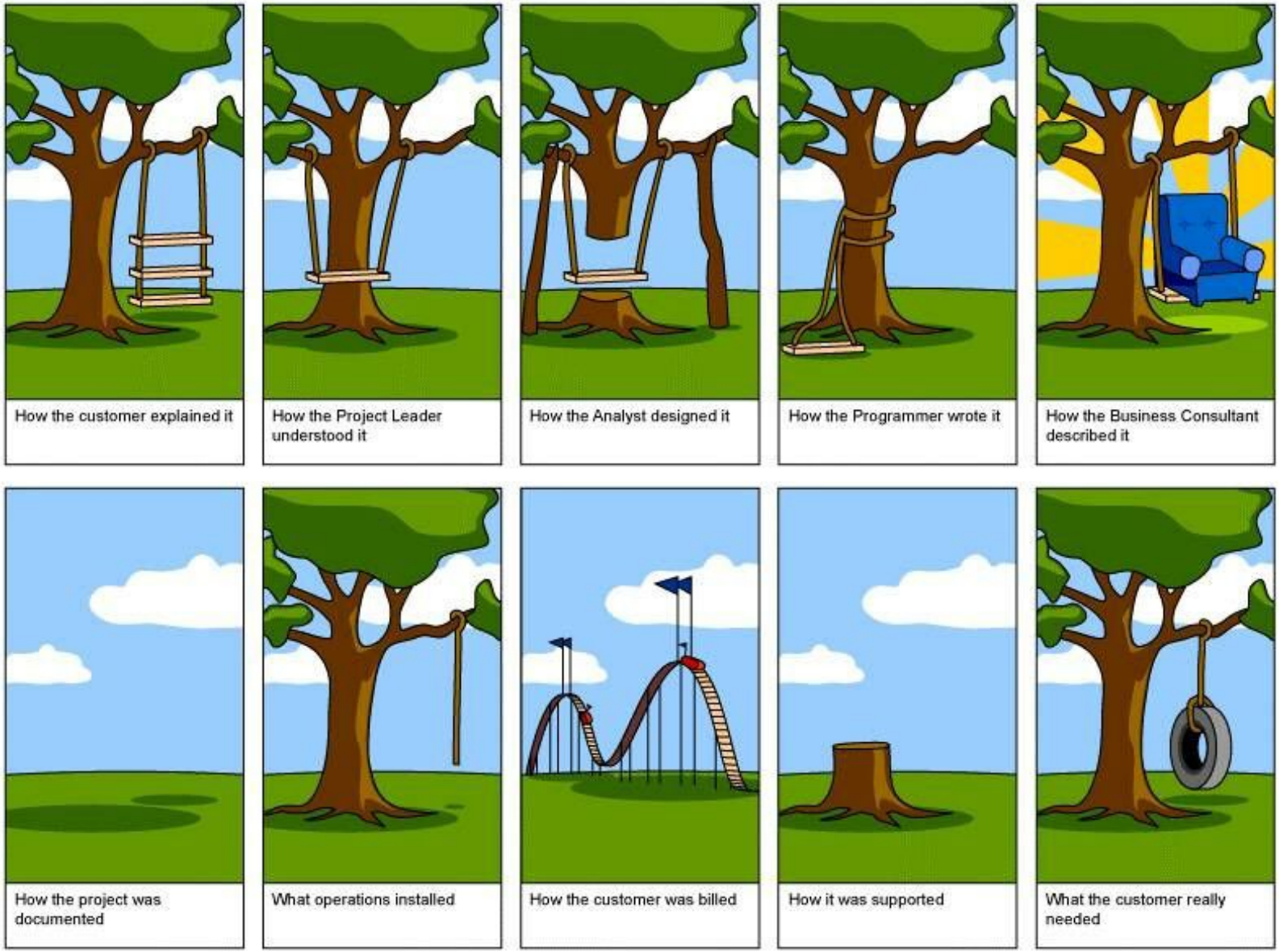
SIMULATION MODELS CONCEPTUALIZATION

Pau Fonseca i Casas; pau@fib.upc.edu

The need of a conceptual model



Hypotheses

- What is inside the model?
- Hypotheses
  - Systemic / Structural
  - Simplification

Advantages of use a conceptual model

- Textual specification is less precise.
- Conceptual model have in a detailed manner, the dynamical relations between the different elements of the interest process.
  - Constitutes an specification by itself.
- Simplifies the dialog between the different parts that are involved in the project.
- Constitutes a representation of the simulation model independent of the selected tool used to build the model.

Conceptual model formalization

- Formalism must be **independent** from the simulation tools.
- The formalized model must **allow** some **analysis.**
  - To **determine relations** between components.

Conceptual model formalization

- Formalism must allow an easy transformation to the representations supported by the existing simulation frameworks.
  - Simplify the codification process.
  - To evaluate alternatives.

Conceptual model formalization

- Some aspects of the model can be no specified, without causing problems in the transformation to other representations. MODULARITY
- The model must be defined in terms that **no constrain** its codification in a **particular mechanism** of simulation **clock update**.

Modularity

- The capacity to describe the behavior of each subsystem, independent from the other subsystems that compose the model
  - Incremental design of the model.
  - Simplifies the verification and the validation of the model.

Assure the Modularity

1. A module cannot access directly to the state of other modules or components.
2. A module must own a set of ports (input/output) to allow the interaction with the other parts of the model.

Conceptual models

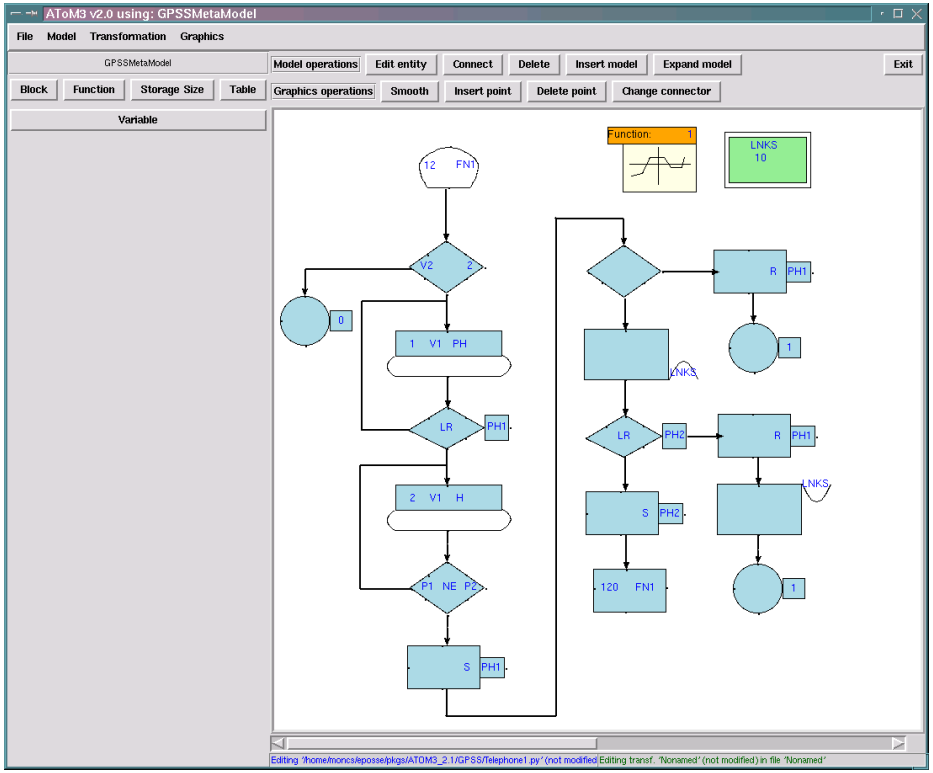
- Flow models.
- Queue networks.
- BPM.
- Service Blueprint.
- Petri nets.
- Colored Petri nets.
- SDL language
- DEVS
- Causal and Forrester diagrams.

