

Notations for Business Process Modeling



SoftEng
<http://softeng.polito.it>

Objective

- Describe, as precisely as possible, a process (or workflow)
- Communicate, document, analyze, validate the workflow
- Implement (execute) it
 - ◆ Only formal notations allow this step

Issues

- Formal notations
 - ♦ Executable
 - ♦ But model can be very complex for high level of detail
- Semiformal
 - ♦ Not executable
 - ♦ But can be starting point for high level analysis

Notations

- Semi formal
 - ♦ IDEF0
 - ♦ Data Flow Diagrams
 - ♦ UML
 - Class diagrams
 - activity diagrams
 - More recent
 - Formal versions under development
- Formal
 - ♦ BPMN
 - ♦ BPEL

IDEF

- Integrated Computer-aided Manufacturing Definition
- Approach of choice in the 1990s (have been around for over 25 years)
- Only one compliant with Federal Information Processing Standards (FIPS)
 - ◆ FIPS Publication 183

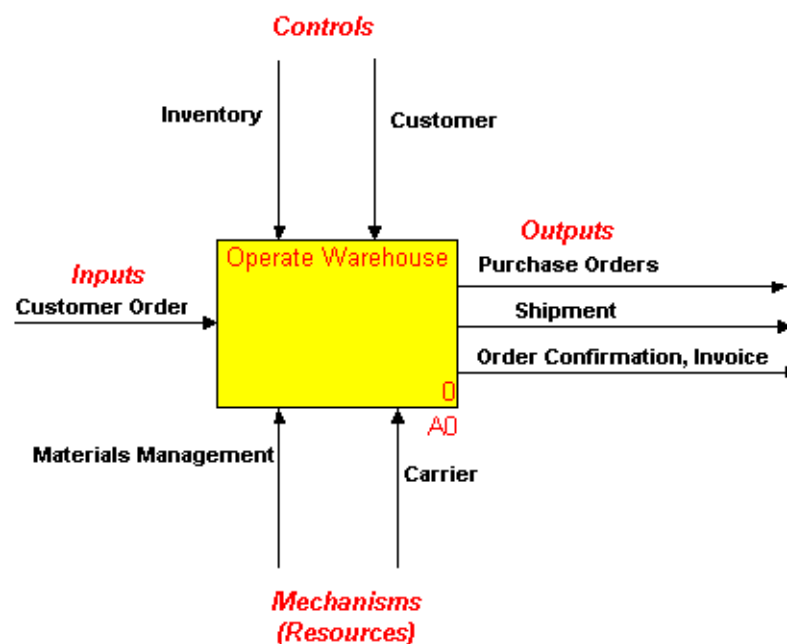
IDEF

- IDEF refers to a group of methods, each of which fulfills a specific purpose
 - ◆ IDEFØ, for example, is used to model an organization's functions
 - ◆ IDEF1x is used for DB modeling

