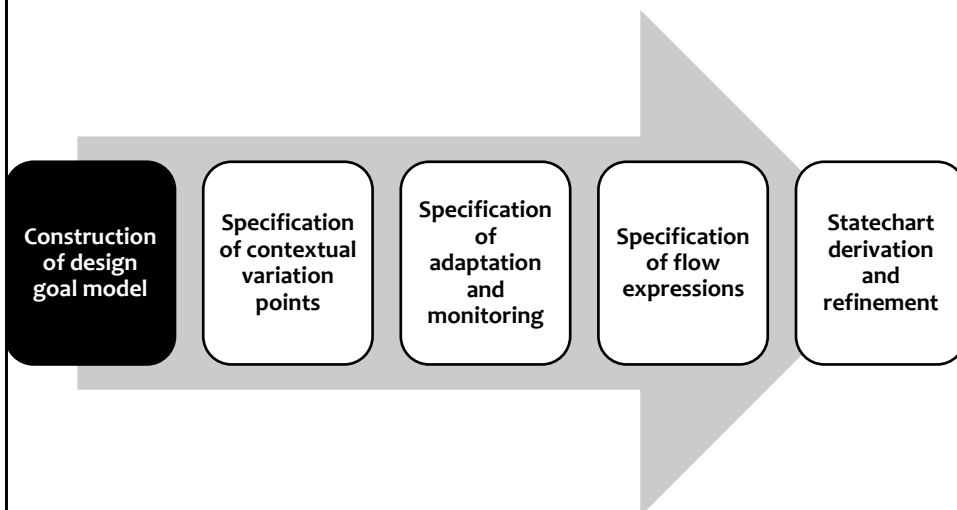
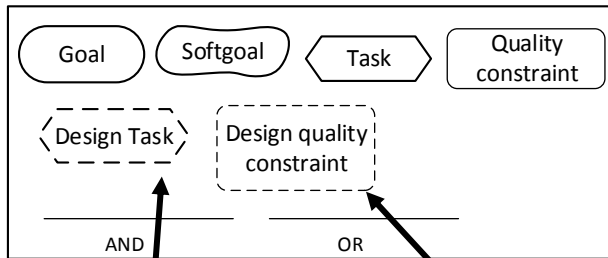


Reference Guide

Overall Process



Goal Model[Notation]



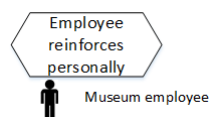
The **tasks** and the **quality constraints** identified in the **design** phase are represented with the dashed notation.

1° Activity of the process

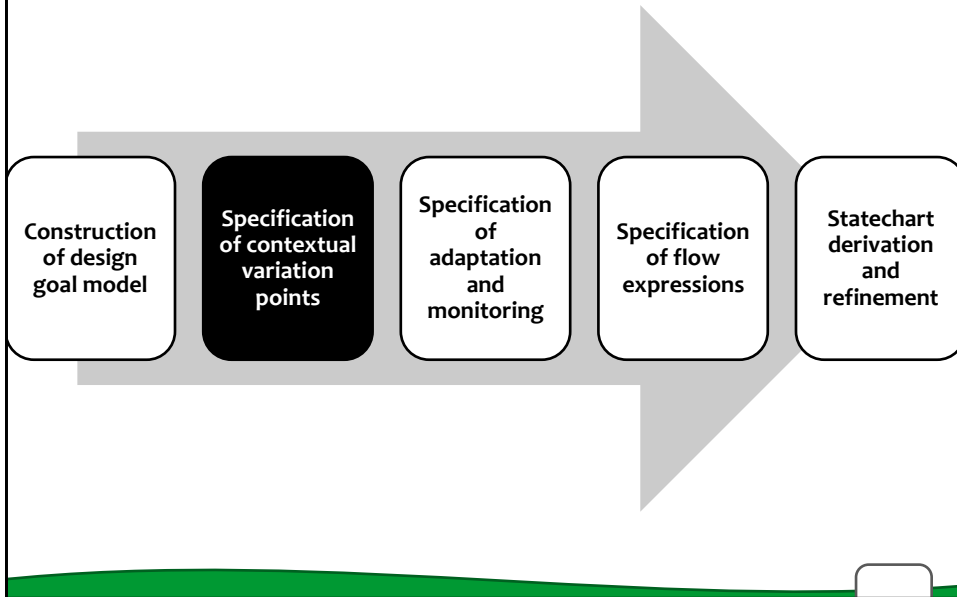
■ Steps:

