MACIASZEK, L.A. (2007):
Requirements Analysis and System Design, 3rd ed.
Addison Wesley, Harlow England
ISBN 978-0-321-44036-5

Chapter 2
Requirements Determination

Topics

- From business processes to solution envisioning
- Functional and nonfunctional requirements
- Requirements elicitation
  - traditional methods and modern methods
- Requirements negotiation and validation
- Requirements management
- Requirements business model
  - system scope, business use case model, business glossary, business class model
- Requirements document

1. From business processes to solution envisioning

- IT solutions address business problems
  - IT solutions are enablers of business innovation
- IT solution is an infrastructure service (a commodity)

What is an IT solution?

- A *business solution*
- Implementation of a *business process*
- Sometimes an enabler of *business innovation*
- An *infrastructure service*
- A *commodity* (http://en.wikipedia.org/wiki/Commodity)
  - “derived from the French word "commodité" , meaning today's "convenience" in term of quality of services
  - “an undifferentiated product whose value arises from the owner's right to sell rather than buy” – like electricity or music

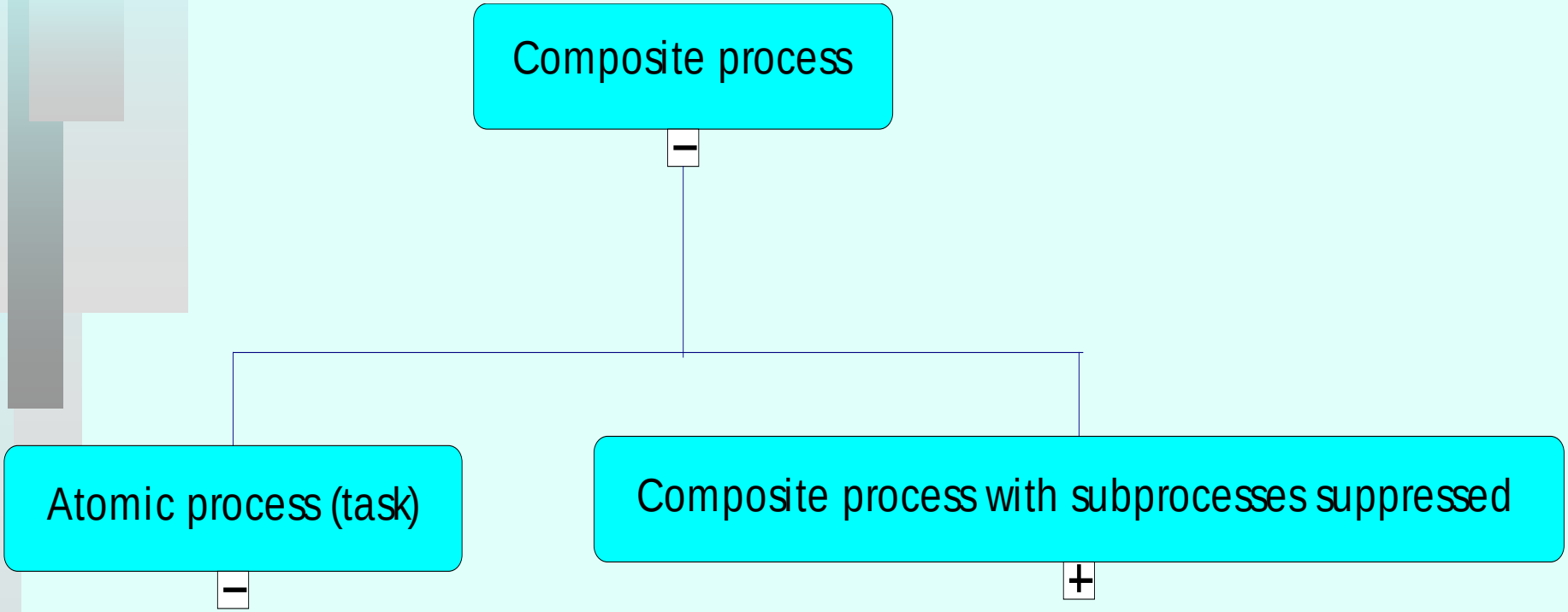
BPMN - Business Process Modeling Notation

- Developed by a consortium Business Process Management Initiative
- Managed by the OMG within the task force Business Modeling & Integration Domain Task Force (BMI DTF)
- Dedicated to modeling business *processes* defined as activities that produce something of value to the organization or to its external stakeholders
- UML activity diagrams – a competing notation
- A goal is to map these notations to an executable language
  - in particular to the Business Process Execution Language (BPEL) – a standard for process execution in systems based on the Service Oriented Architecture (SOA)

