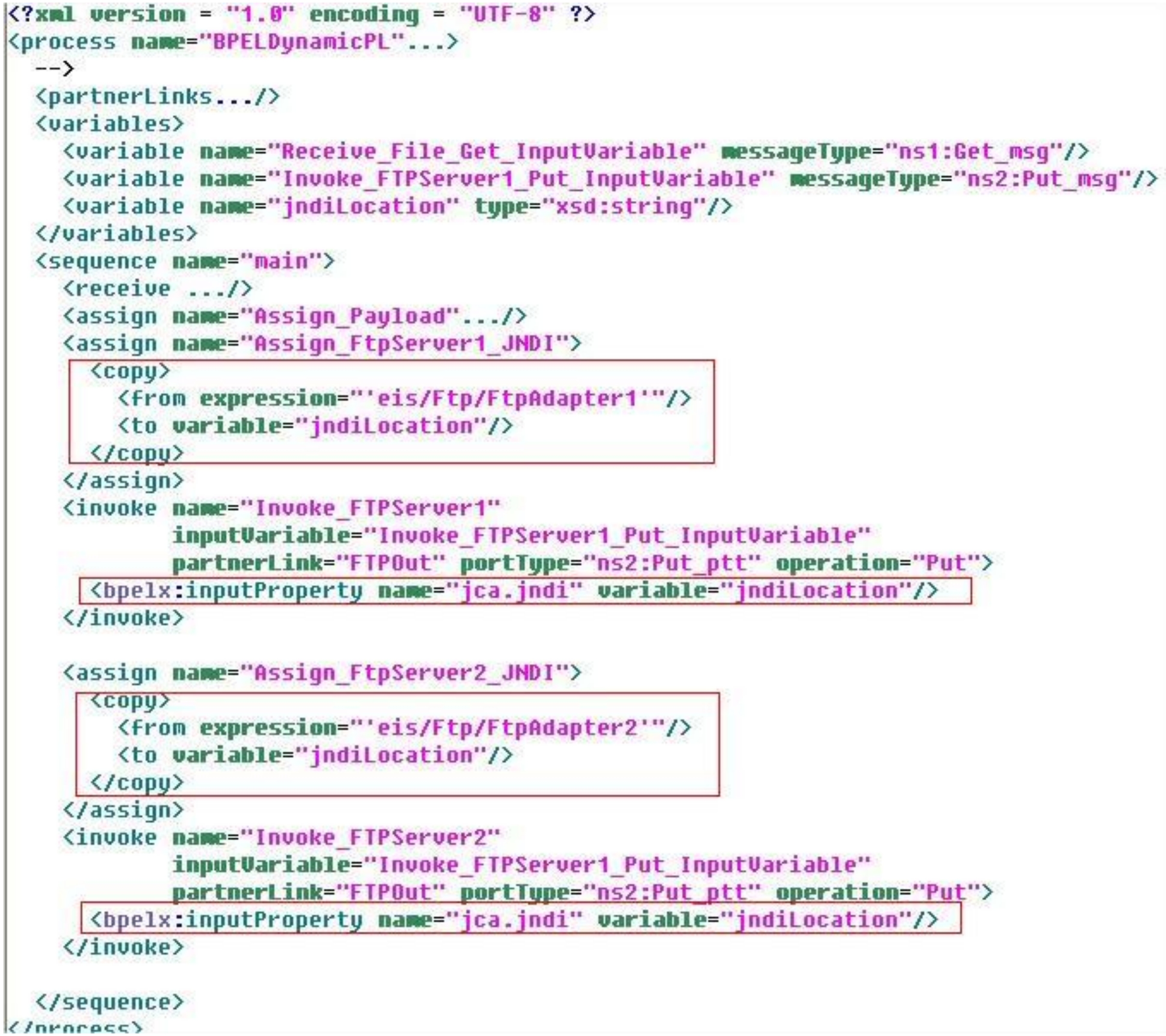
- Serialized in XML and aims to enable programming in the large (generally refers to the high-level state transition interactions of a process)

- No standardized graphical notation for BPEL – XML is used as the standardized syntax