- Identify the requirements appeared in the design phase such as
 - Non-functional requirements
 - *Design tasks*
 - *Design quality constraints*
- Assign a design task (that will not be executed by the system) to an user (if there is any task of this type)

Example:

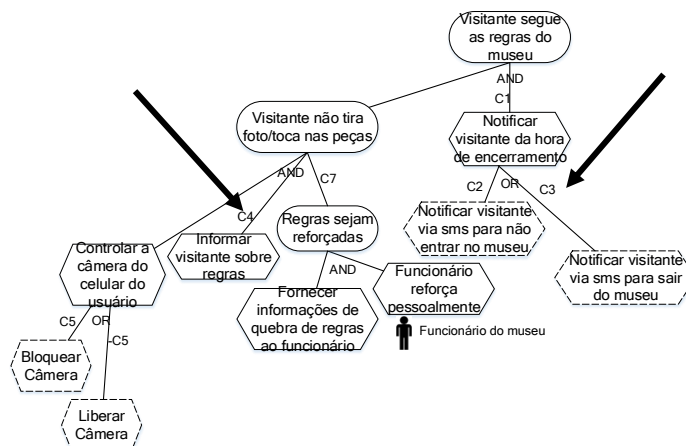


Overall Process



Contextual Goal Model [Notation]

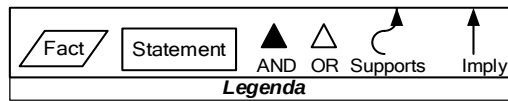
The contextual variation points **can appear in AND/OR** refinements and they are specified through a **label (C1, C2.. Cn)** in the goal model.



2º Passo do processo

■ Steps:

- Identify and specify the contextual variation points.
 - tip: look for situations that represent conditions for a task to be executed.
- Refine each context using the following notation



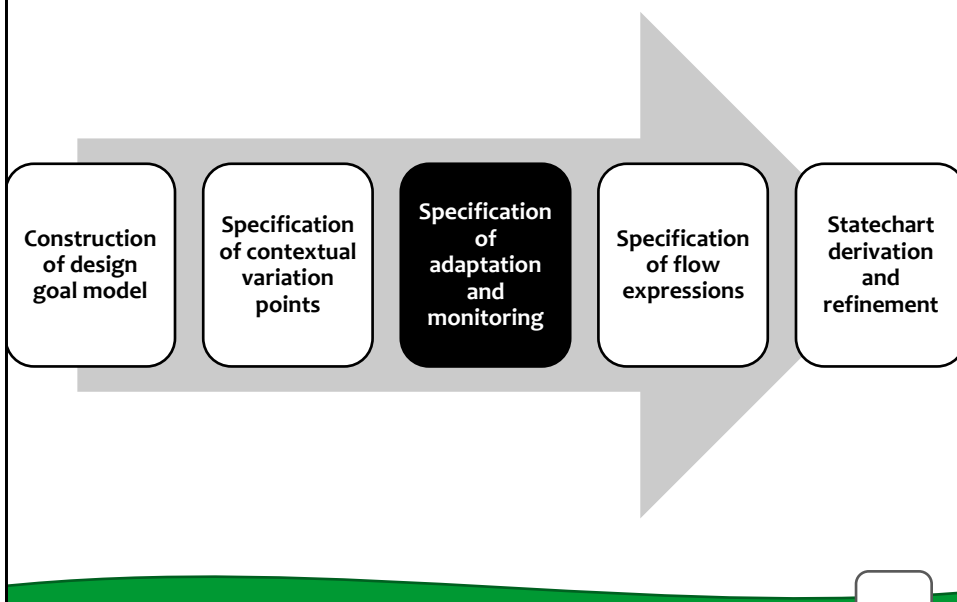
Supports: used to connect *facts* and *statements*.

Imply: used to connects *facts* or *statements* to the context.

Fact: it is possible to check if its value is true ou false directly.

Statement: its value (V or F) is infered through the *facts*.