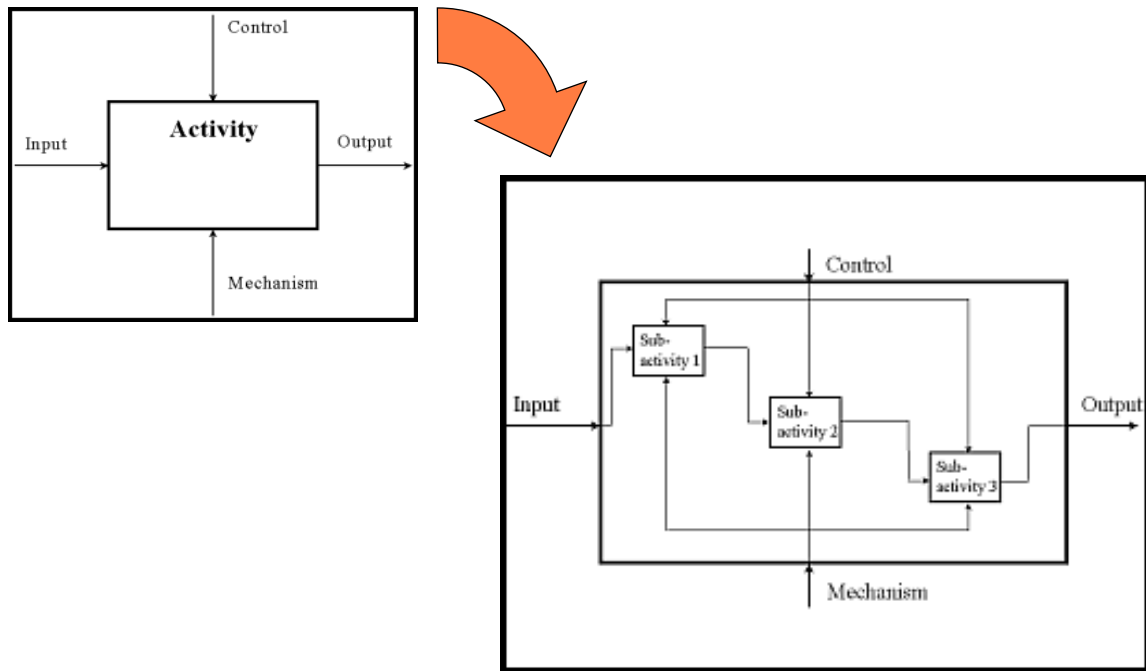
IDEFØ

- “Box and arrow” graphics
 - ♦ function as a box
 - ♦ interfaces to or from the function as arrows entering or leaving the box
- Context diagram (main)
- Constraint diagrams (sub)
- Decomposition

IDEFØ – Example



IDEFØ – Decomposition



DFD

- Data Flow Diagram
 - ♦ Yourdon and Coad
 - ♦ Gane and Sarson

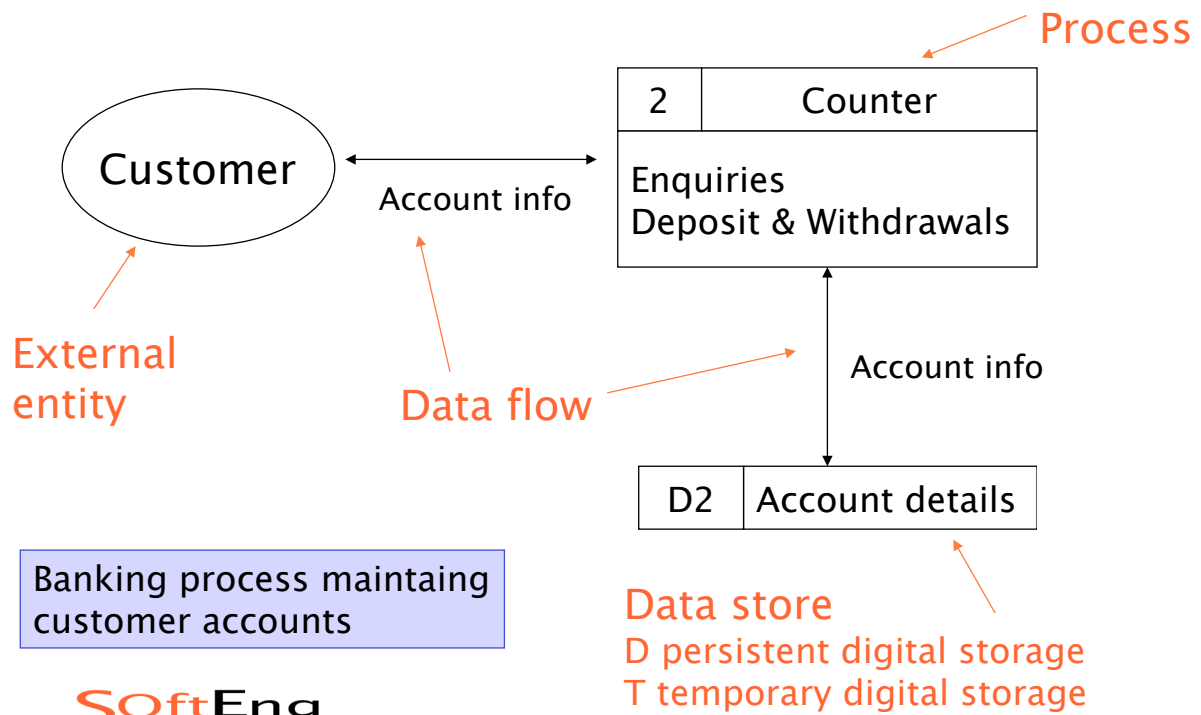
DFD – Decomposition

- Initially a **context diagram** is drawn, which is a simple representation of the *entire system* under investigation
- This is followed by a **level 1 diagram**, which identifies *major business processes* at a high level
- These processes can then be analyzed further with **level 2** diagrams
- And so on...