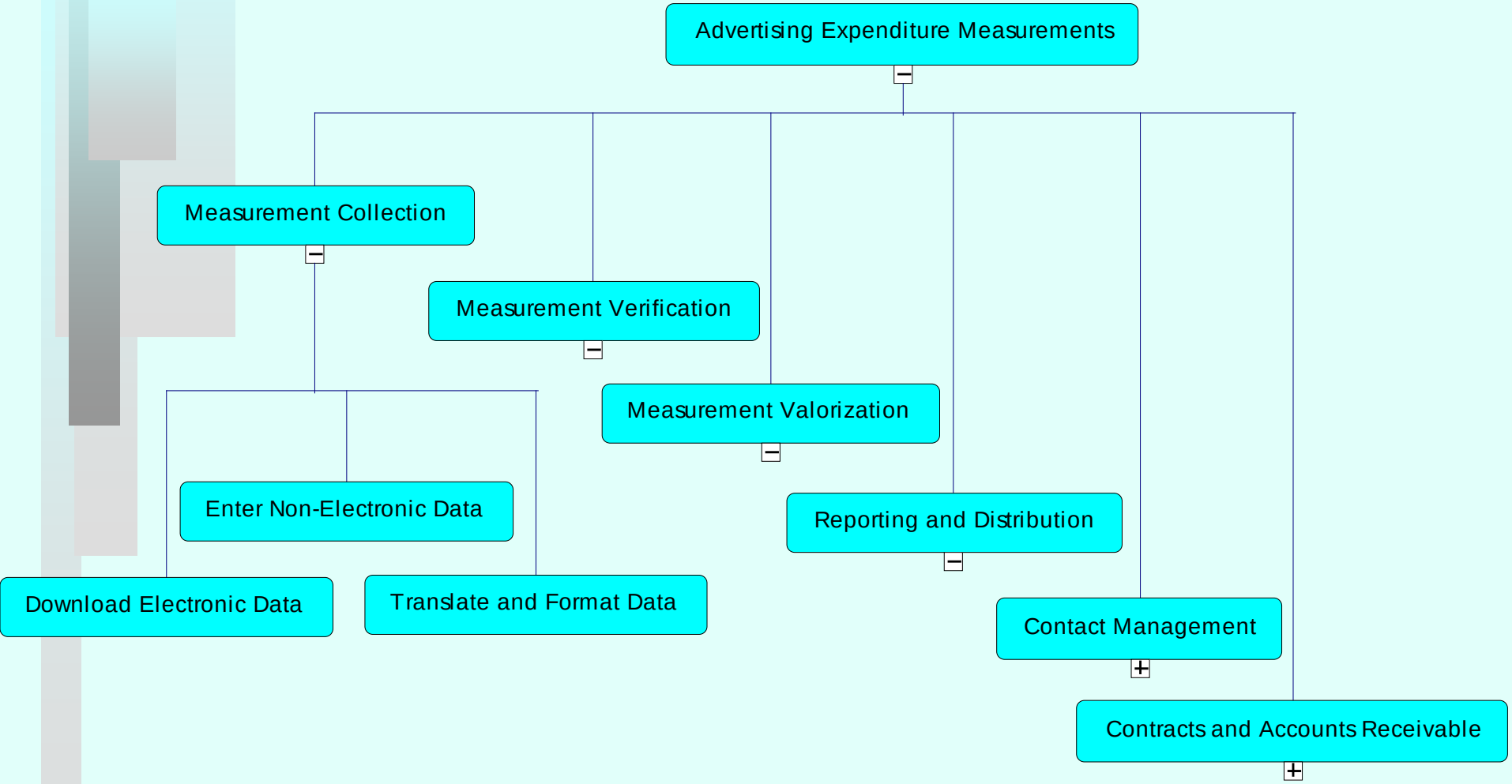
Process hierarchy modeling

- A process may contain other processes (***sub-processes***)
- An atomic activity within a process is called a ***task***
- A ***process hierarchy diagram*** – not part of BPMN
  - A business process can be performed manually or an automated service
  - A process has at least one input flow and one output flow
    - When the process gains the control, it performs its activity, typically by transforming the input flow into the output flow
  - A process can be atomic or composite

Process hierarchy diagram - notation



Process hierarchy model (advertising expenditure)