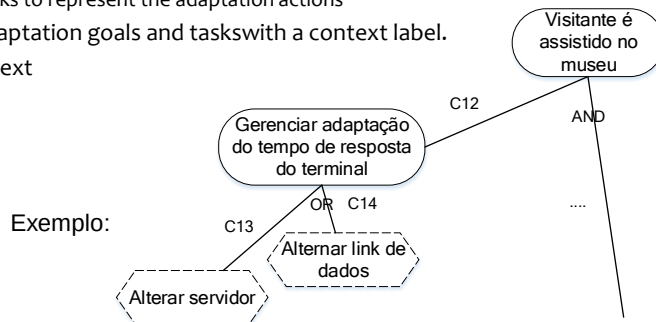
Overall Process



3º Passo do processo

■ Steps:

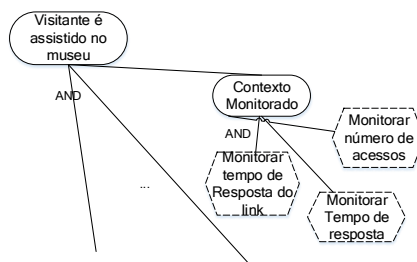
- Define the requirements that requires na action in case of failure (critical requirements for the system)
- Represente the adaptation management
 - Adid a new goal in the root node to the management
 - Add subgoals for each critical goal that must be monitored and adapted (in case of more than one critical requirement)
 - Add design tasks to represent the adaptation actions
- Associate the adaptation goals and taskswith a context label.
- Refine each context



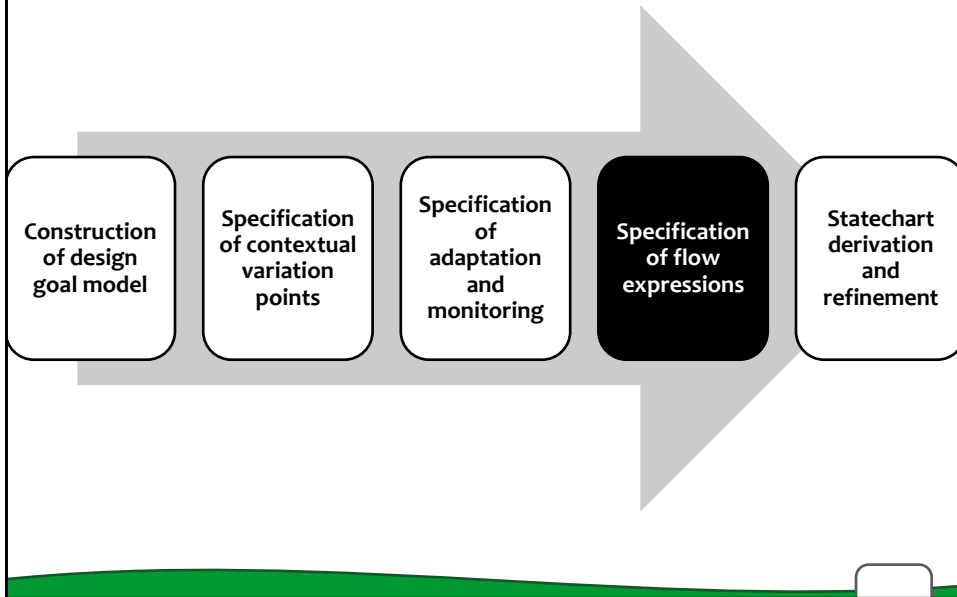
3º Step of the process

■ Steps:

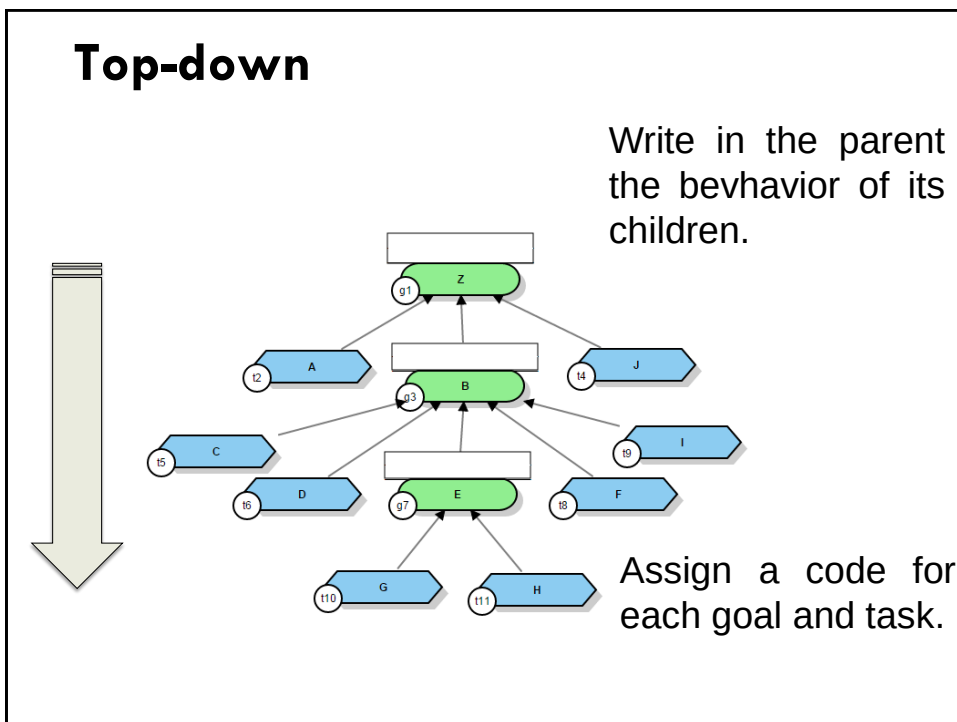
- Identify the dynamic contextual elements (properties of real-world present in the refinements of the contexts facts that need to be monitored all the time from the system because its value constantly changes at runtime)
- Represent the context monitoring
 - Add a new goal in the root node
 - Adddesign tasks to monitor each dynamic contextual element



Overall Process



Top-down

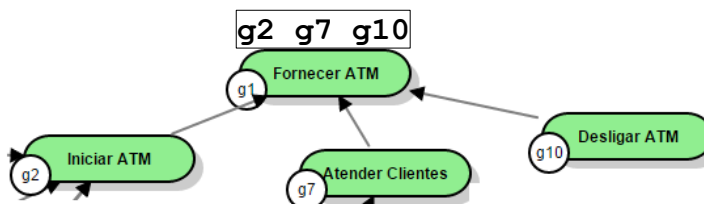


Flow Expressions [Notation]

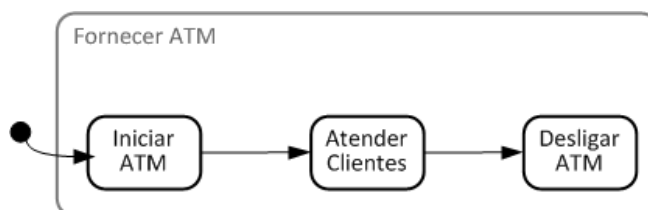
- $A B \rightarrow$ sequence (first A, then B)
- $A | B \rightarrow$ alternative (A or B)
- $A? \rightarrow$ A is optional
- $A+ \rightarrow$ repetition (1 or more times)
- $A^* \rightarrow$ repetition (zero or more times)
- $A - B \rightarrow$ A and B in parallel (ortogonal)
- $i1 \rightarrow$ idle state

1) Sequence (result)

- Goal model:

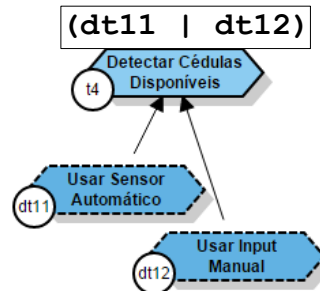


- Statechart:

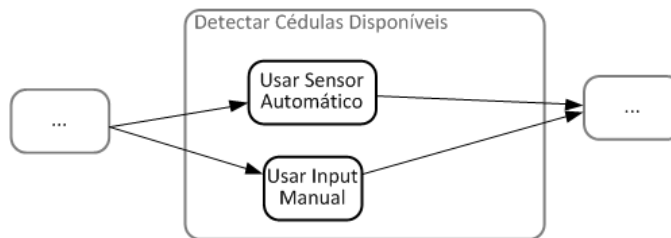


2) Alternatives (result)