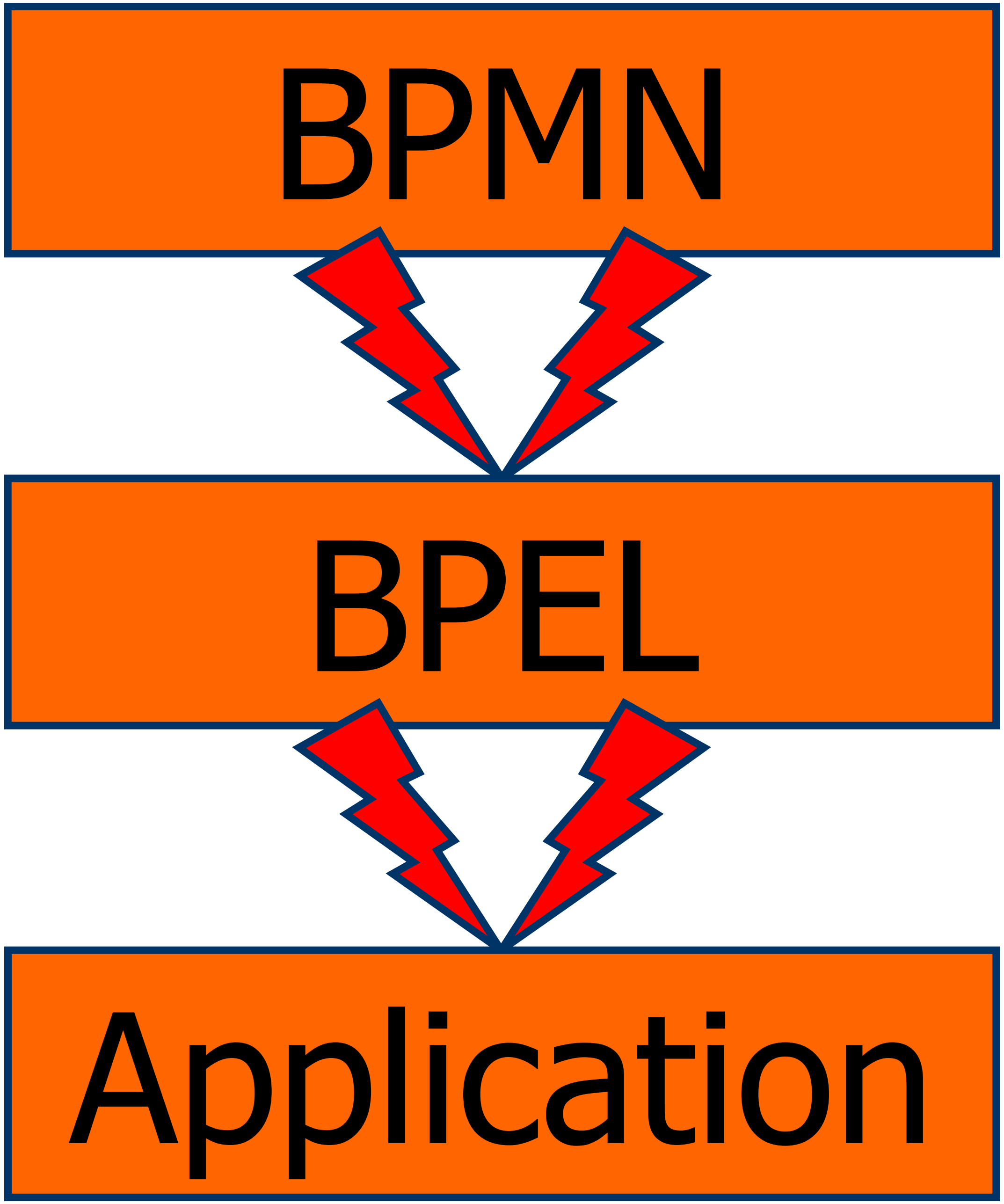
BPEL as Language

- BPEL processes can be executed and thus are programs
- BPEL is a specialized and dedicated programming language
- BPEL combines two tasks
  - Creates a new Web Service which is described by a WSDL interface
  - Implements the Web Service by orchestrating a number of partners