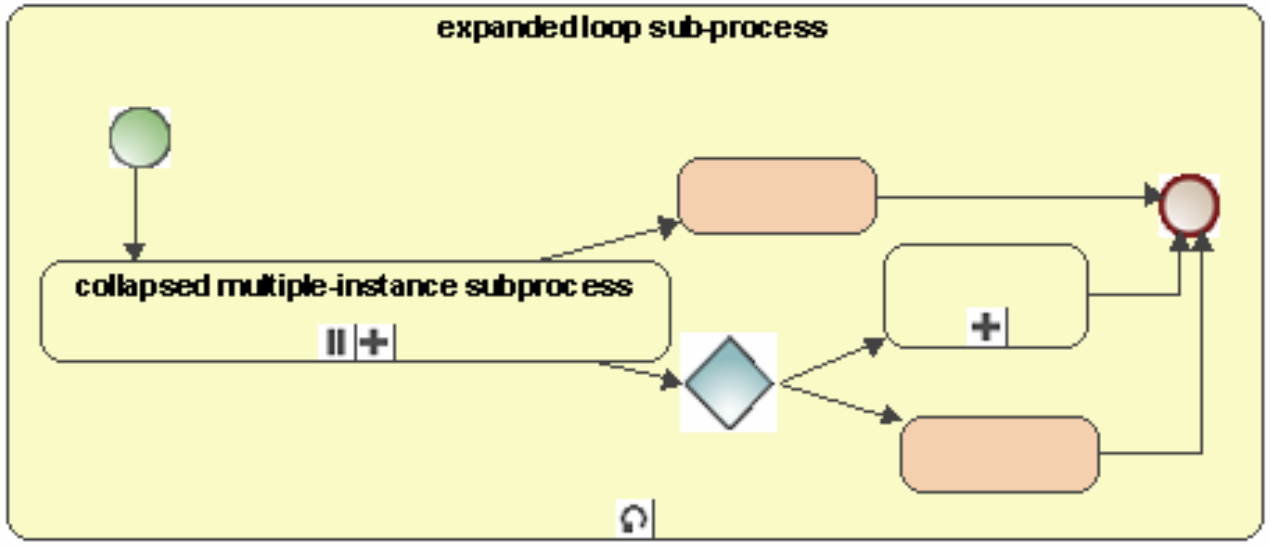
BPMN flow objects

- BPMN offers four basic categories of modeling elements:
  - Flow objects:
    - Events
    - Activities
    - Gateways
  - Connecting objects
  - Pools (Swimlanes)
  - Artifacts