■ Goal model:

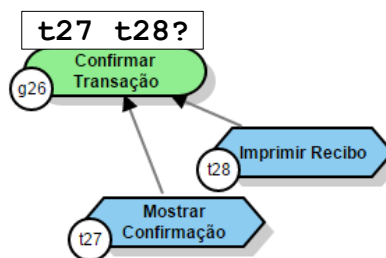


■ Statechart:

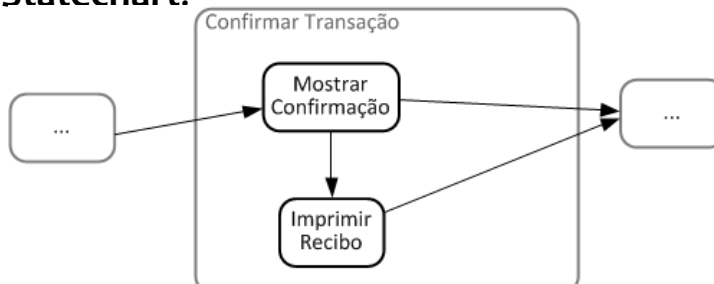


3) Optional (result)

■ Goal model:

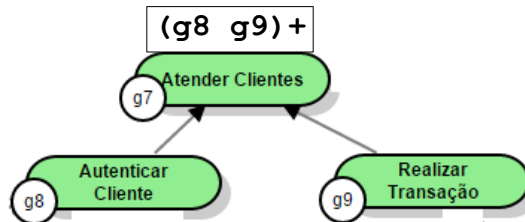


■ Statechart:

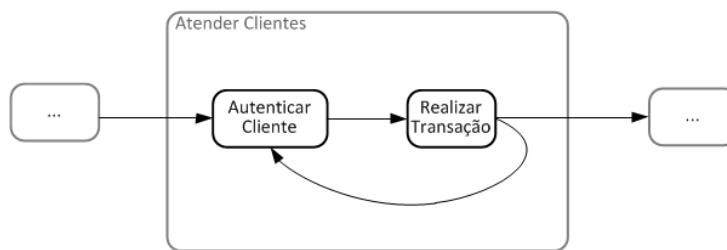


4) Repetition 1 or more (result)

■ Goal model:

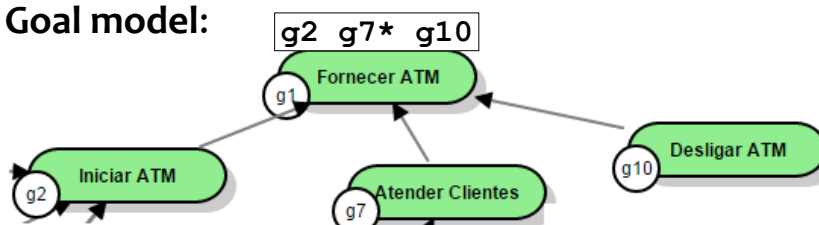


■ Statechart:

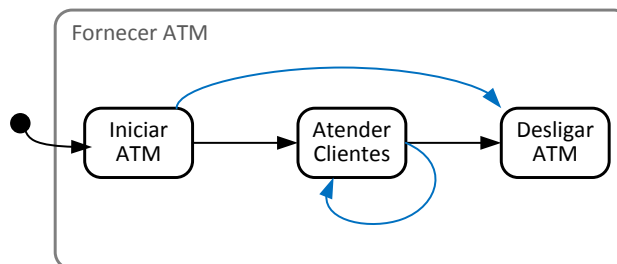


5) Repetition 0 or more (result)

■ Goal model:

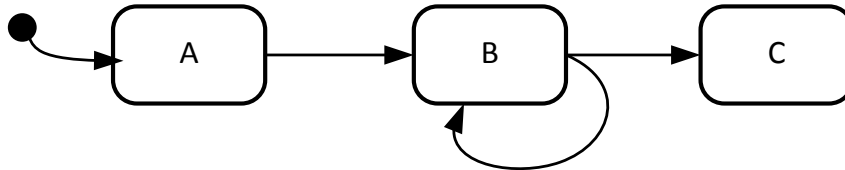


■ Statechart:

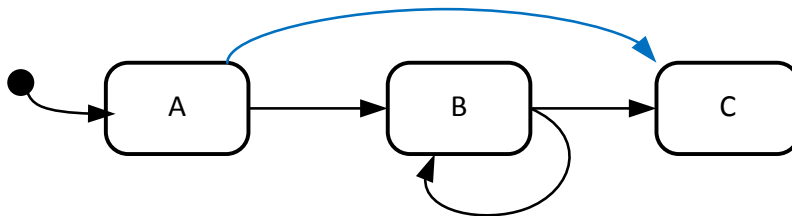


Comparison between 1+ e 0+

- $A B^+ C$ (1 or more times)

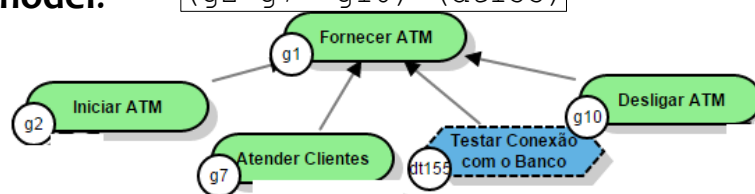


- $A B^* C$ (zero or more times)

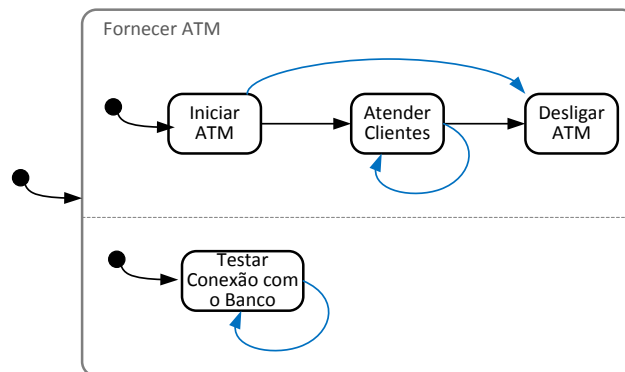


6) Paralell (result)

- Goal model: $(g2 \ g7^* \ g10) - (dt155)$



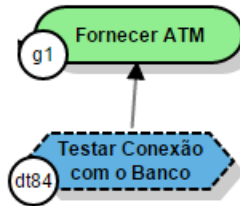
- Statechart:



7) Idle states

- **Case:** when a state in which nothing will be done is needed - the system is just waiting for the occurrence of any event or context to be true.

- **Example:**



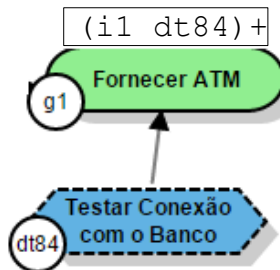
- **Expression:**

- Use iX (where X is a number) to add a idle state:

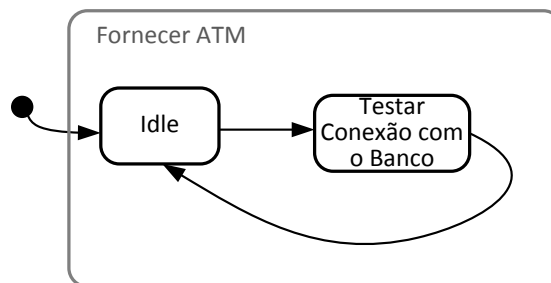
$(i1 \ dt84) +$

7) Idle states (result)

- **Goal model:**



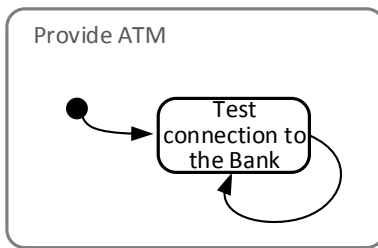
- **Statechart:**



Comparison...