BPEL as Language



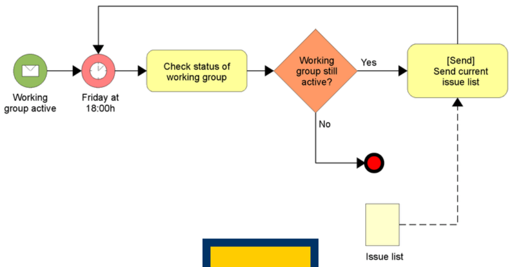
```
<?xml version = "1.0" encoding = "UTF-8" ?>
<process name="BPELDynamicPL"...>
  -->
  <partnerLinks.../>
  <variables>
    <variable name="Receive_File_Get_InputVariable" messageType="ns1:Get_msg"/>
    <variable name="Invoke_FTPServer1_Put_InputVariable" messageType="ns2:Put_msg"/>
    <variable name="jndiLocation" type="xsd:string"/>
  </variables>
  <sequence name="main">
    <receive .../>
    <assign name="Assign_Payload".../>
    <assign name="Assign_FtpServer1_JNDI">
      <copy>
        <from expression="'eis/Ftp/FtpAdapter1'"/>
        <to variable="jndiLocation"/>
      </copy>
    </assign>
    <invoke name="Invoke_FTPServer1"
            inputVariable="Invoke_FTPServer1_Put_InputVariable"
            partnerLink="FTPOut" portType="ns2:Put_ptt" operation="Put">
      <bpelx:inputProperty name="jca.jndi" variable="jndiLocation"/>
    </invoke>

    <assign name="Assign_FtpServer2_JNDI">
      <copy>
        <from expression="'eis/Ftp/FtpAdapter2'"/>
        <to variable="jndiLocation"/>
      </copy>
    </assign>
    <invoke name="Invoke_FTPServer2"
            inputVariable="Invoke_FTPServer1_Put_InputVariable"
            partnerLink="FTPOut" portType="ns2:Put_ptt" operation="Put">
      <bpelx:inputProperty name="jca.jndi" variable="jndiLocation"/>
    </invoke>

  </sequence>
</process>
```



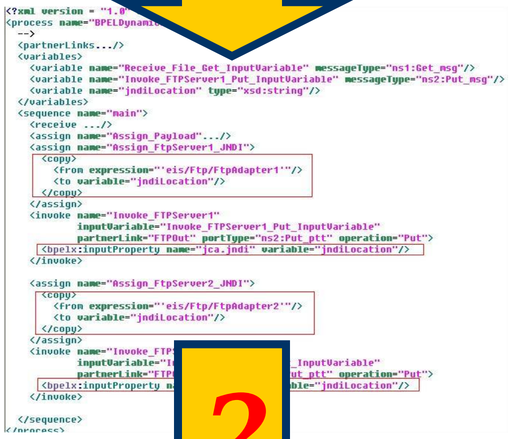
BPMN

BPEL

Application


Working group active
Friday at 18:00h
Check status of working group
Working group still active?
Yes
No
[Send] Send current issue list
Issue list

?



```
<?xml version = "1.0"
<process name="BPELDynami
  -->
  <partnerLinks.../>
  <variables>
    <variable name="Receive_File_Get_InputVariable" messageType="ns1:Get_msg"/>
    <variable name="Invoke_FTPServer1_Put_InputVariable" messageType="ns2:Put_msg"/>
    <variable name="jndiLocation" type="xsd:string"/>
  </variables>
  <sequence name="main">
    <receive .../>
    <assign name="Assign_Payload".../>
    <assign name="Assign_FtpServer1_JNDI">
     <copy>
       <from expression="'eis/Ftp/FtpAdapter1'"/>
       <to variable="jndiLocation"/>
     </copy>
    </assign>
    <invoke name="Invoke_FTPServer1"
            inputVariable="Invoke_FTPServer1_Put_InputVariable"
            partnerLink="FTPOut" portType="ns2:Put_ptt" operation="Put">
     <bpelx:inputProperty name="jca.jndi" variable="jndiLocation"/>
    </invoke>

    <assign name="Assign_FtpServer2_JNDI">
     <copy>
       <from expression="'eis/Ftp/FtpAdapter2'"/>
       <to variable="jndiLocation"/>
     </copy>
    </assign>
    <invoke name="Invoke_FTP
            inputVariable="I                    _InputVariable"
            partnerLink="FTP                   ut_ptt" operation="Put">
     <bpelx:inputProperty n                    ble="jndiLocation"/>
    </invoke>

  </sequence>
</process>
```

?

BPMN

- Process example

from A4B33SI tutorials by Michal Čáp