DFD – Objects

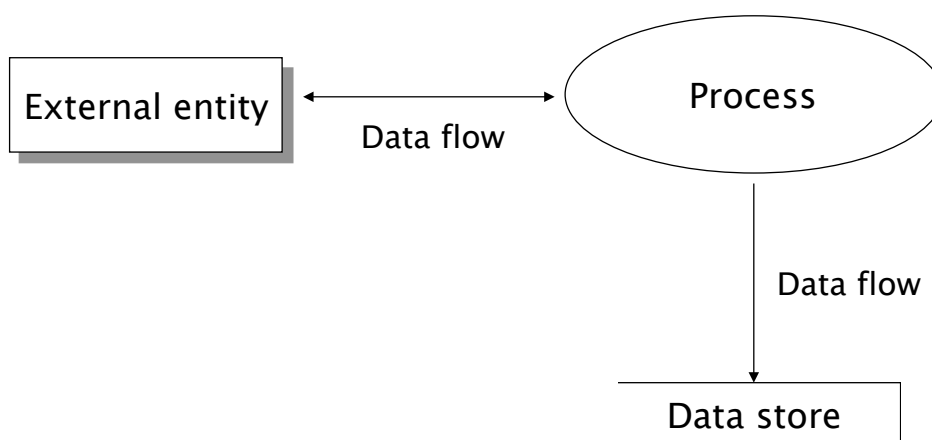
- Process
 - ♦ A process is a unit of work that operates on the data
- Data flow
 - ♦ A data flow is a named flow of data through a system of processes
- Data store
 - ♦ A data store is a logical repository of data
- External entity
 - ♦ An external agent is a source or destination of data

DFD - Example



13

DFD - Example II



14

DFD – Rules

- Data flows are allowed between
 - ♦ different external entities
 - ♦ processes and external entities
 - ♦ processes and data stores
- Data flows are **not** allowed between
 - ♦ external entities and data stores
 - ♦ one data store and another

BPMN

BPMN

- Business Process Modeling Notation
 - ♦ Business Process Diagram (BPD)
- Business Process Management Initiative
 - ♦ <http://www.bpmi.org/>
- Recent (May 2004)
- Endorsed by major players

BPMI.org

- Business Process Modeling Language (BPML)
 - ♦ Meta-language for the modeling of business processes
 - ♦ provides an abstracted execution model for business processes based on FSM
- Business Process Modeling Notation (BPMN)
 - ♦ provides a graphical notation for expressing business processes
 - ♦ provides a binding between graphical elements and the constructs of BPML

BPMI.org

- Business Process Query Language (BPQL)
 - ◆ Defines a standard interface to forthcoming business process management systems (BPMS)
 - ◆ Allows system administrators to manage the BPMS
 - ◆ Allows business analysts to query the instances of business processes it executes

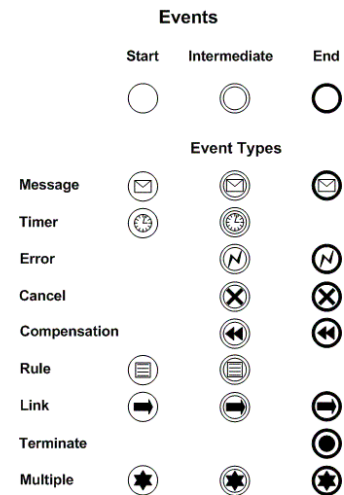
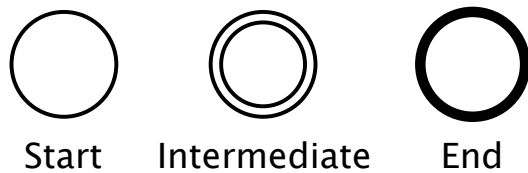
BPMN – Elements

- Four basic element categories
 - ◆ Flow objects
 - ◆ Connecting objects
 - ◆ Swimlanes
 - ◆ Artifacts