BPMN – events and activities

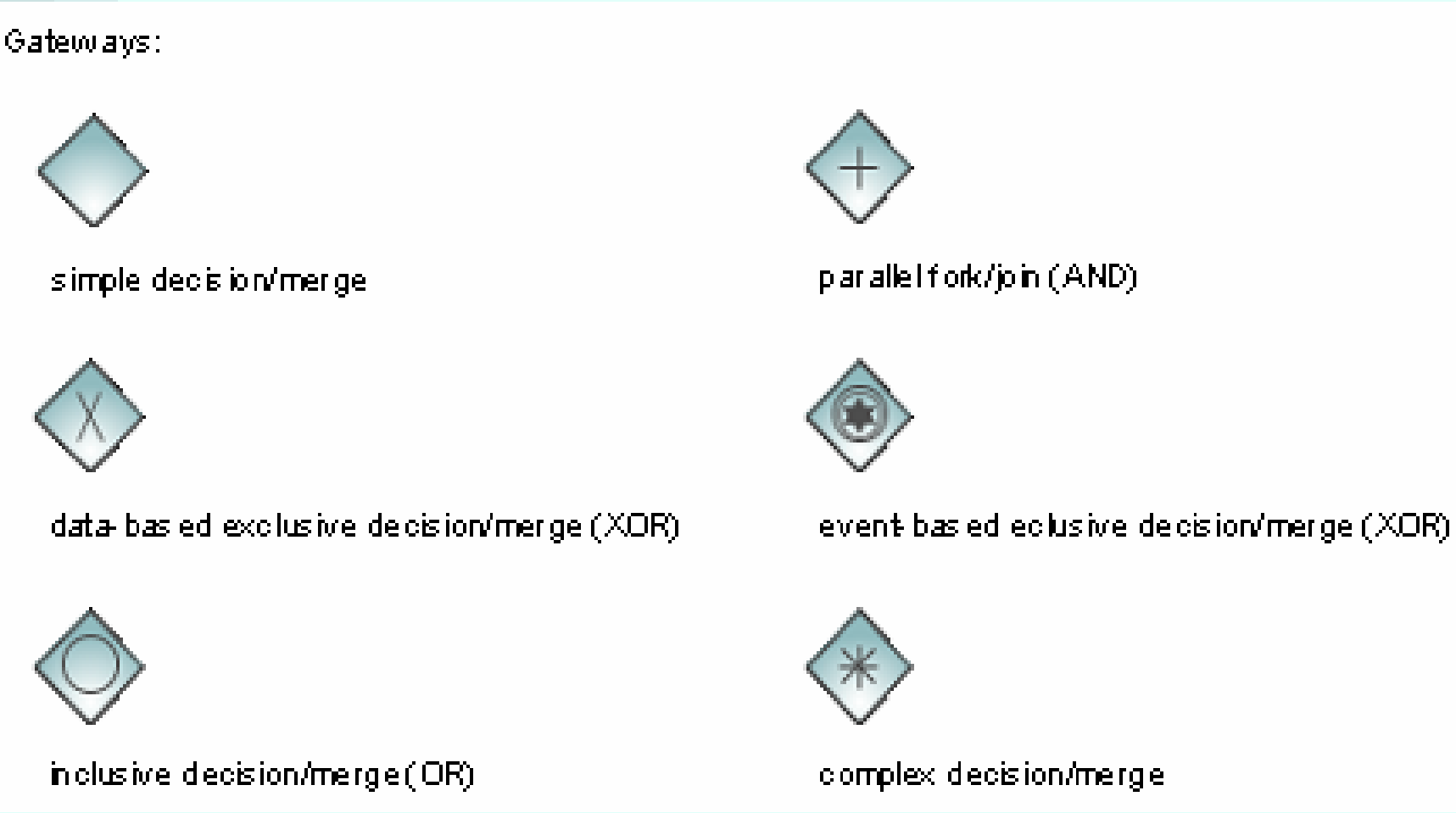
- An *event* is something that "happens".
- There are various types of events, such as timer, error, or cancel.

- An *activity* represents some work that must be done.
- This could be a task or a sub-process.



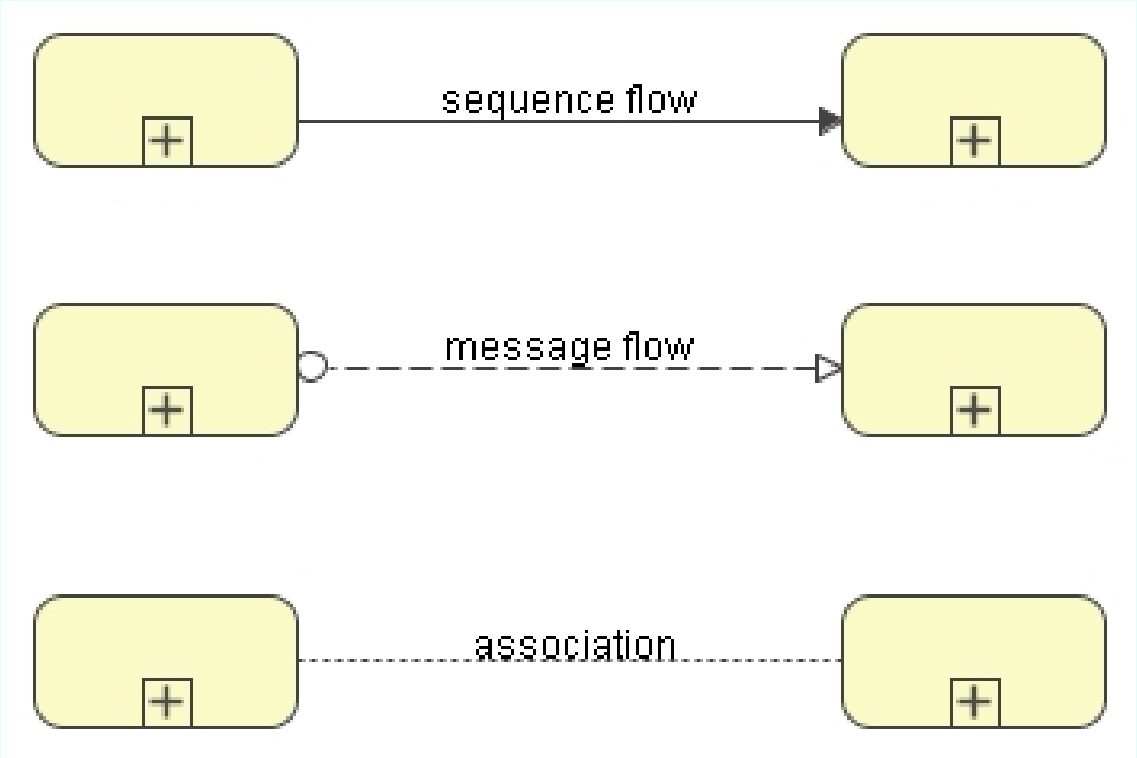
BPMN - gateways

A *gateway* is used to control the divergence and convergence of multiple sequence flows.