Working with different formal languages

- Three of the main mechanisms for doing this:
  - Meta-formalism.
  - Common formalism.
  - Co-simulation.
- Vangheluwe, H. L. (2000). DEVS as a common denominator for multi-formalism hybrid systems modelling. *IEEE International Symposium on Computer-Aided Control System Design* (pp. 129--134). IEEE Computer Society Press.

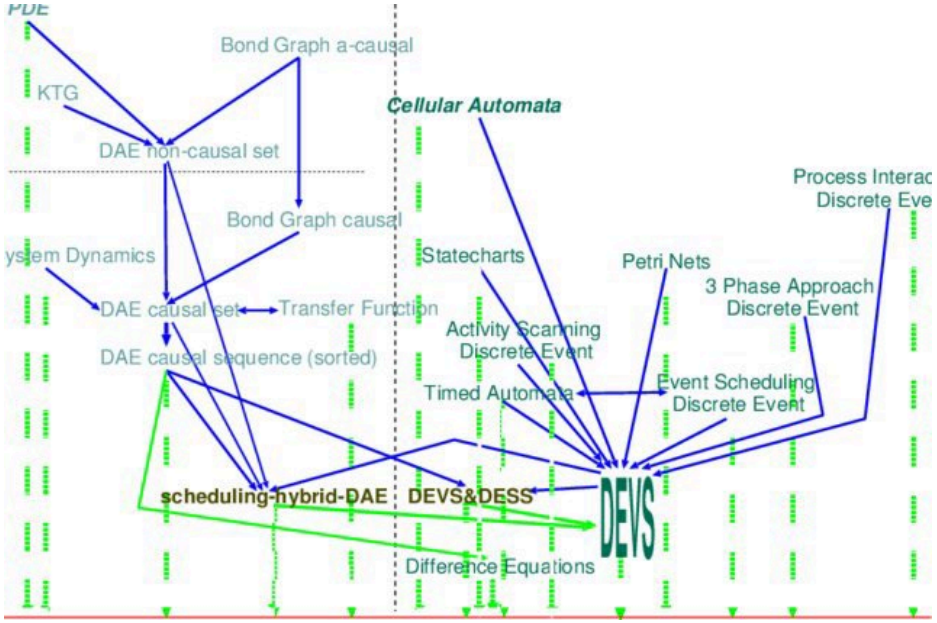
Meta-formalism

- A formalism that incorporates the different formalisms of the various sub models that makes up the system.
- ATOM3: http://atom3.cs.mcgill.ca/



Common formalism

- A mechanism that converts all formalisms to a common formalism.
- Transforming algorithms from:
  - SDL→DEVS→Petri Nets…



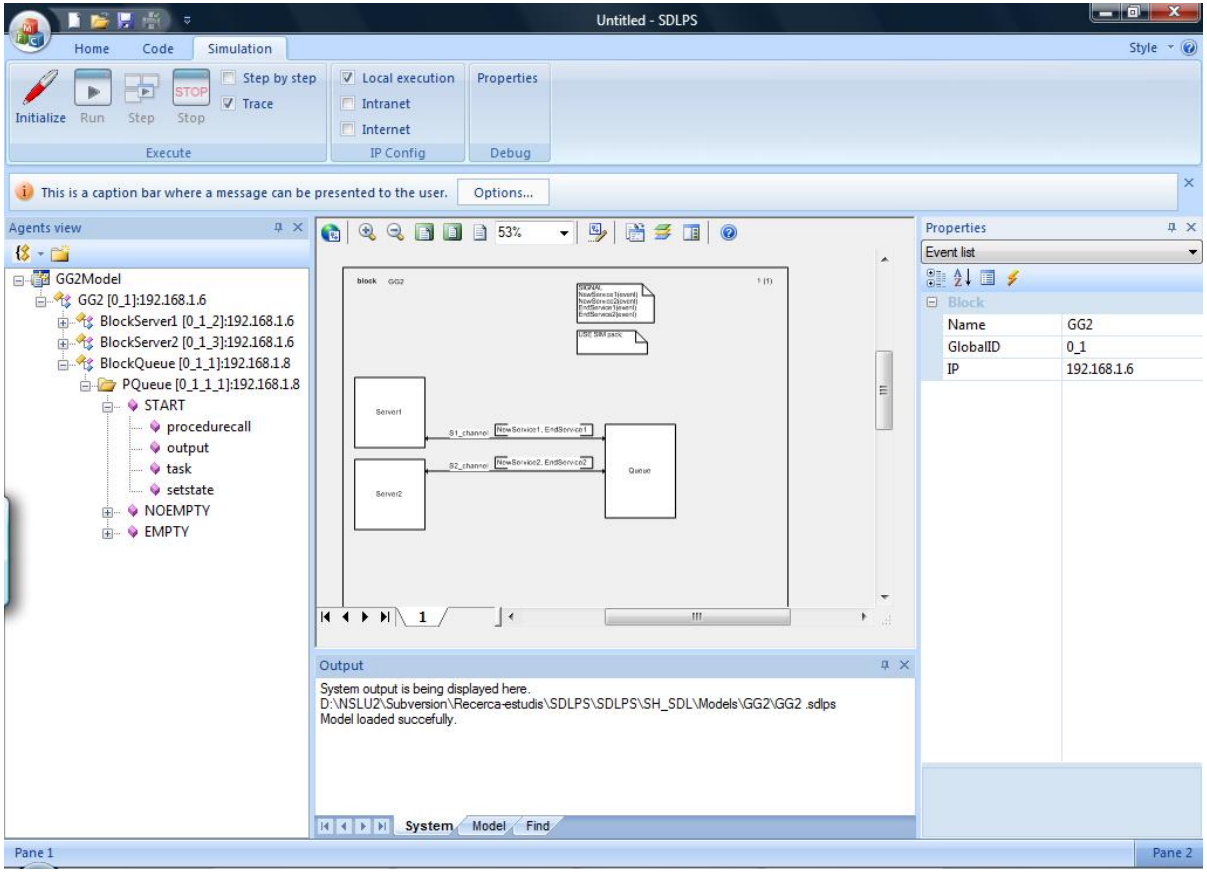
https://www.researchgate.net/publication/220713394_Computer_Aided_Multi-paradigm_Modelling_to_Process_Petri-Nets_and_Statecharts

Co-simulation

- Independent simulators that work together
- HLA: The **High-Level Architecture** (**HLA**) is a general-purpose architecture for distributed computer simulation systems. Using HLA, computer simulations can interact to other computer simulations regardless of the computing platforms. The interaction between simulations is managed by a Run-Time Infrastructure (RTI).