BPMN – Flow objects

■ Events

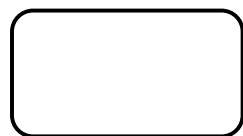
- ◆ Something that happens during the course of a BP



BPMN – Flow objects

■ Activity

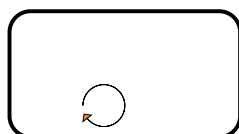
- ◆ A generic term for work that company performs



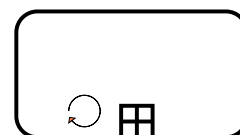
Task



Multiple instances



Loop



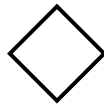
Sub-Process

Decomposition

BPMN – Flow objects

- Gateway

- ♦ Convergence/divergence point for the Sequence Flow



Gateways

Exclusive Decision/Merge (XOR)

Data-Based



Name



Event-Based



Inclusive Decision/Merge (OR)






Complex Decision/Merge



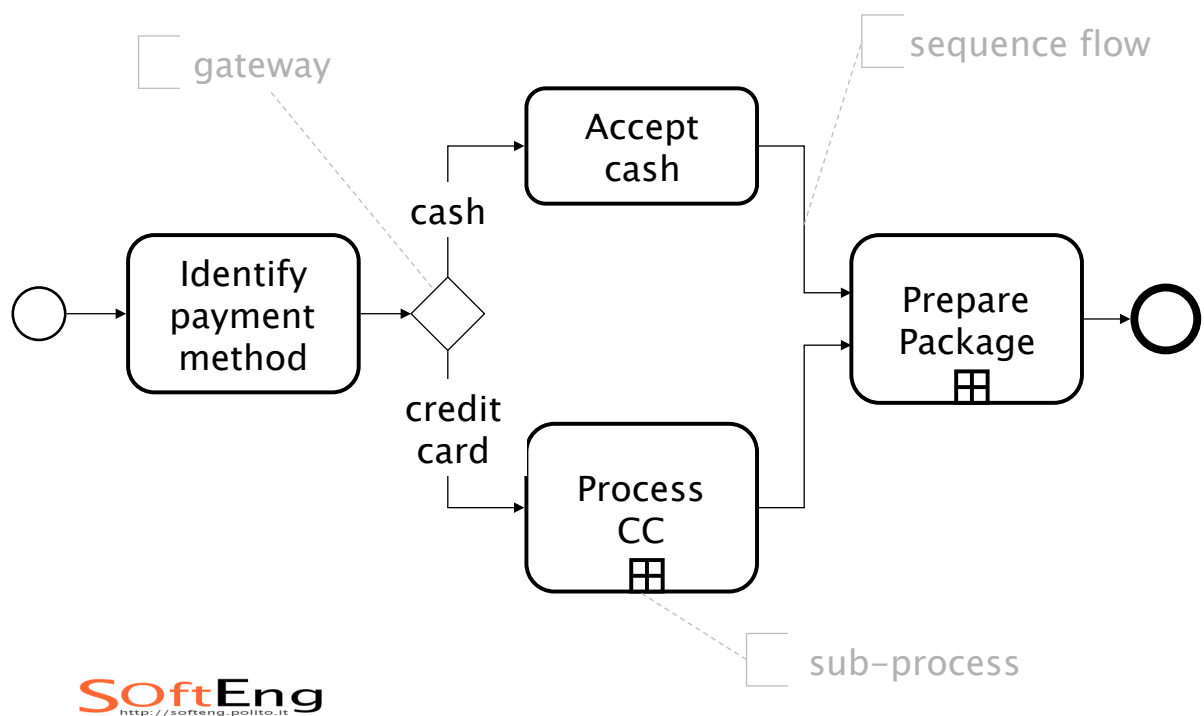
Parallel Fork/Join (AND)



BPMN – Connecting Objects

- Sequence Flow 
 - ♦ Shows the order that activities will be performed in a Process
- Message Flow 
 - ♦ Shows the flow of messages between two separate Process Participants (e.g. two Pools)
- Association 
 - ♦ Associates data, text, and other Artifacts with FlowObjects