BPMN – connecting objects

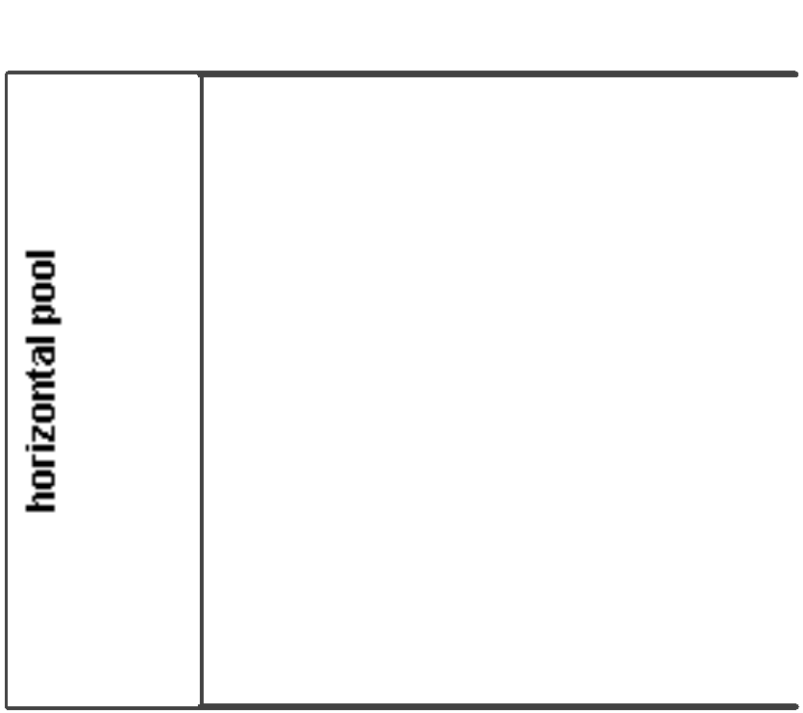
The *connecting objects* (or connectors) are used to connect flow objects in order to define the structure of a business process.

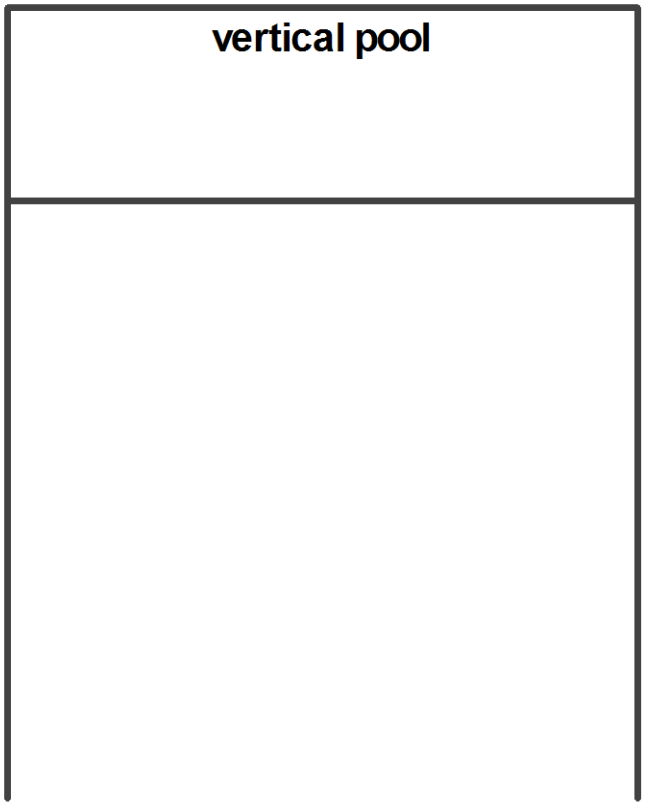


• A *sequence flow* is used to show the order in which the activities will be performed in a process.
• A *message flow* is used to show the flow of messages (data) between two business entities (two process participants) that are prepared to send and receive them.
• An *association* is used to associate flow objects or to associate artifact with a flow object.