Co-simulation with SDL

- We use SDLPS (on the practical sessions)



https://sdlps.com/

Flow models

Simulation models formalization

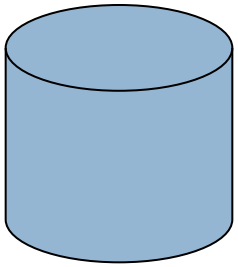
Flow diagrams

- Flow chart or flow diagram... is a diagram that visually displays interrelated information such as events, steps in a process, functions, etc., in an organized fashion, such as sequentially or chronologically.
- Flow diagram [is] a graphic representation of the physical route or flow of people, materials, paperwork's, vehicles, or communication associated with a process, procedure plan, or investigation.

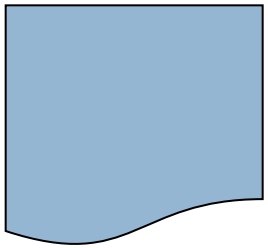
- See *Information Graphics: A Comprehens*investigation*ive Illustrated Reference* by Harris (1999)

https://en.wikipedia.org/wiki/Flow_diagram

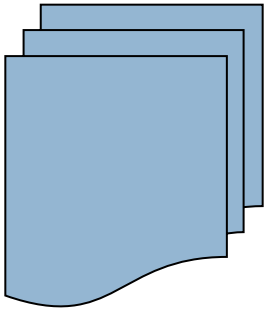
Flow models (data)



- Magnetic disc



- Document

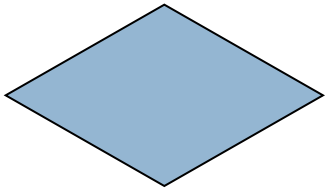


- Multiple document

Flows models (Processes)

- State

- Process

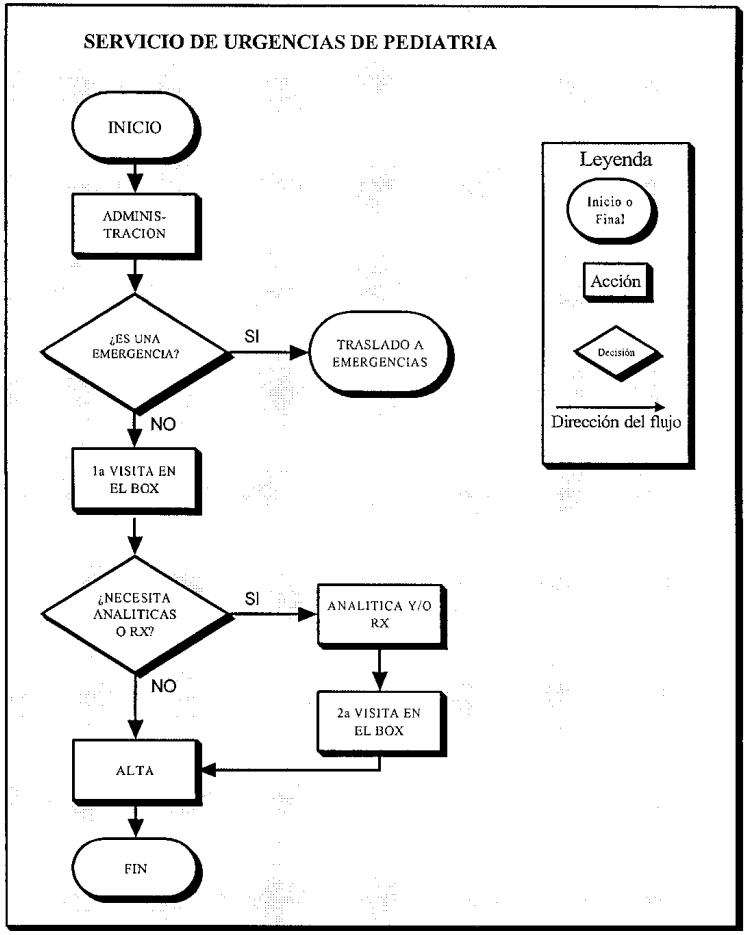


- Decision point

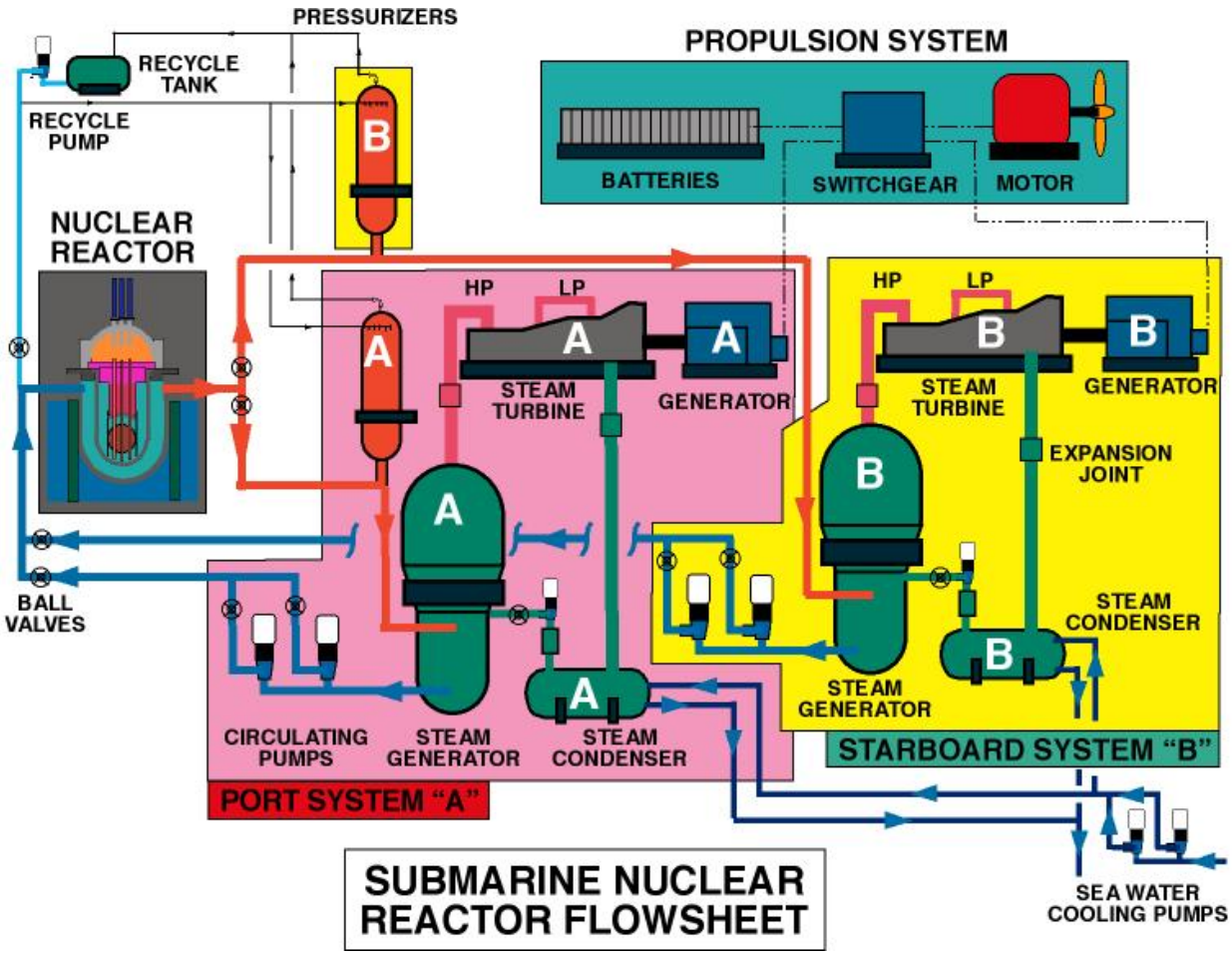
Pediatrics example

- Models a pediatrics example.
- If a new emergency arrives a special process takes cure of it.
- If X ray is needed, or blood analysis, is done in a second visit
- Finally the patient release the system.