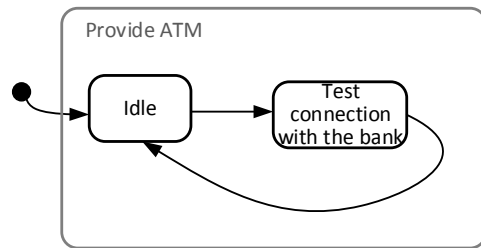
■ Without *idle*

dt84+

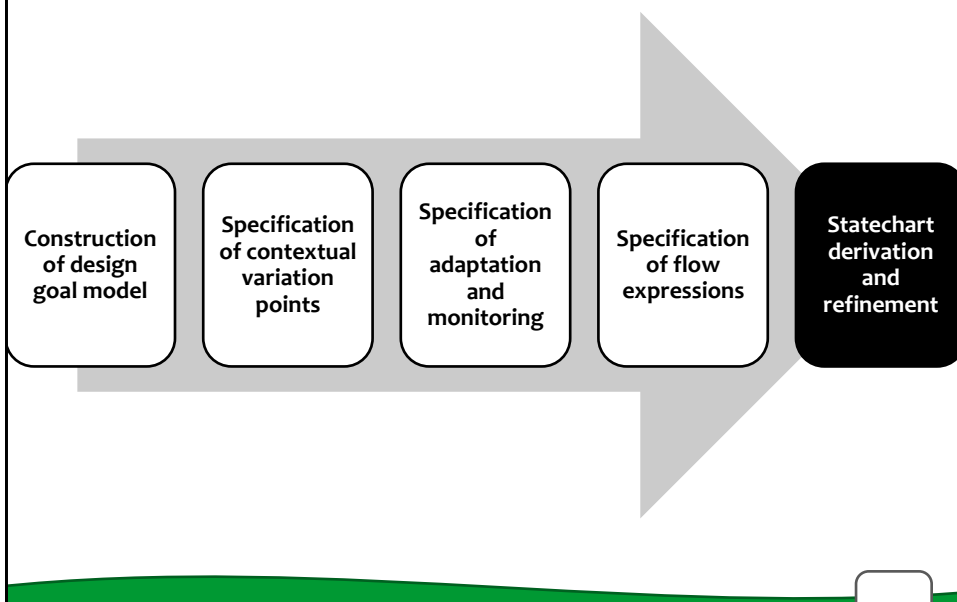


■ With *idle*

(i1 dt84)+



Overall Process

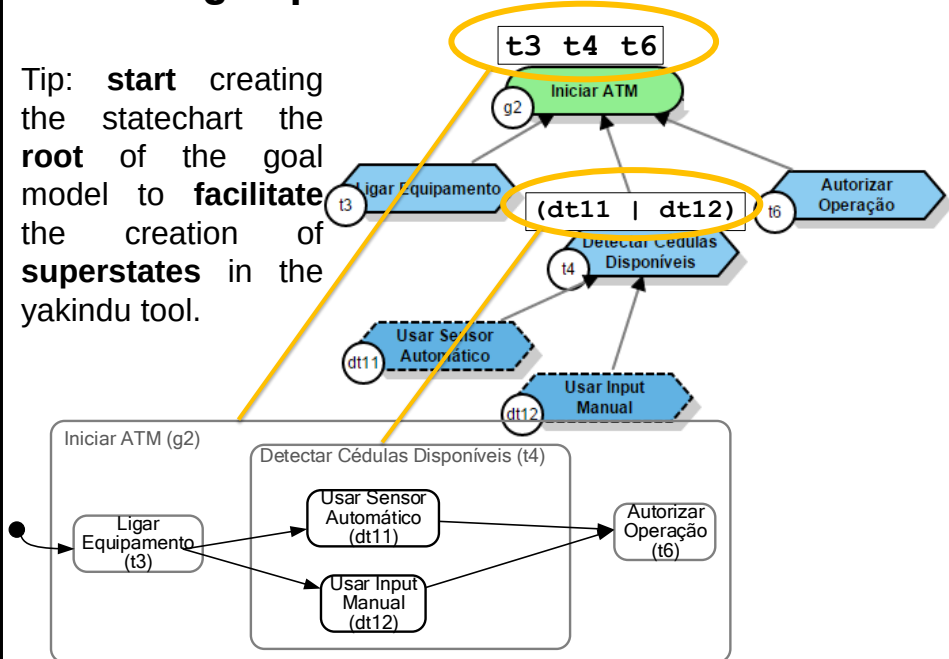


5º Step of the process: Derivation strategy

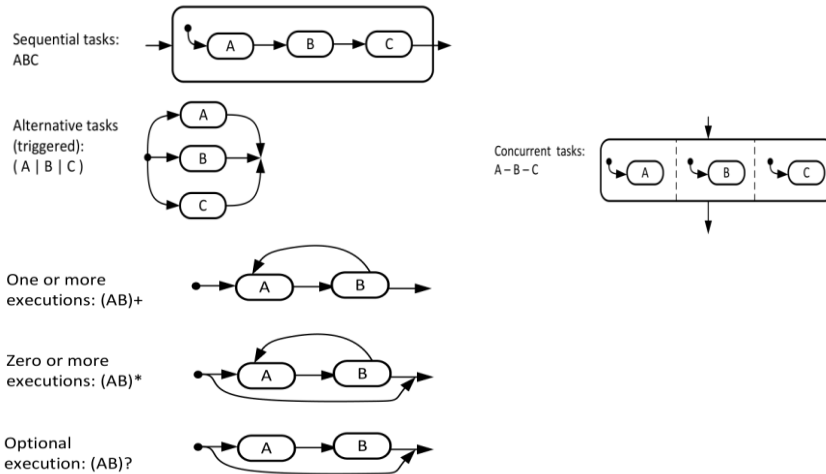
- Create statechart from the goal model:
 - Model the **goals/tasks** in **refinements** as super-states
 - The children elements of these elements should be sub-states of the parent.
 - Model the tarefas states
 - Create **idle states** to wait until the context be true to occur state change.

Combining expressions

