Toward a formal theory of systems

CAL07

Claude FELIOT

03/10/2007

TRANSPORT

ALSTOM

INTRODUCTION

Toward a formal theory of systems

Why?:

- No shared understanding of the concept of system
- Need for checking soundness and clarifying concepts
- Need for a clear understanding of system specifications entities and related “proof obligations” for sound system design

Systems as Phenomena or Vision?

System phenomena?

- *A matter of size?*
- *A matter of complexity?*
- *An intrinsic property?*
- *…*

On being a system

- *To be considered as a system*
- *To be seen as a system*
- *To be represented as a System*

“System” is the denotation of a vision, a way of thinking!

We thinking relies on models

A System theory is basically a a theory of modeling

BUILDING A SYSTEM VISION

TRANSPORT

ALSTOM

About Systems

A system always exists within an environment that set up the **operational contexts** to which it will have to adapt.

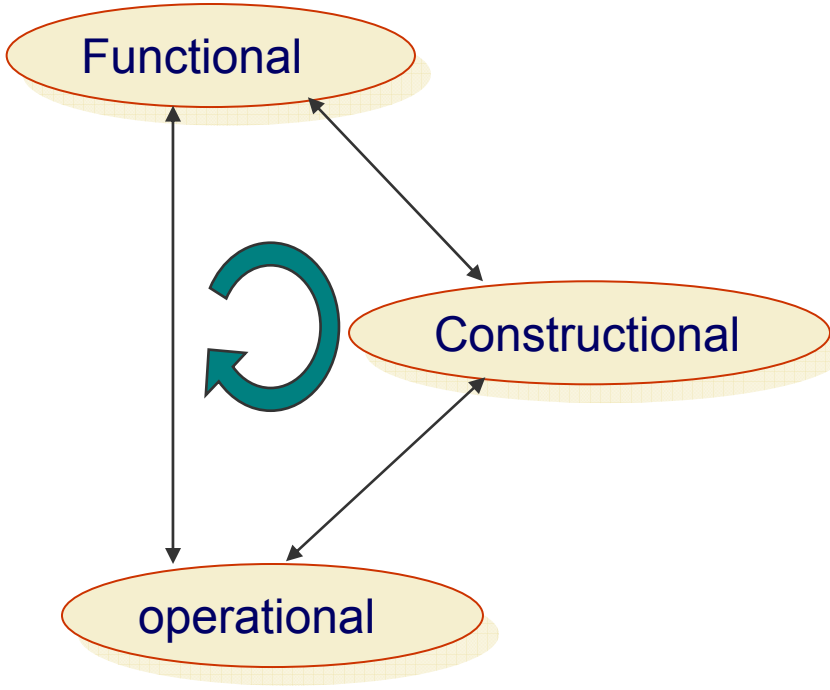
In these operational contexts, some interacting external systems have **needs** that it will have to contribute to either by:

- Doing, or
- Being

something

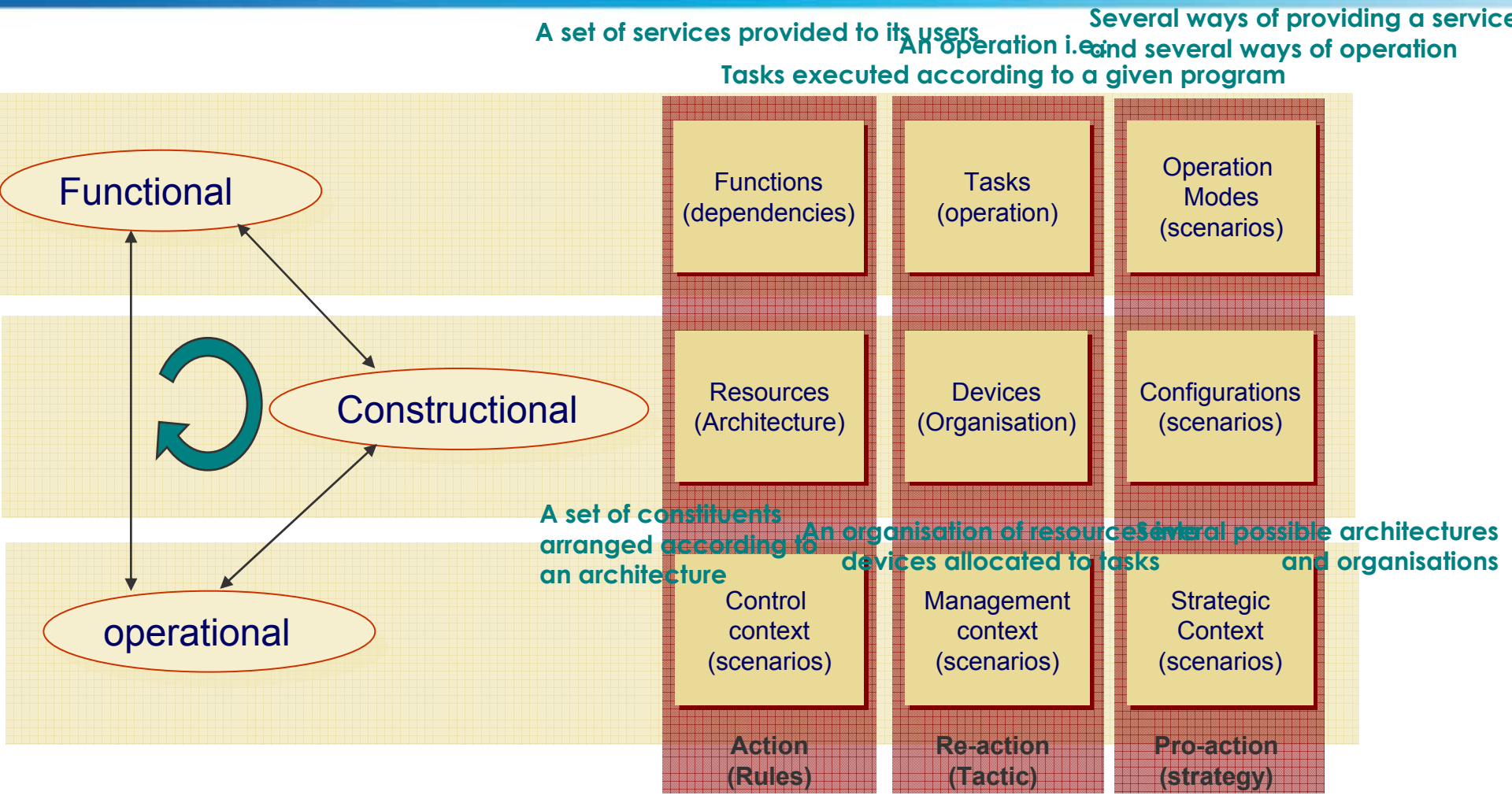
There are thus two ways for System adaptation

- **Functional**
- **Constructional**



What the System Does Shall be Consistent with what it is.