BPMN – Example



25

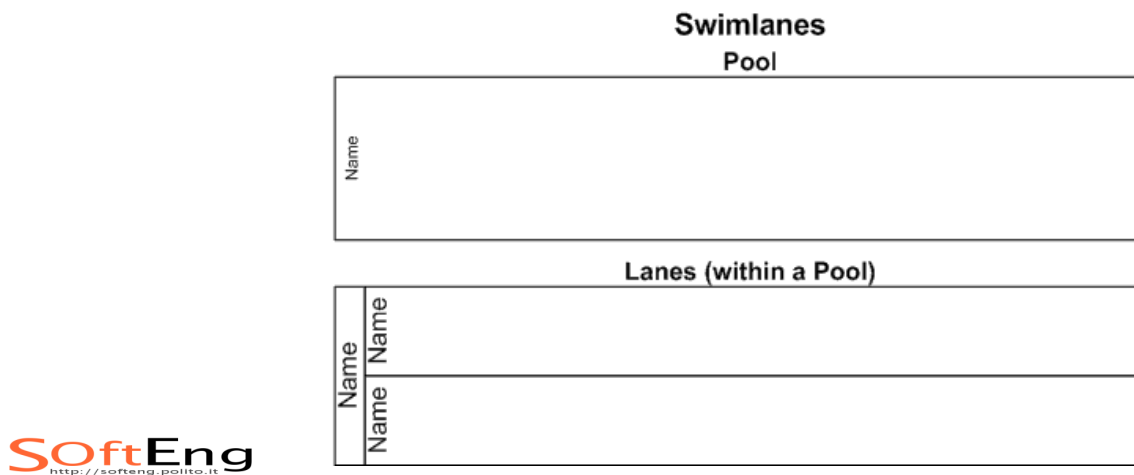
BPMN – Lanes

- A **Pool** represents a Participant (business entity) in a Process
- A **Lane** is sub-partition of a Pool
- Example
 - ♦ Customer
 - ♦ Enterprise
 - Manufacturing
 - Accounting
- Sequence Flow may not cross the boundaries of a pool

26

BPMN – Lanes

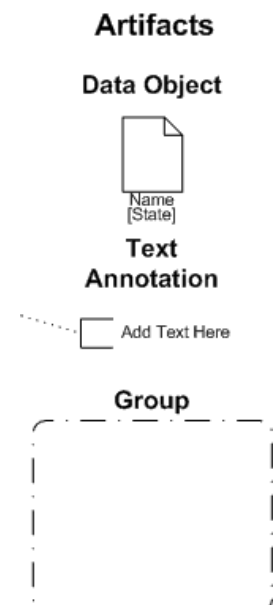
- A **Pool** represents a Participant (business entity) in a Process
- A **Lane** is sub-partition of a Pool