Flows models



Submarine nuclear reactor



"SUB REACTOR SYSTEM FLOW". Licensed under Public Domain via Wikimedia Commons - http://commons.wikimedia.org/wiki/File:SUB_REACTOR_SYSTEM_FLOW.jpg#mediaviewer/File:SUB_REACTOR_SYSTEM_FLOW.jpg

Flows models

Good

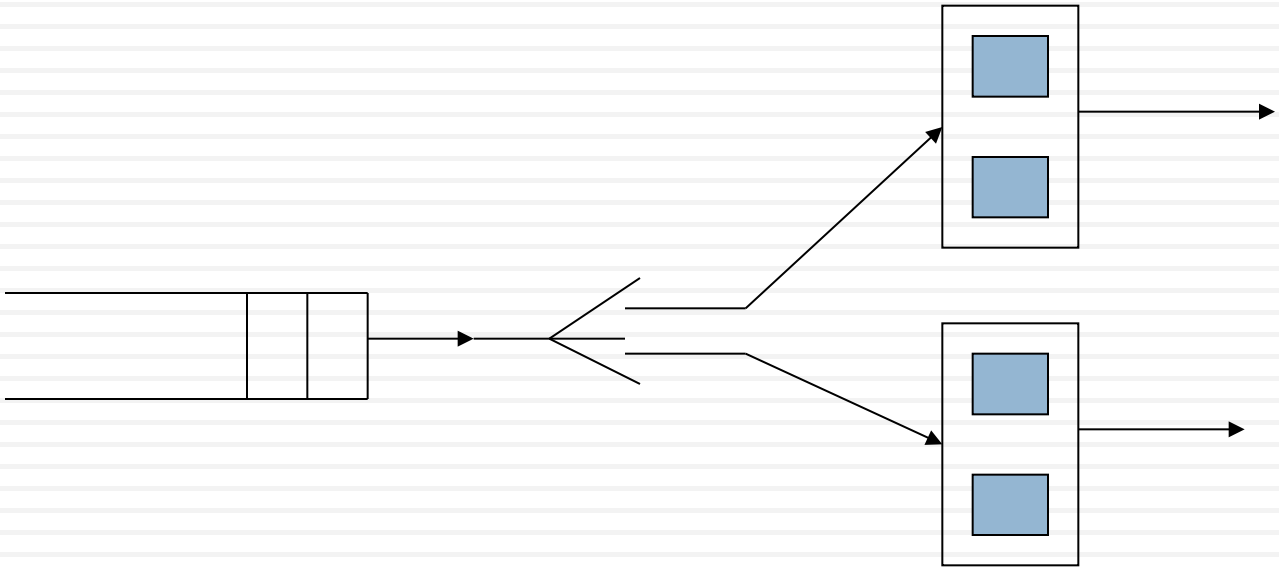
- Simple
- Allows to describe the system faster.

Bad

- No description about the codification.
- No description about the events.
- Is not calculable.
- Not structured methodology, not specific of the OR.