A system vision



TRANSPORT ALSTOM

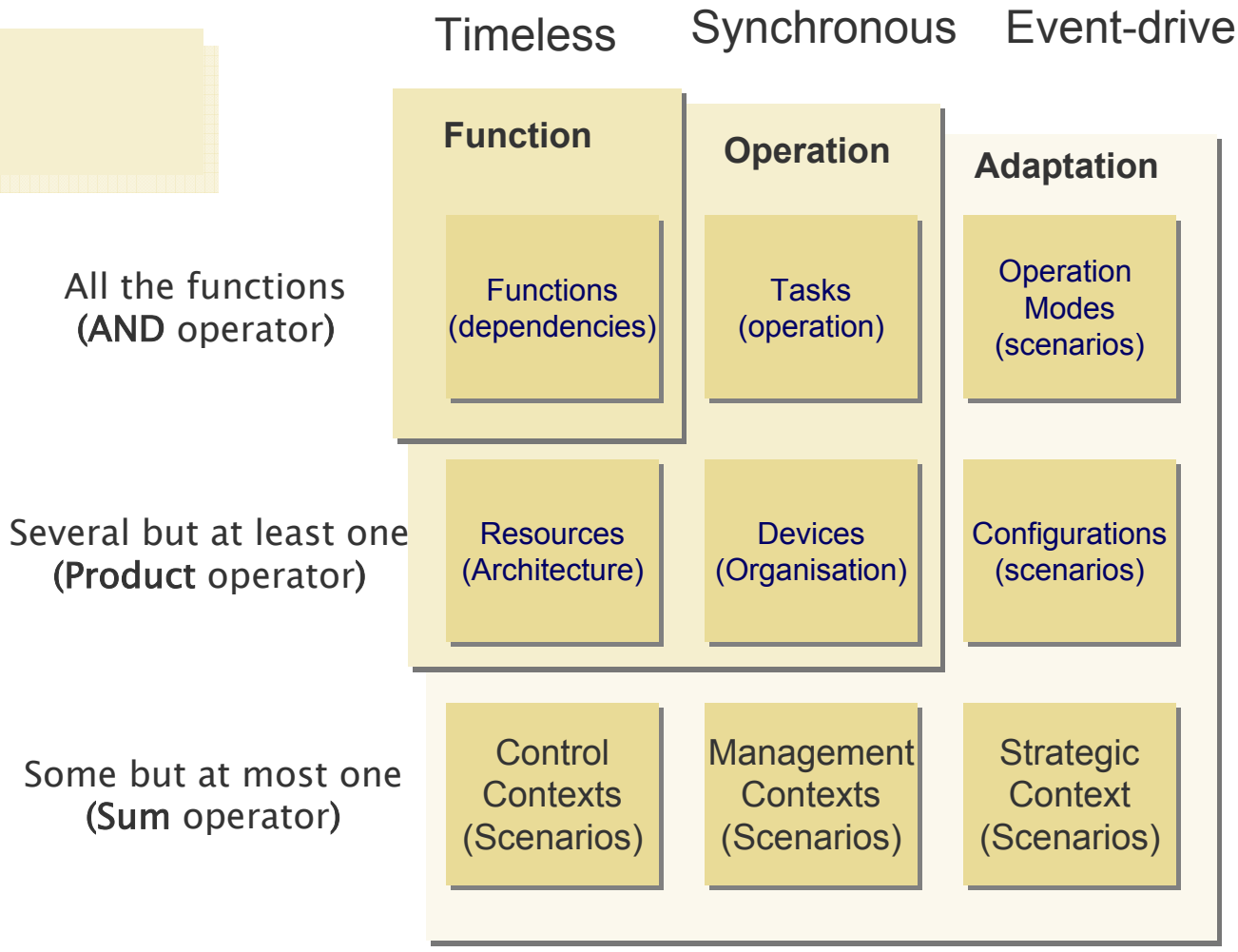
TOWARD A FORMAL SEMANTIC

TRANSPORT

ALSTOM

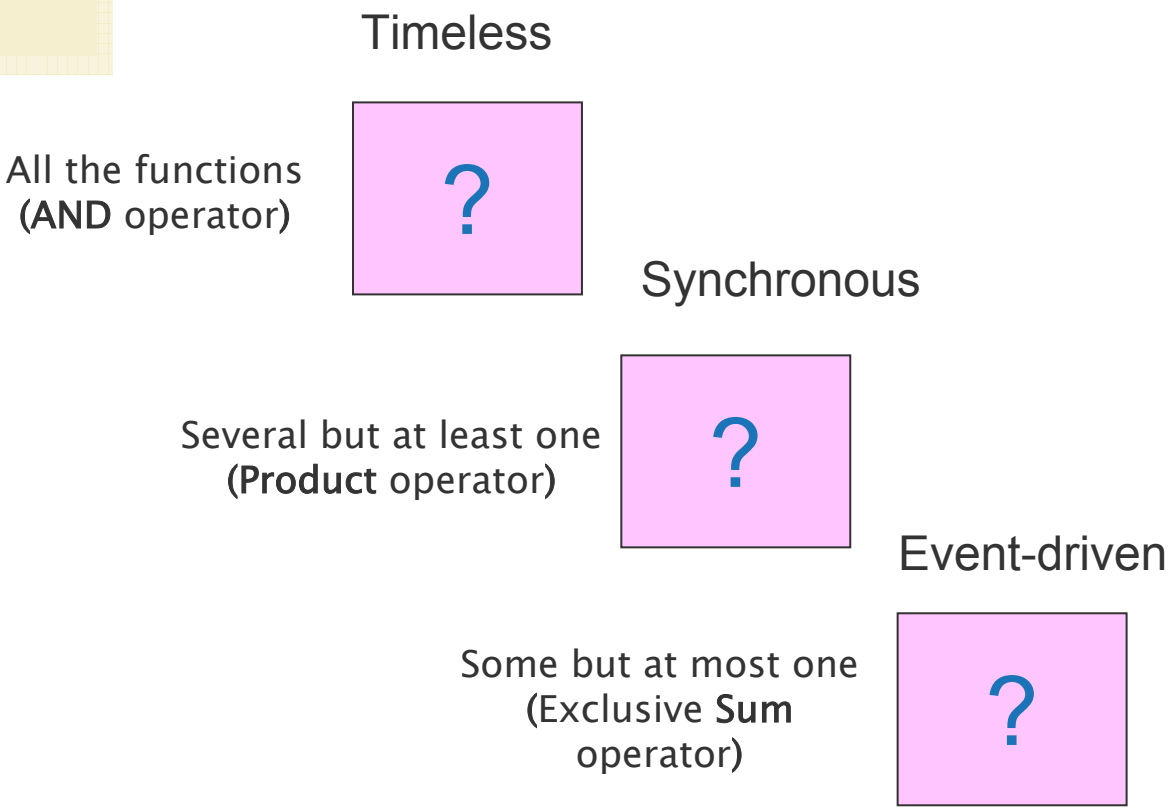
Time perspectives and Logical operators

There are basically three:
- Time semantics
- Type of operators



Looking for canonical forms

Looking for **three canonical forms** of statement (e.g. specifications) in the system vision.