BPMN – pools (swimlanes)

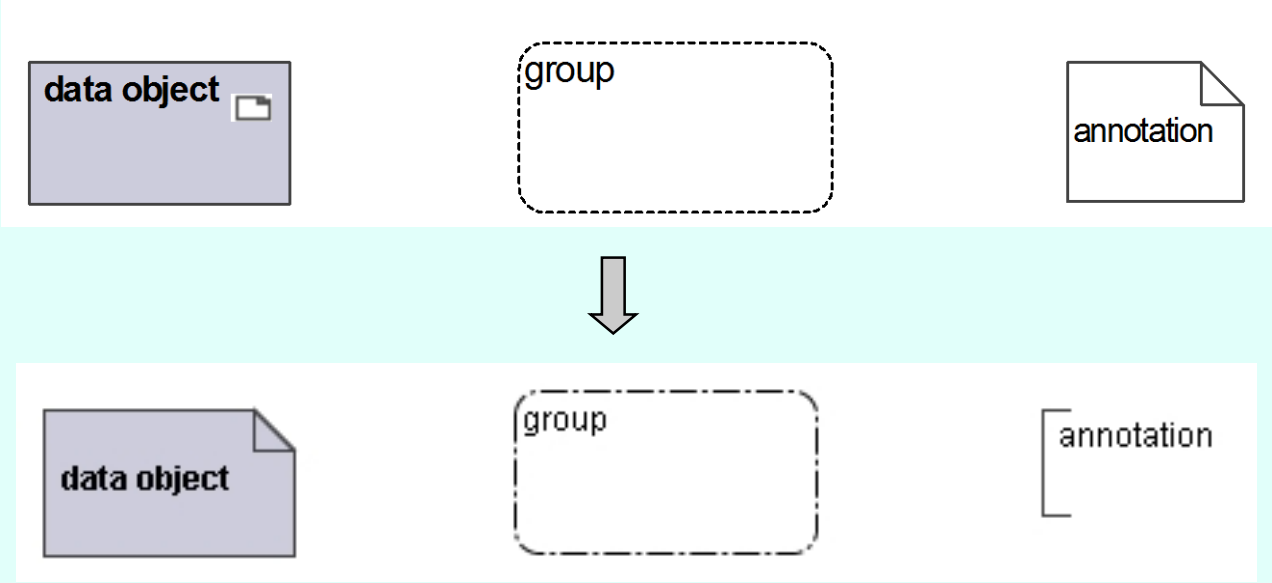
• A *pool* represents a business entity (participant) in a process.
• It acts as a “swimlane” mechanism to organize activities into separate visual categories in order to illustrate different functional capabilities or responsibilities.
• Pools represent sets of self-contained processes.
• Accordingly, the sequence flow may not cross the boundary of a pool.
• Participants in various pools can communicate via message flows or associations to artifacts.