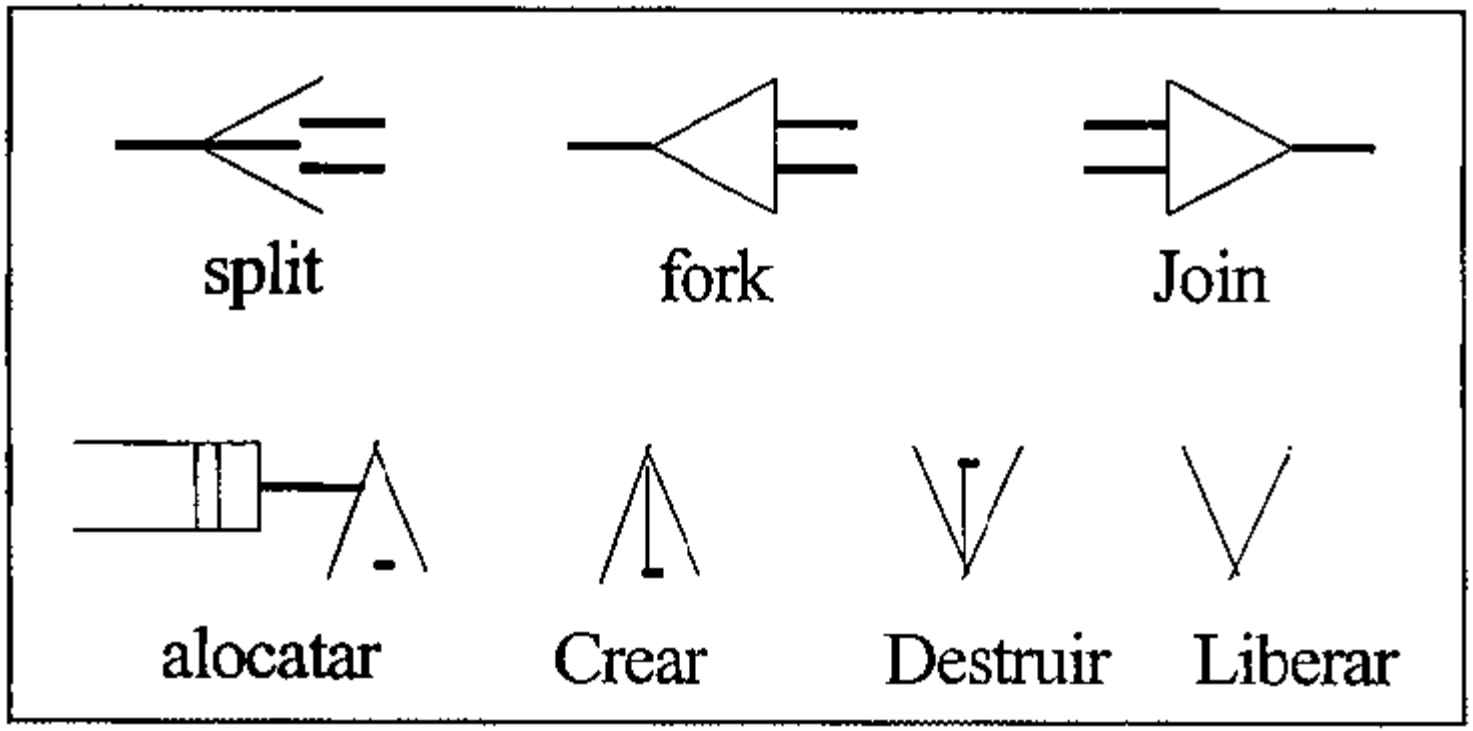
Queue networks

Simulation models formalization

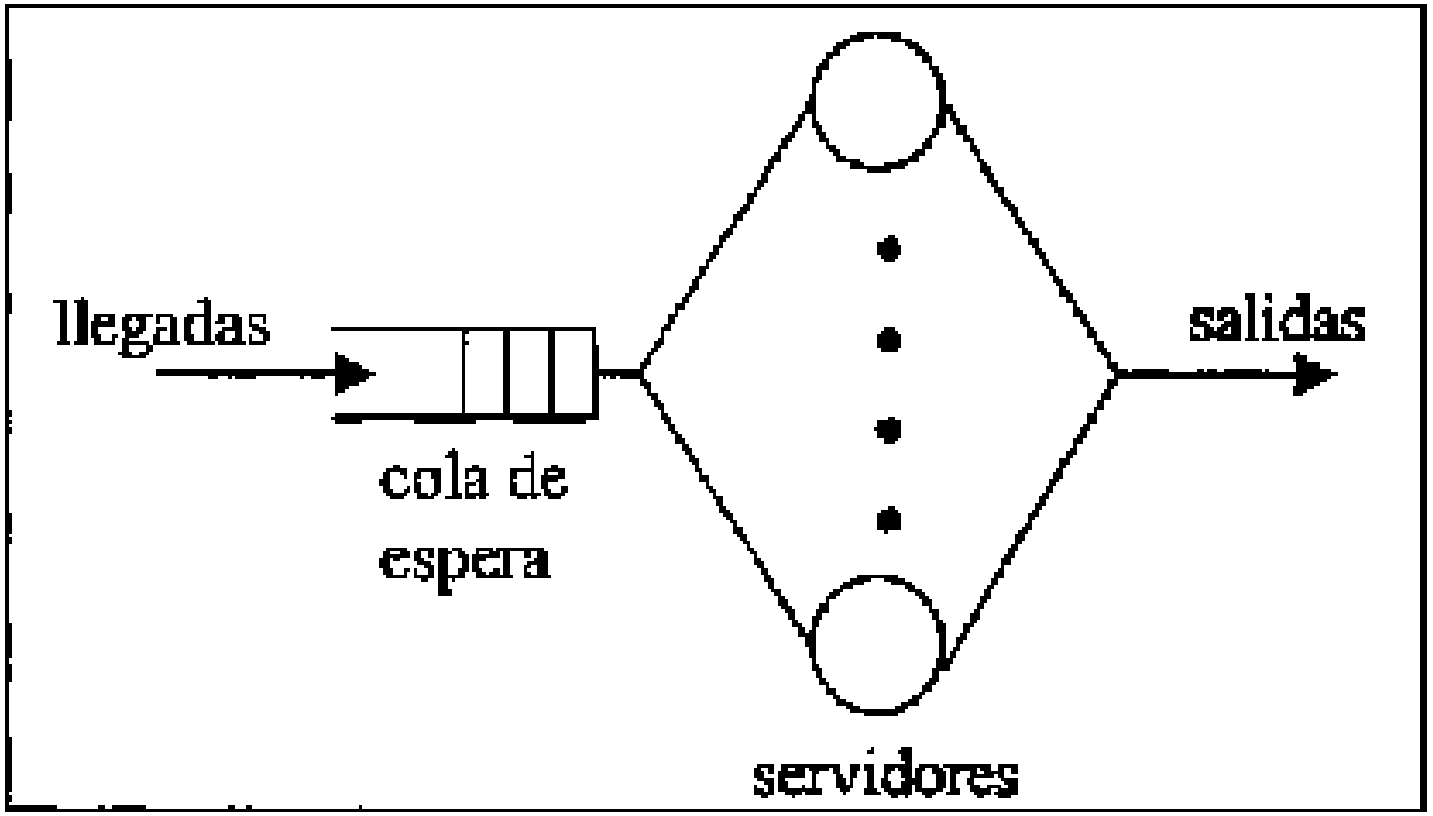


Basic structure of queue models

Queue networks

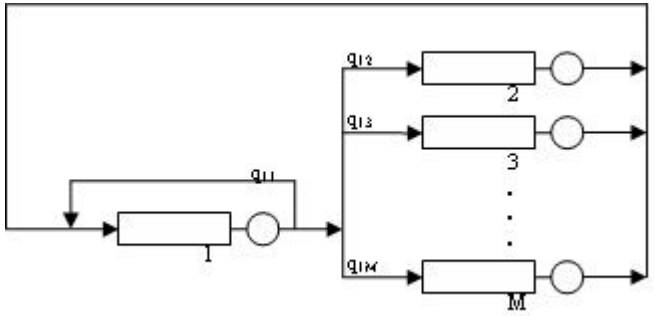


Queue networks (M|M|S)



Queue networks (best)

- Simple
- Allows to understand the system faster.
- Specific to describe queue models.



Queue networks (worse)

- No description about the implementation.
- Do not describe too much about the events management.
- Is not always calculable.
  - Some models can be calculated following the queue theory.

Service blueprint

From the customer perspective

Service Blueprint

- The **service blueprint** is a technique originally used for service design and innovation but has also found applications in diagnosing problems with operational efficiency.
- The technique was first described by G. Lynn Shostack, a bank executive, in the *Harvard Business Review* in 1984.[1]
- The service blueprint is an applied process chart which shows the service delivery process from the customer's perspective. The service blueprint has become one of the most widely used tools to manage service operations, service design and service positioning.