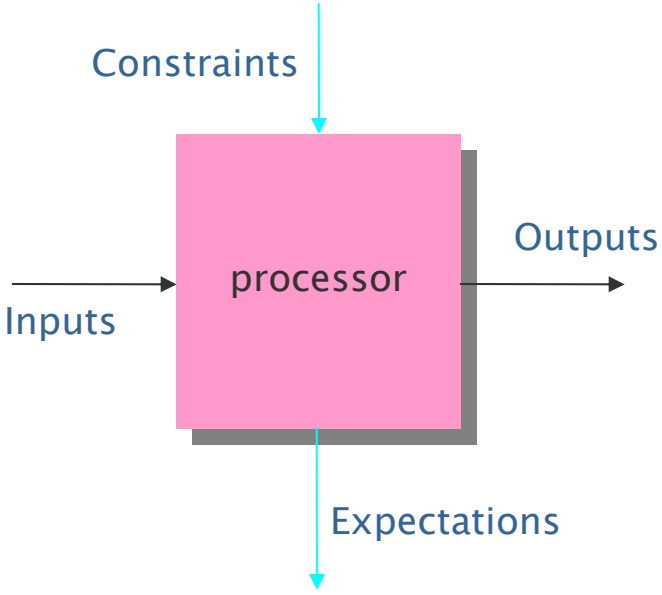


Canonical forms semantic

TRANSPORT

ALSTOM

A graphical language



•Under the **constraints** the process provides its **outputs** from its **inputs** such that the **expectations** been satisfied.

Predicate transformers

- Conditions

$$\{p\} : pred \rightarrow pred$$
$$:= \{p\}.q = p \wedge q$$

$$\{False\} = Abort$$
$$\{True\} = Skip$$

$$S_1;\{x \geq 10\};S_2 \Rightarrow \begin{cases} S_1;\{False\};S_2 \Leftrightarrow S_1;Abort \\ S_1;\{True\};S_2 \Leftrightarrow S_1;S_2 \end{cases}$$

- Event

$$[p] : pred \rightarrow pred$$
$$:= [p].q = p \Rightarrow q$$

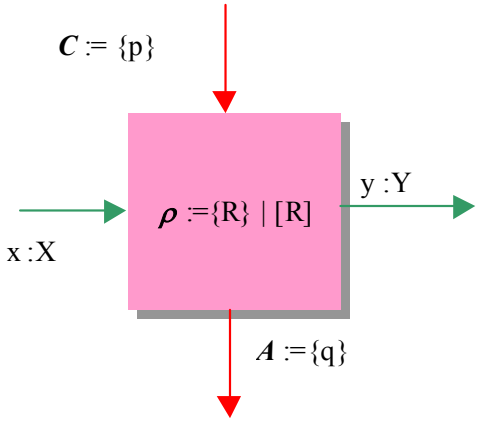
$$[False] = Magic$$
$$[True] = Skip$$

$$S_1;[x \geq 10];S_2 \Rightarrow \begin{cases} S_1;[False];S_2 \Leftrightarrow S_1 \\ S_1;[True];S_2 \Leftrightarrow S_1;S_2 \end{cases}$$