Tip: **start** creating the statechart the **root** of the goal model to **facilitate** the creation of **superstates** in the yakindu tool.



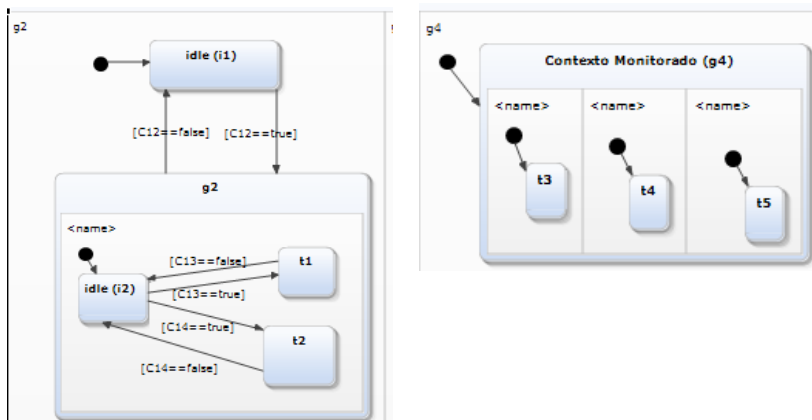
5° step of the process: statechart derivation patterns



5° step of the process: events and conditions

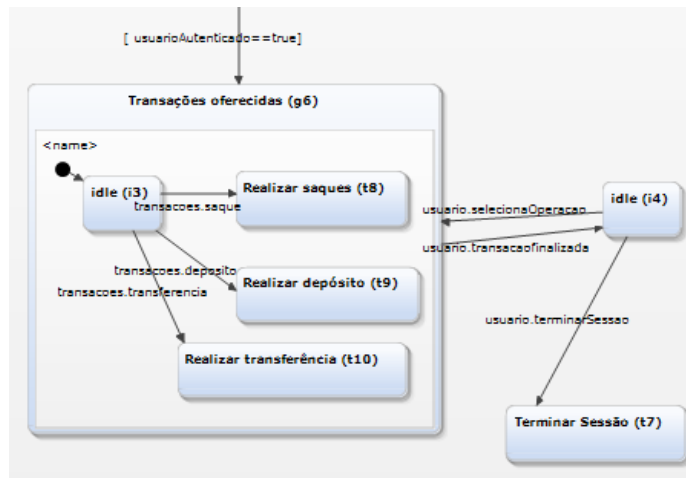
- After generating the statechart, it is necessary to write the events and conditions of the transitions.
- The context monitoring and the management of adaptation actions should be represented in parallel.

Example:



5º step of the process: events and conditions

Exemplo:



Sources of inspiration for defining events

- Context is true/false
- Task accomplished
- Task request by a user
- Timer