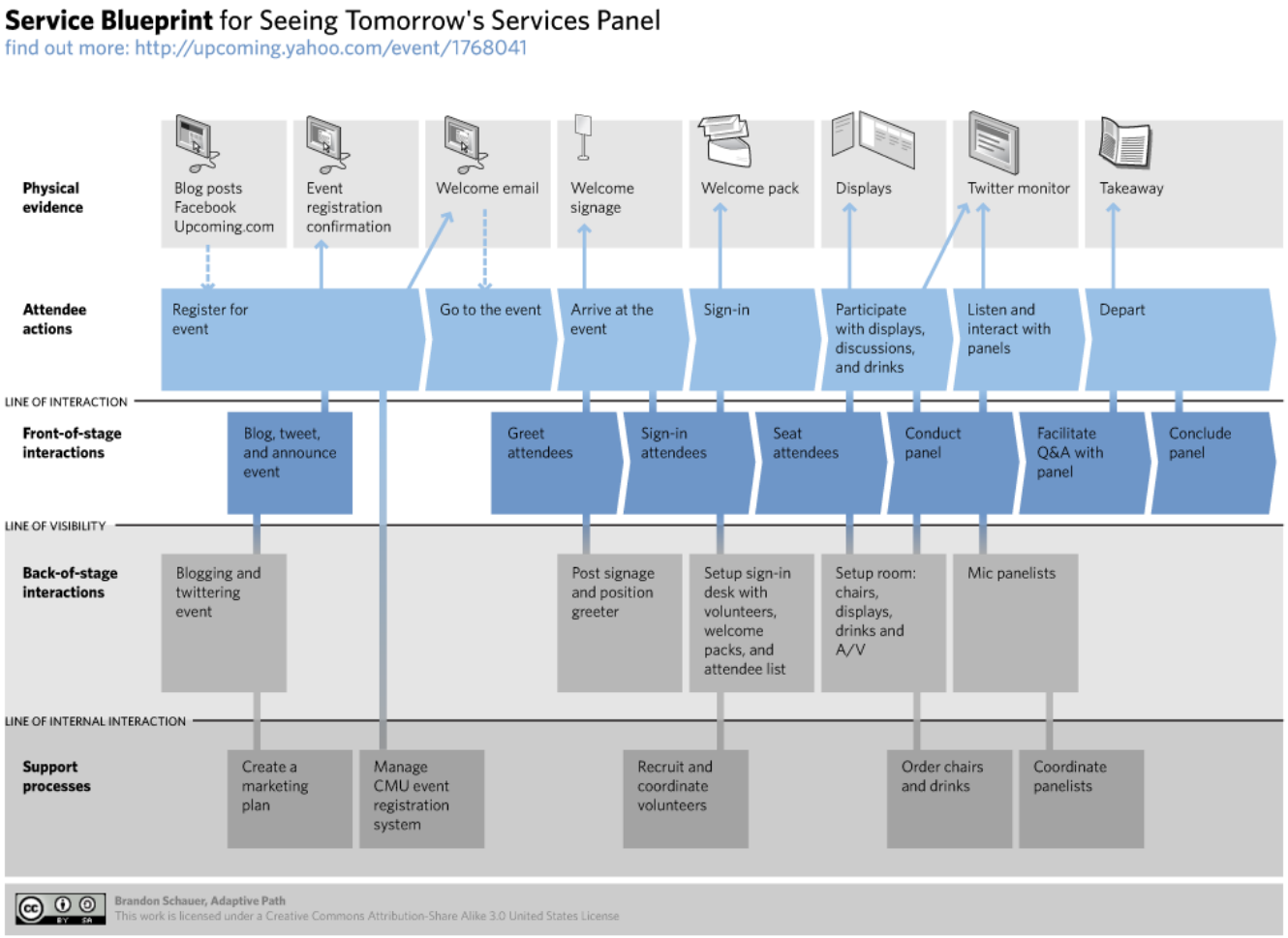
Farm example

Key element of CS Blue Print

Physical evidence : All tangible that the pig experience during within in box.
Pig action : Any step, choices, interaction or activities of the pig experience while entering o exit from the box.
Front stage : current interaction or steps that are visible to the pig.
Back stage : current interaction or steps that are invisible to the pig.
Support Process : system that support the action of the pig.

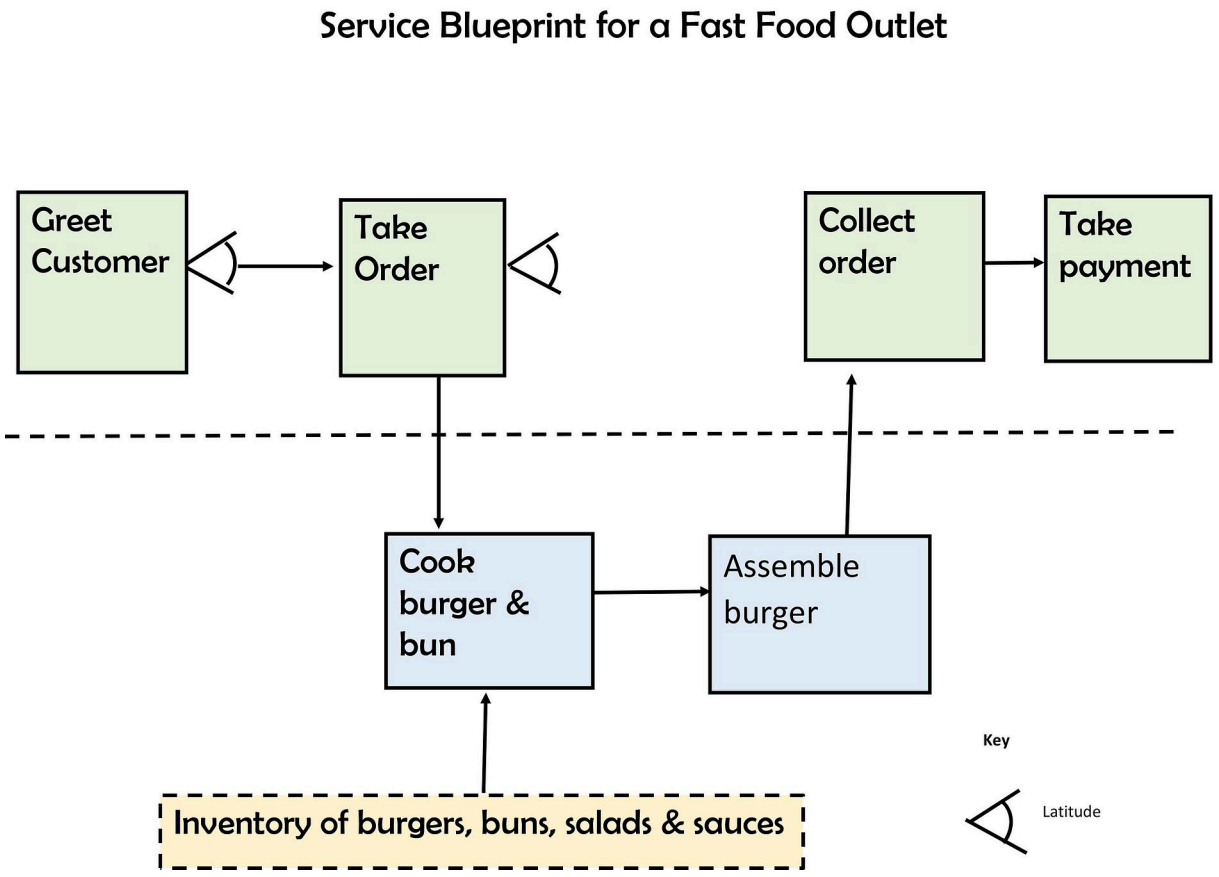
Physical Evidence	Time to eat	Farm & Food box	Helper Person	Pig side to Food side		
Pigs Action	Call for food (Hungry)	To enter in the box	Interaction with other pig	Start to eat	Process start Food side	Untill pig full
Line of interaction						
Front Stage			Took them to the food	Eating food		
Line of visibilty						
Back Stage	Prepare food			Recording system		
Line of internal interaction						
Support Process	Getting to the food box		Start processing	Electronic chip	Register the entire system	