Specification Canonical forms

Invariant

$$\{p\};S;\{q\}$$



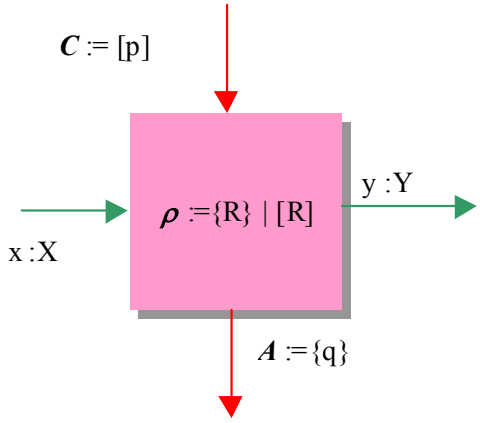
$$\{False\};S = Abort$$

$$\{True\};S = Skip;S = S$$

$$S;\{False\} = Abort$$

Synchronous

$$[p];S;\{q\}$$



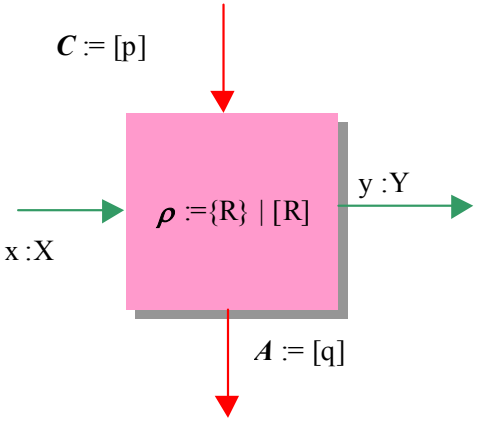
$$[True];S = Skip;S = S$$

$$[False];S = Magic;S = Magic$$

$$S;\{False\} = Abort$$

Event-driven

$$[p];S;[q]$$



$$[True];S = S$$

$$[False];S = Magic$$

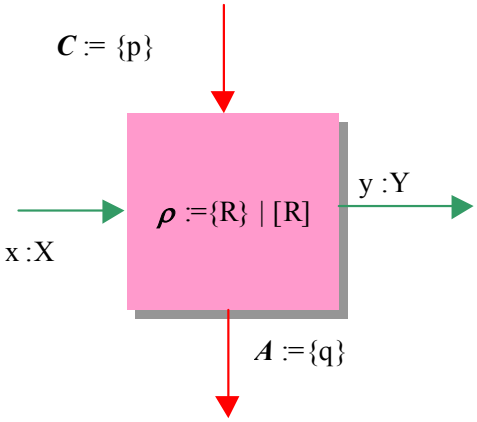
$$S;[False] = S;Magic = S$$

Application to functional specifications

TRANSPORT

ALSTOM

Function / Sub-function



Pre-condition weakening

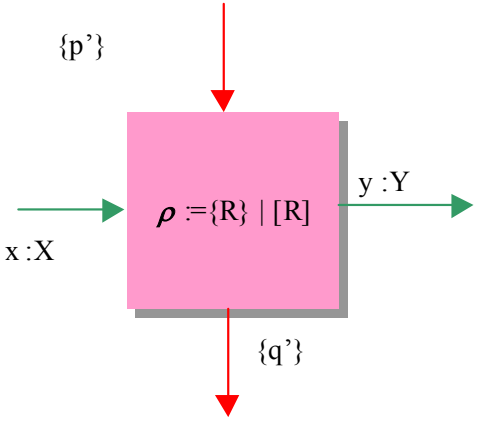
Pre-->pre'

•A refinement is a function that establish the same requirements under most unfavourable conditions