27

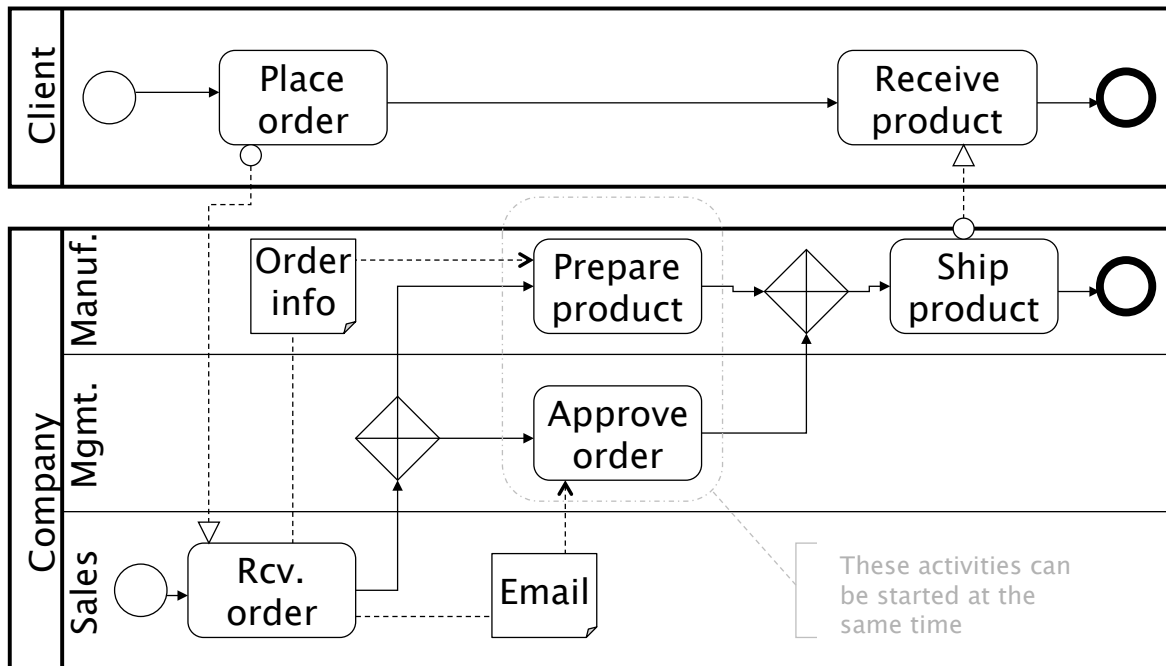
BPMN – Artifacts

- Data Object
 - ♦ Data required or produced by Activities
- Group
 - ♦ No effect, used for documentation
- Text Annotation
 - ♦ Comment connected by Association