Service Blueprint

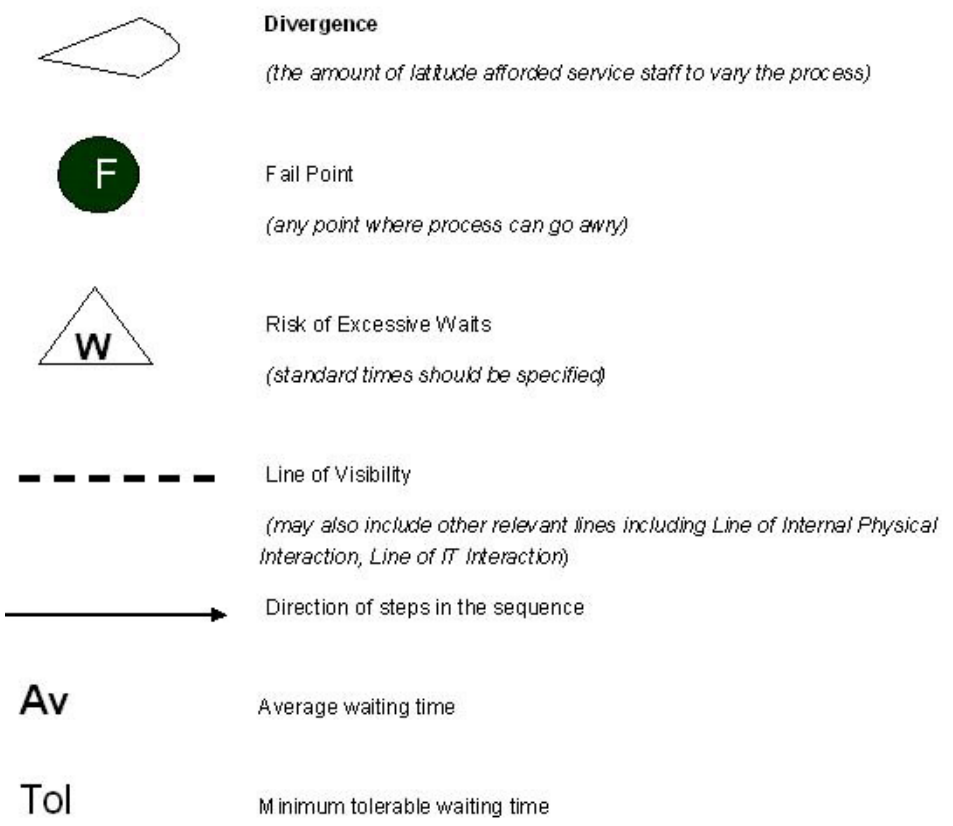


By brandon schauer - https://www.flickr.com/photos/brandonschauer/3363169836, CC BY-SA 2.0, https://commons.wikimedia.org/w/index.php?curid=41292602

Service Blueprint



Blueprint common simbols



BPM

Business Process Modeling

BPML

Business Process Modeling Language

BPML

- BPML was designed as a formally **complete language**, able to model any process, and, via a BPMS (business process management system), deployed as an executable software process without generation of any software code.
- This is not possible with BPEL, since BPEL is not a complete process language. In practice BPEL is often used in conjunction with Java to fill in the "missing" semantics. In addition, BPEL is often tied to proprietary implementations of workflow or integration broker engines. Whereas, BPML was designed, and implemented, as a pure concurrent and distributed processing engine. It was designed to be semantically complete according to the Pi-calculus formal representation of computational processes.
- BPEL and BPML are examples of a trend towards process-oriented programming. BPEL and BPML herald the concept of a BPMS as an IT capability for management of business processes, playing a role similar to a RDBMS for business data.

BPEL

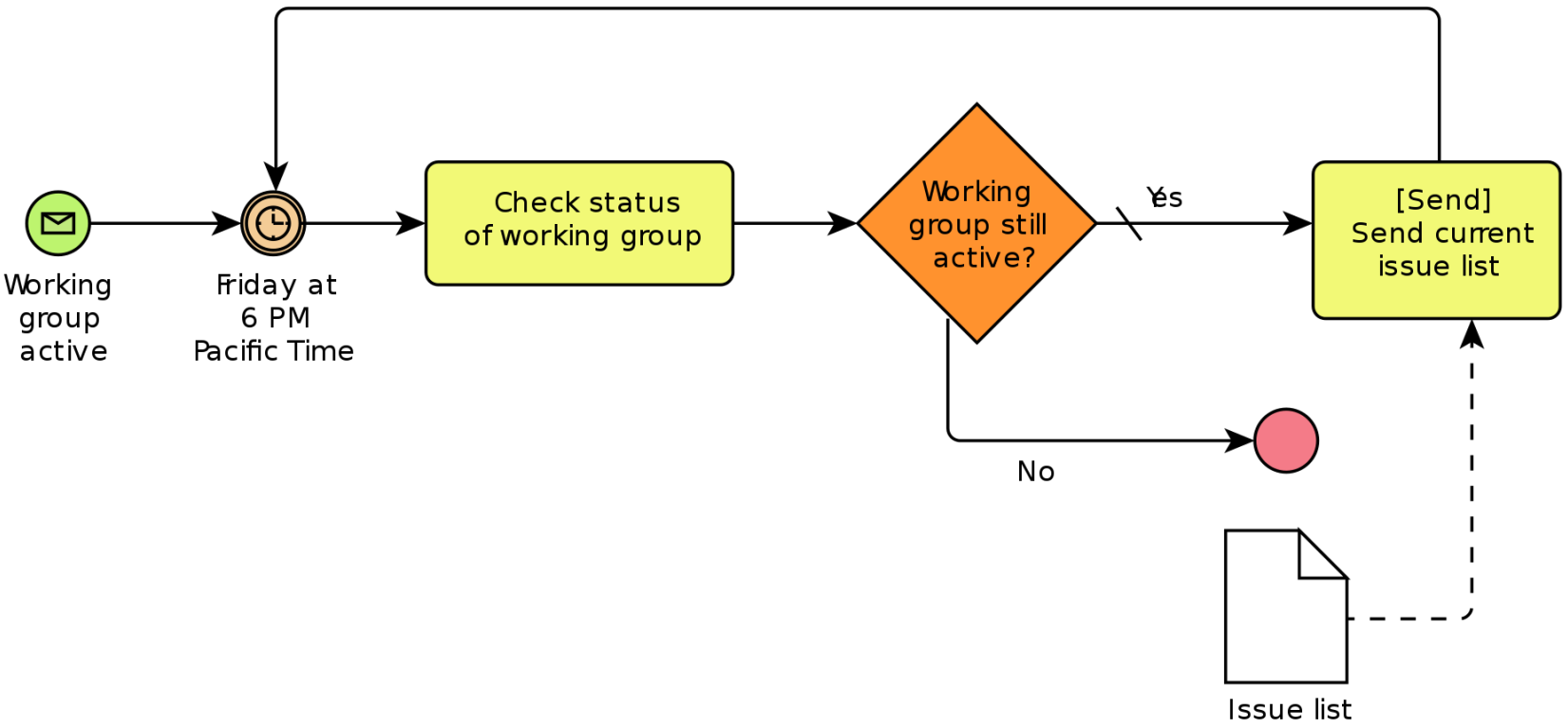
Business Process Execution Language

- The **Web Services Business Process Execution Language** (**WS-BPEL**), commonly known as **BPEL** (**Business Process Execution Language**), is an OASIS[1] standard executable language for specifying actions within business processes with web services. Processes in BPEL export and import information by using web service interfaces exclusively.