Post condition strengthening

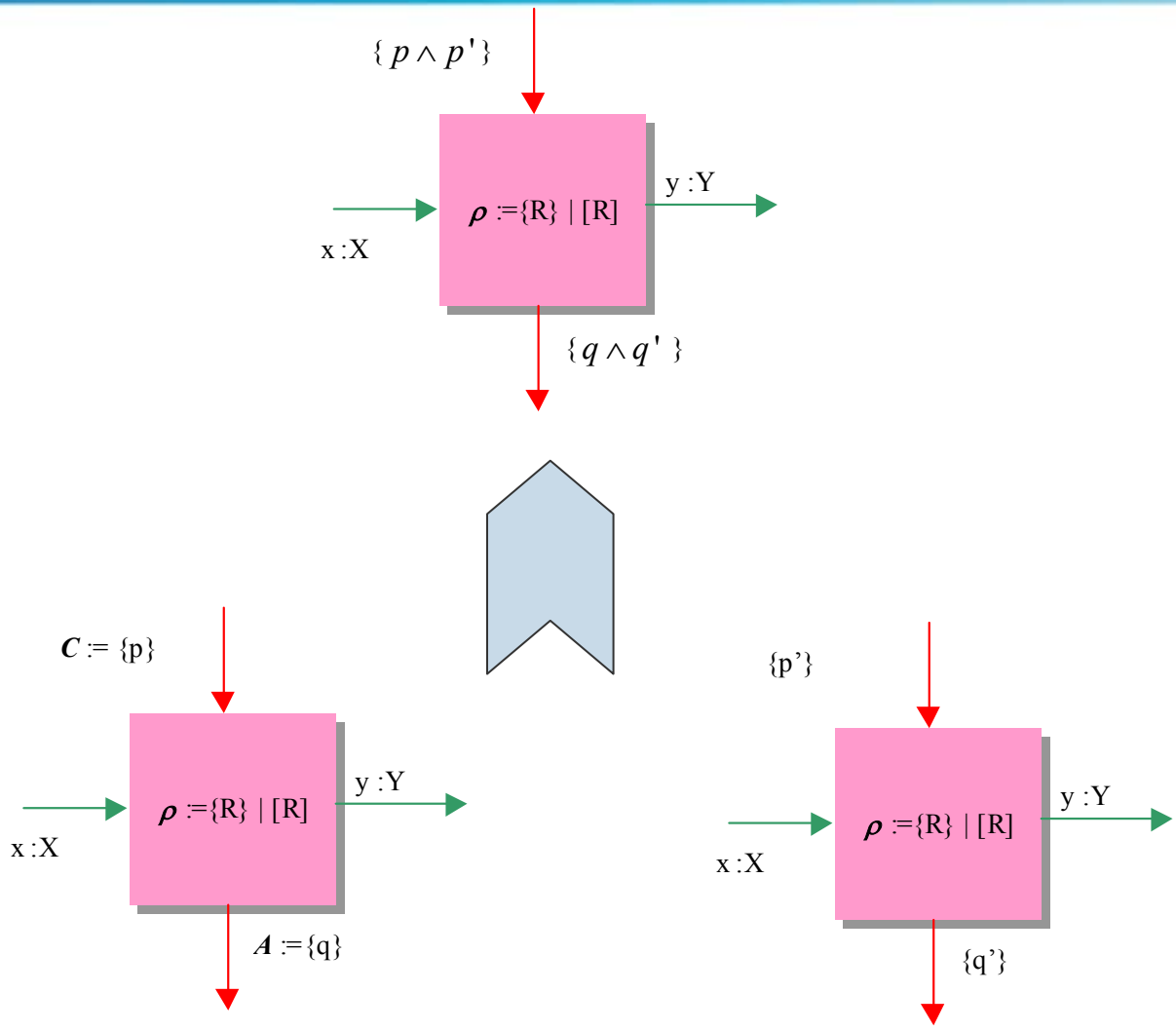
Post'-->post

•A refinement is a function that establish the stronger requirements under the same conditions