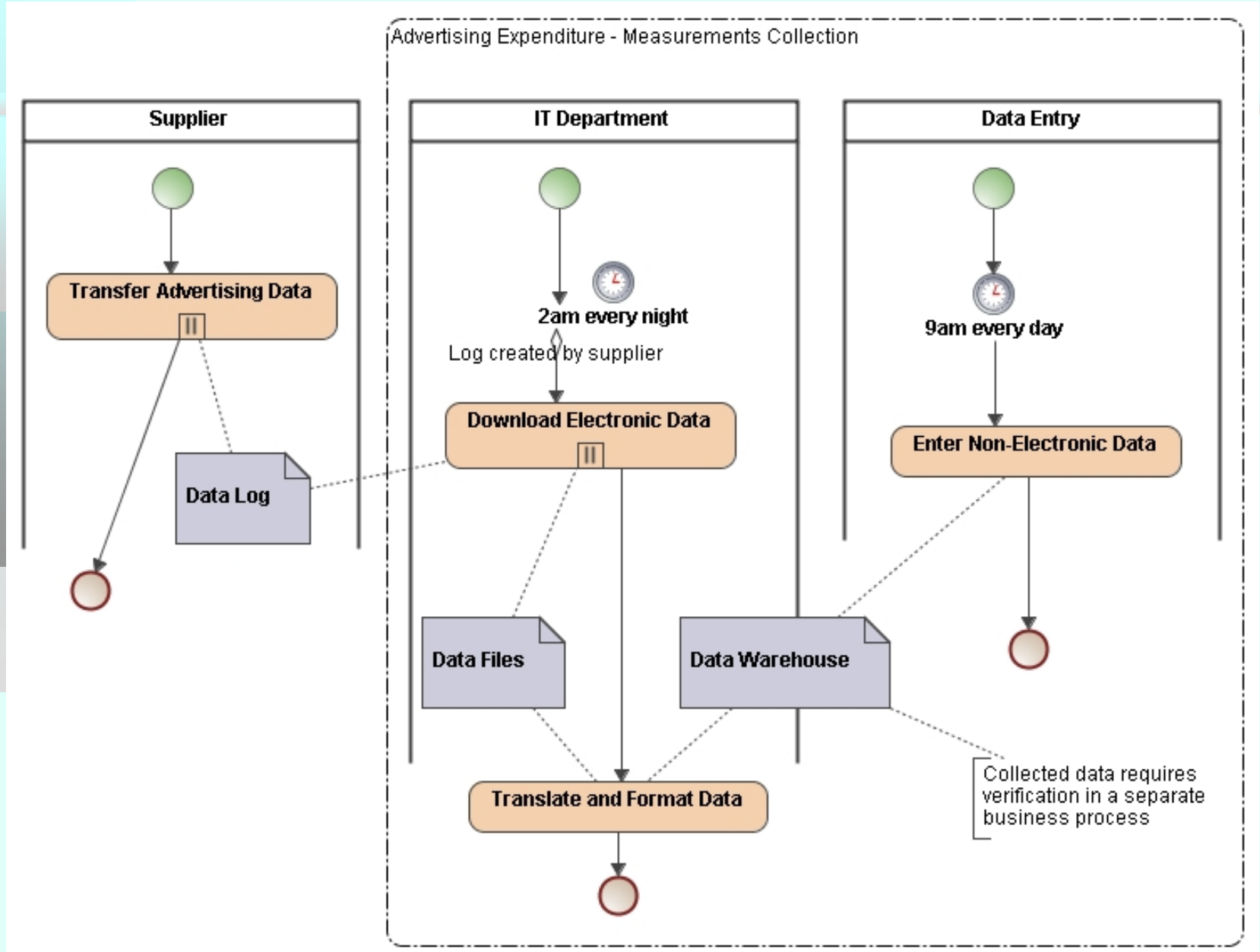
BPMN - artifacts

Artifacts provide additional modeling flexibility by allowing to extend the basic notation to respond to specific modeling situations, such as for so-called vertical markets (e.g. telecommunication, hospitals or banking).



• *Data objects* represent data required or produced by activities.
• A *group* is a grouping of activities that does not affect the sequence flow of the process. The grouping can be used for documentation or analysis purposes (e.g. to identify the activities of a distributed transaction across pools).
• *Annotations* provide additional text information for the reader of a business process diagram.

Business process model (advertising expenditure)