28

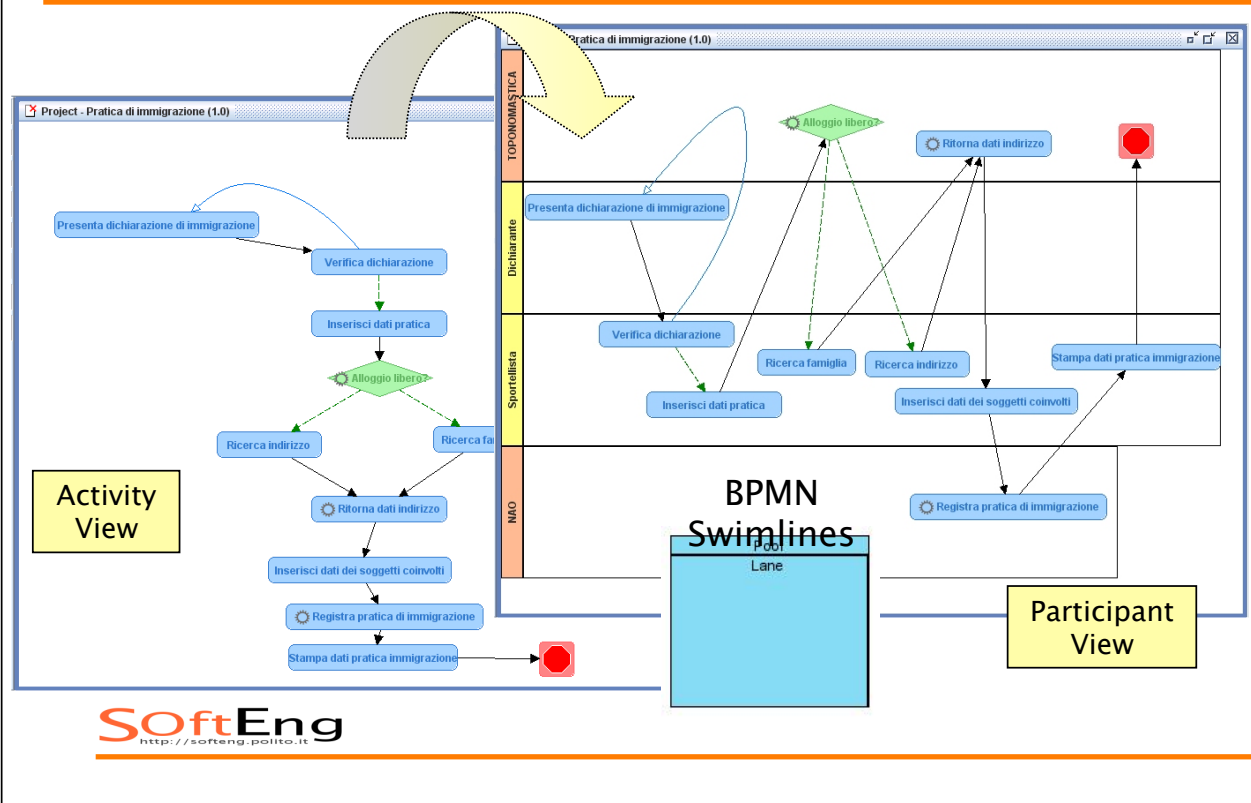
BPMN – Example II



Graphical Symbols

ProEd	Name	Description	BPMN
	Basic Activity	Is shown by blue rounded rectangle - automatic or manual start and various participants	
	Route Activity	Is shown by green diamond - always automatic start and the SYSTEM participant	
	Block Activity	Is shown by violet rounded rectangle - always automatic start and the SYSTEM participant	
	Subflow Activity	Is a fat blue rectangle with a squared plus icon - always automatic start and the SYSTEM participant	
	End Activity	Is shown by a red rounded square with a stop sign - always automatic start and the SYSTEM participant	
	Transition	Ordinary transitions are shown by solid black arrows	
	Conditioned Transition	Transitions that have a condition are shown by dashed green arrows	
	Iteration	Iterations are shown by solid, curved blue arrows; an iteration may also curve back to the same activity it started from	
	Automatic	The gear symbol indicates an activity that has the automatic start mode	-

Activity vs. Participant View



Free Tools

- Bonita
 - ♦ bonita.objectweb.org
 - ♦ Graphical editor (ProEd) + execution engine

BPEL

Some History

- IBM's **WSFL** and Microsoft's **XLANG** were proprietary languages.
- OMG acquired the standardization body **BPML.org** (defining **BPML**, formal BP language based on pi-calculus) and it worked together with the open **BPMS** movement is led by **JBoss** and **Intalio**.
- IBM and Microsoft decided to combine BPML + BPMS into a new language, **BPEL4WS**.
- April 2003, BEA Systems, IBM, Microsoft, SAP and Siebel Systems submitted **BPEL4WS** to **OASIS** for standardization in Web services.
- OASIS WS-BPEL technical committee voted on 2004 to name their spec **WS-BPEL 2.0**, but **BPEL** is commonly used.
- Since 2007, Active Endpoints, Adobe, BEA, IBM, Oracle and SAP published the **BPEL4People** specification
 - ♦ describe how human interaction in BPEL processes can be implemented.
 - ♦ Still under review by OASIS

BPEL orchestration

- BPEL is an orchestration language, not a choreography
 - ♦ An orchestration specifies an executable process that involves message exchanges with other systems, such that the message exchange sequences are controlled by the orchestration designer.
 - ♦ A choreography specifies a protocol for peer-to-peer interactions, defining, e.g., the legal sequences of messages exchanged among peers (WS-CDL).
 - ♦ A choreography can be realized by writing an orchestration (e.g. as a BPEL process) for each peer involved in it.
 - ♦ Orchestration refers to the central control (by the conductor) of the behavior of a distributed system
 - ♦ Choreography refers to a distributed system which operate according to rules but without centralized control.

BPEL features

- BPEL to adopt web services as its external communication mechanism
 - ♦ use of the Web Services Description Language (WSDL) 1.1 to describe outgoing and incoming messages.
- A property-based message correlation mechanism
- XML and WSDL typed variables
- An extensible language to allow writing expressions and queries in multiple languages (XPath)
- Structured-programming constructs including if-then-elseif-else, while, sequence (to enable executing commands in order) and flow (to enable executing commands in parallel)
- A scoping system to allow the encapsulation of logic with local variables, fault-handlers, compensation-handlers and event-handlers
- Serialized scopes to control concurrent access to variables

Relationship of BPEL to BPMN

- There is no standard graphical notation for WS-BPEL.
 - Some vendors have invented their own notations;
 - Others have proposed to use BPMN as a graphical front-end
- BPMN specification includes an informal and partial mapping from BPMN to BPEL 1.1.
- Fundamental differences between BPMN and BPEL
 - make it very difficult (sometimes impossible) to generate BPEL code from BPMN models.

Summary

- Various notations to express business processes
- UML from IT world
 - ♦ Defacto standard in sw development, but limited support for workflows
- BPMN, BPEL from business community
 - ♦ Execution supported
 - ♦ No single standard yet, competition between big players
 - ♦ Open standards: XPDL