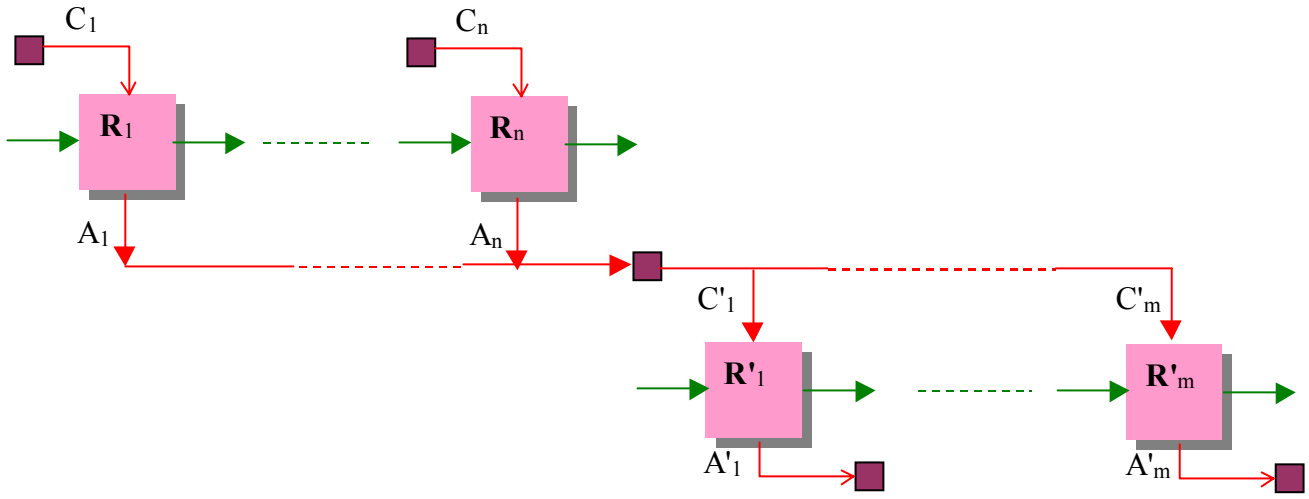


TRANSPORT ALSTOM

Function Abstraction



Function Composition rule

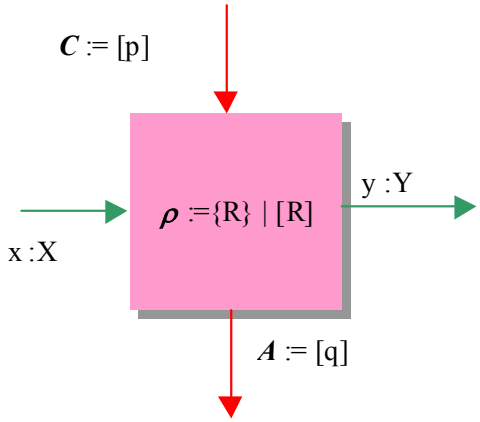


Functional specification composition

$$\bigcap_i A_i \Rightarrow \bigcap_j C_J$$

- All the preconditions must be established by at least “someone “

Mode and Sub-Mode



Input conditions strengthening

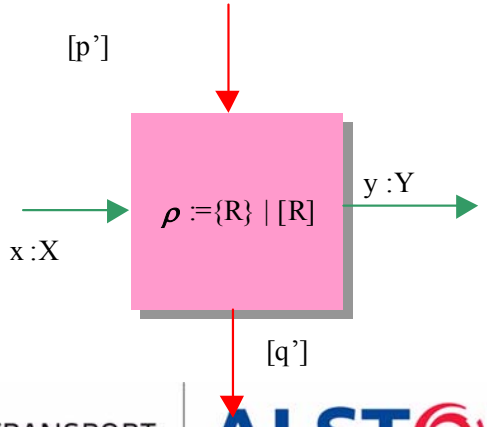
P'-->P

•To be inside a sub-mode implies to be inside the mode

Output condition weakening

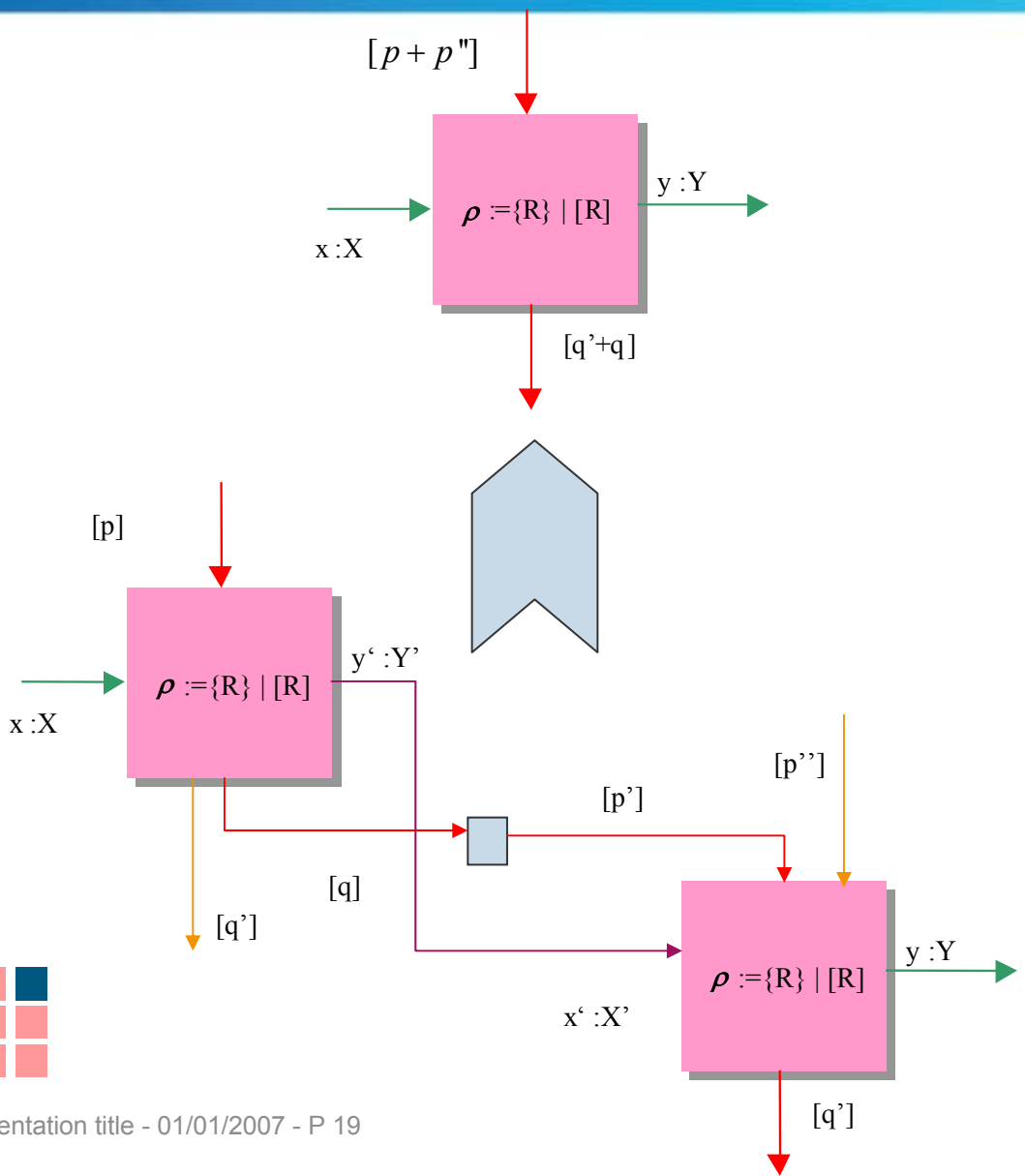
q-->q'