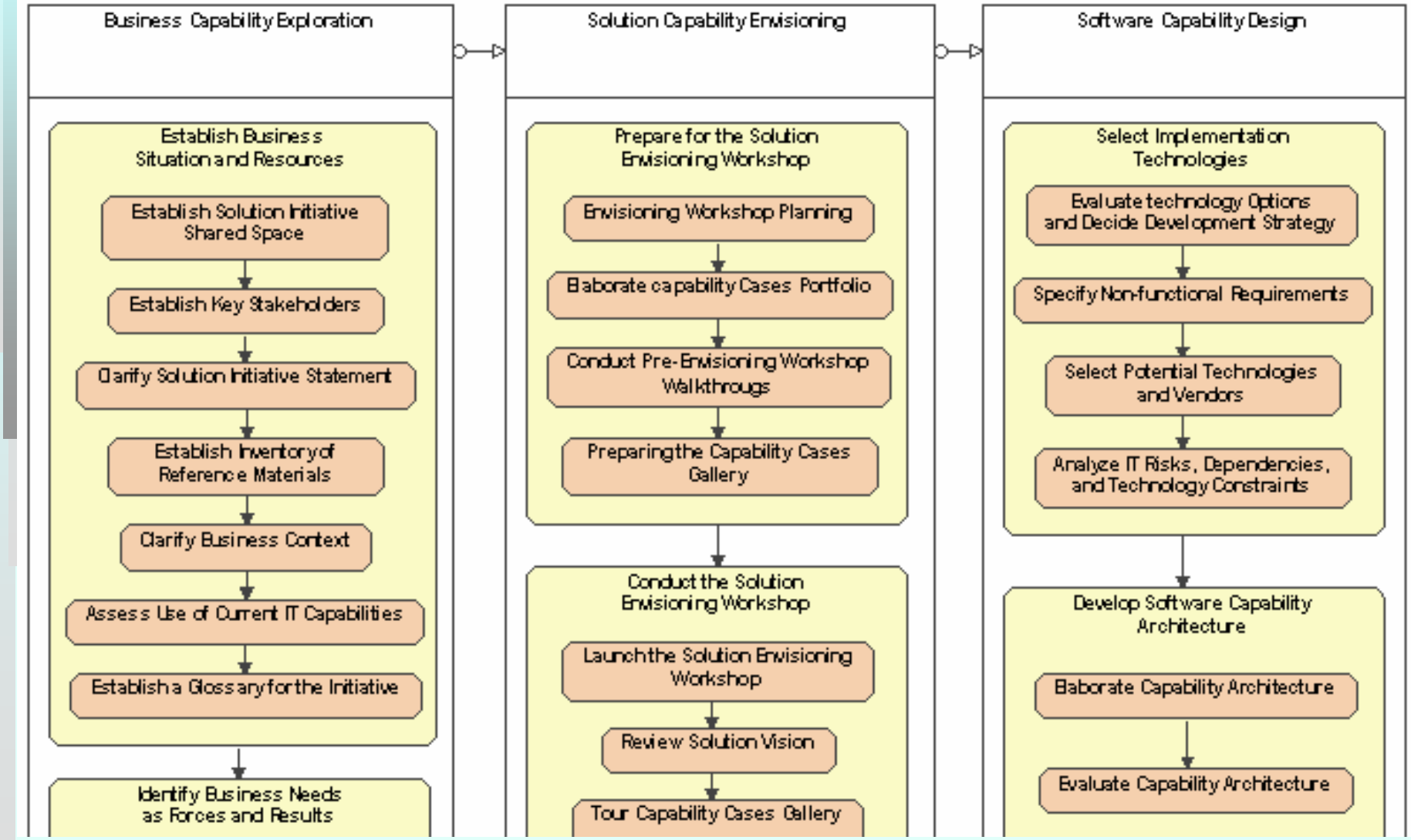
Solution envisioning

- *Solution envisioning* is a business value-driven approach to delivering an IT service (i.e. not merely a software system)
  - to solve an As-Is business problem or
  - to foster a To-Be business innovation.
- Solution envisioning makes a close connection between business and IT stakeholders and integrates business strategy methods and software development capabilities.
- It is about the three “E-s” – efficiency, effectiveness, and edge.

Three phases of solution envisioning process

- ***Business capability exploration*** – determines *business capabilities* understood as the capacities relating to how a business IT solution can deliver specific results.
  - This phase describes *capability cases* – solution ideas making the business case for each capability.
- ***Solution capability envisioning*** – aims at developing the capability case into a *solution concept* and at ensuring that the solution is agreed upon by the stakeholders.
  - The solution concept takes the business context as input and produces future scenarios for new ways to work as output.
  - The solution concept converges on the ultimate solution architecture and is developed in solution envisioning workshops.
- ***Software capability design*** – decides on system implementation technologies, develops software *capability architecture*, and elaborates the business case with project plans and risk analysis.
  - Software capability design is an activity in software modeling.

Three phases of solution envisioning process



Implementation strategy and capability architecture

- In the first phase of the solution envisioning process, capability cases function as multiple *solution sketches* to allow many solution possibilities to be explored.
- Later on, the selected capability cases become technical *blueprints* for the solution.
- Finally, three prevalent *implementation strategies* need to be considered:
  - *Custom development*
    - performed in-house and/or
    - contracted out to consulting and development firms.
  - *Package-based development* that derives the solution by customizing pre-existing software packages, such as COTS, ERP or Customer Relationship Management (CRM) systems.
  - *Component-based development* that builds the solution by integrating software components sourced from multiple vendors and business partners and likely to be based on SOA and/or MDA.

Review Quiz 2.1

1. What is the name of the most popular language for visual modeling of business processes that aims at bridging the gap between business and IT people?
2. What are the four categories of modeling elements in BPMN?
3. Can a sequence flow connect two pools?
4. What is the name of a business value-driven approach to delivering an IT service to solve an As-Is business problem or to foster a To-Be business innovation?
5. What is the main modeling outcome of software capability design?
6. What are the three distinct implementation strategies to be considered in solution envisioning process