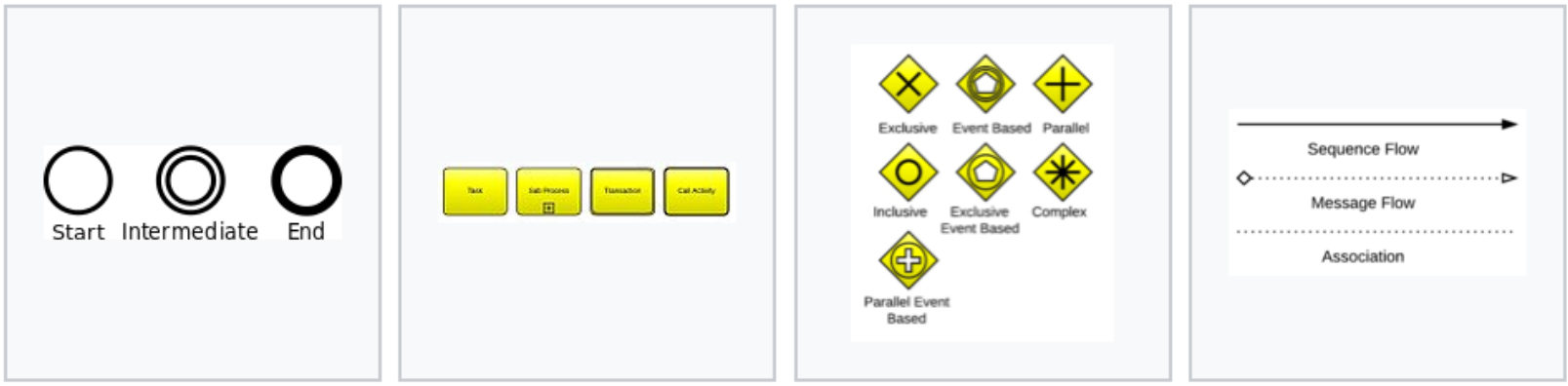
BPMN

Business Process Model and Notation

BPMN

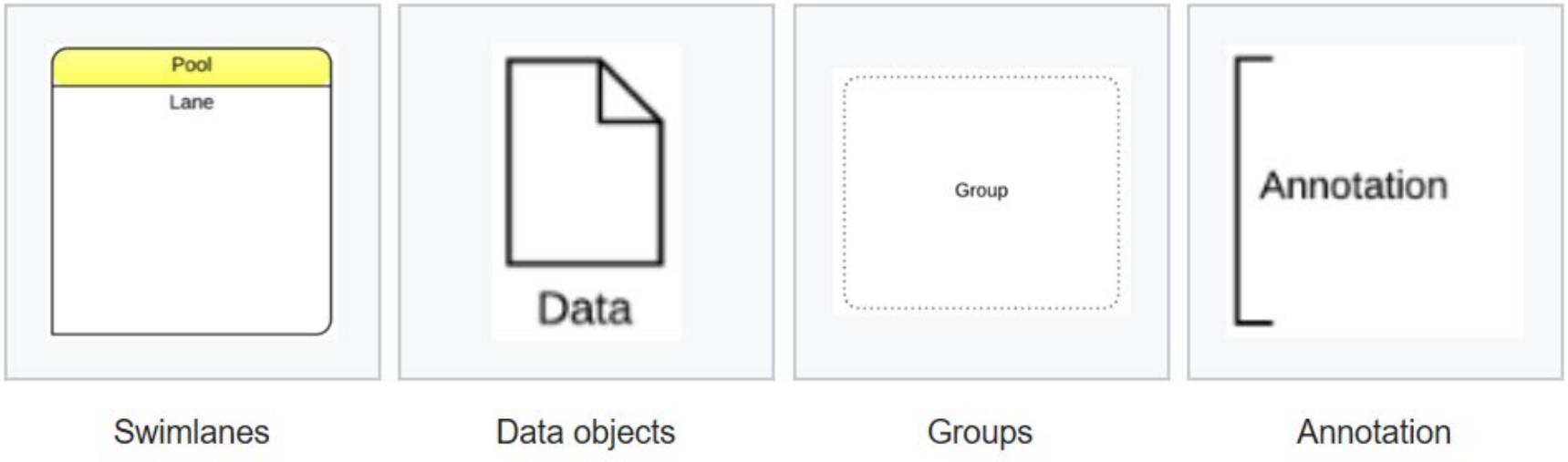


BPMN

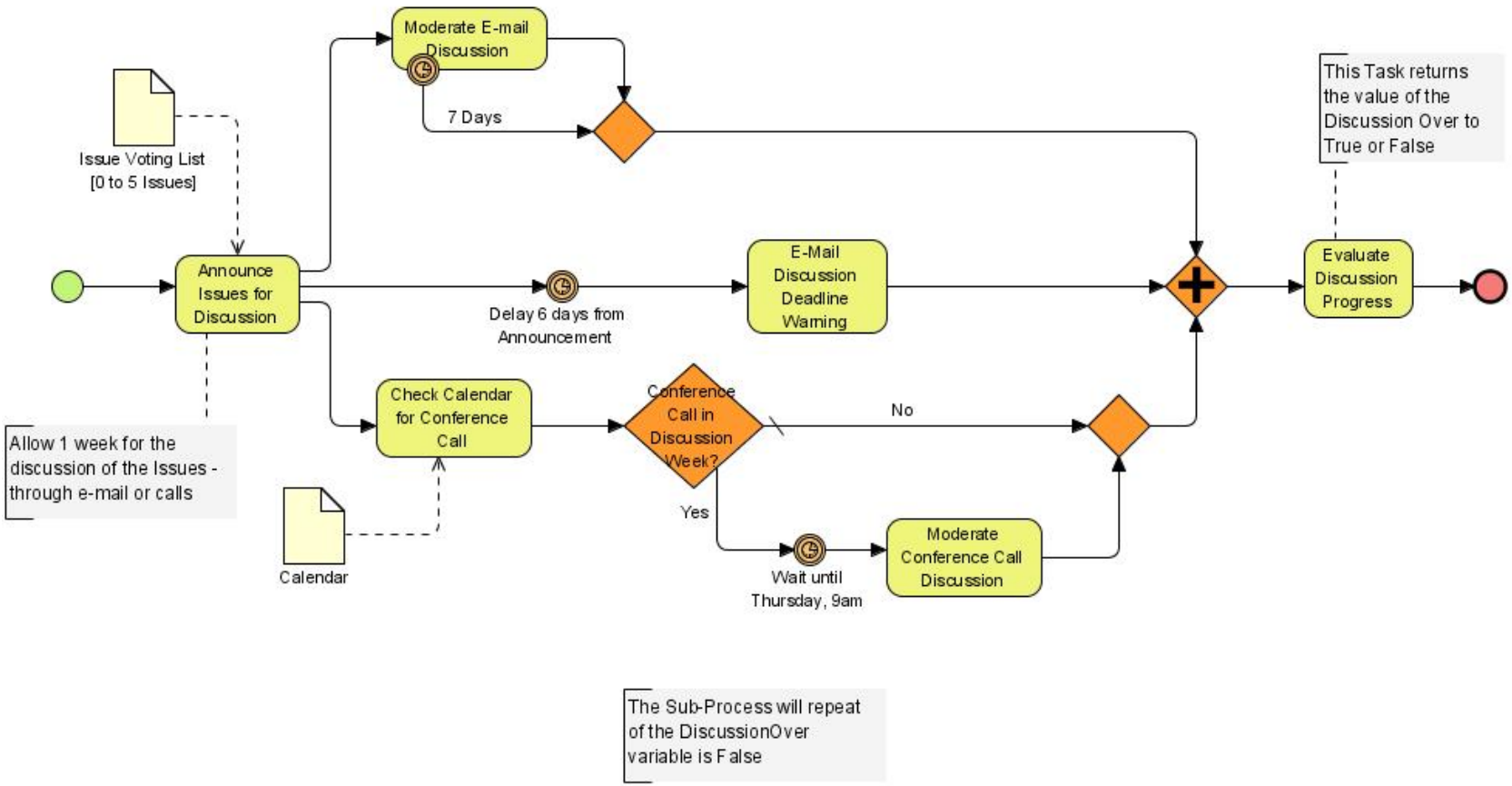


Event Activity Gateway Connections

Swim lanes and artifacts



Discussion cycle BPMN example



By Tttt1234 - Own work, Public Domain, https://commons.wikimedia.org/w/index.php?curid=4845066