•To get outside the mode implies to get outside its sub-mode



TRANSPORT ALSTOM

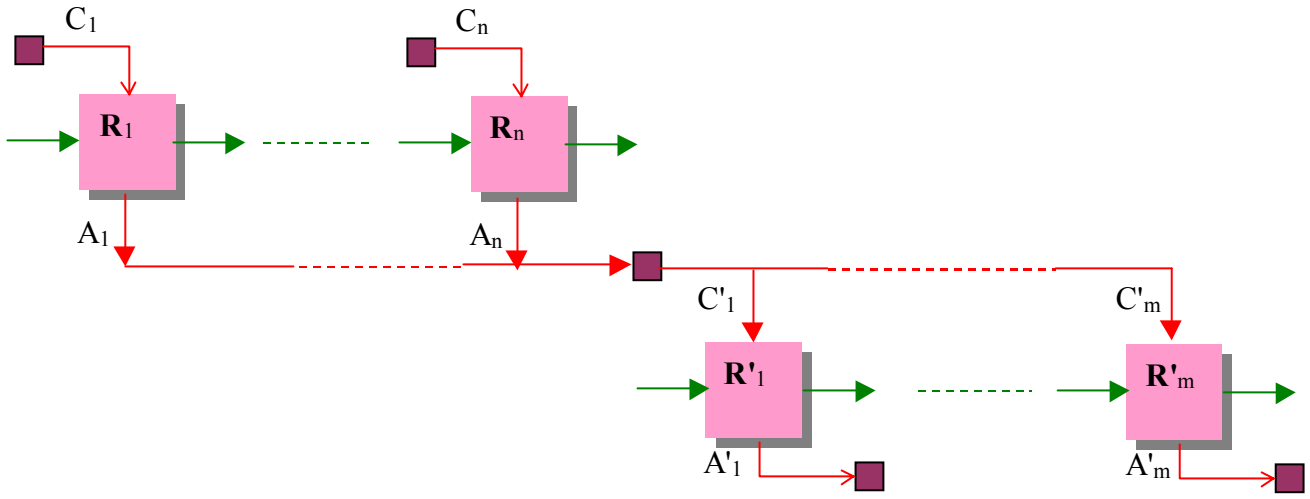
Mode abstraction



- abstraction of scenario

TRANSPORT ALSTOM

Mode Composition rules



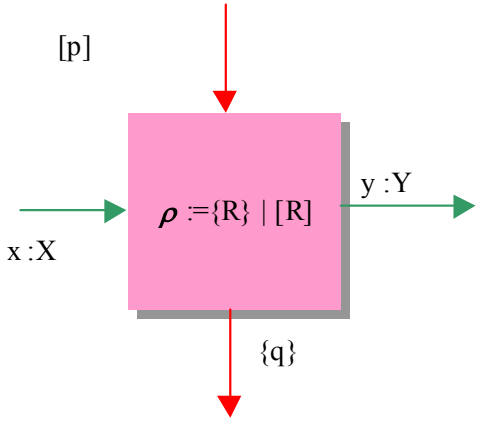
At least one,

at most one mode:

$$\left.\begin{array}{l} \bigcup_i A_i \Rightarrow \bigcup_j C_j \\ \bigcap_{\substack{i,j \\ i \neq j}} C_i \cap C_j \end{array}\right\} \quad +_i A_i \Rightarrow +_j C_j$$

I.e one & only one mode

Tasks and Sub-Task



Triggering strengthening

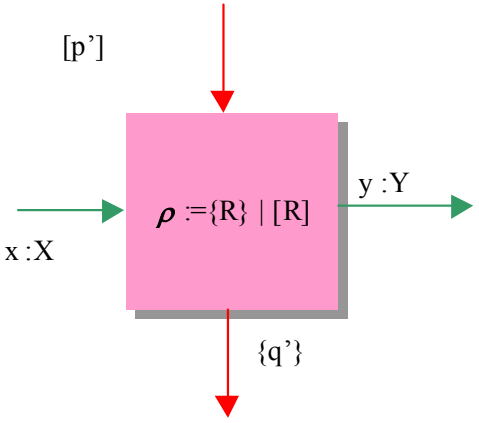
P'-->P

- the sub-task triggering implies the task triggering

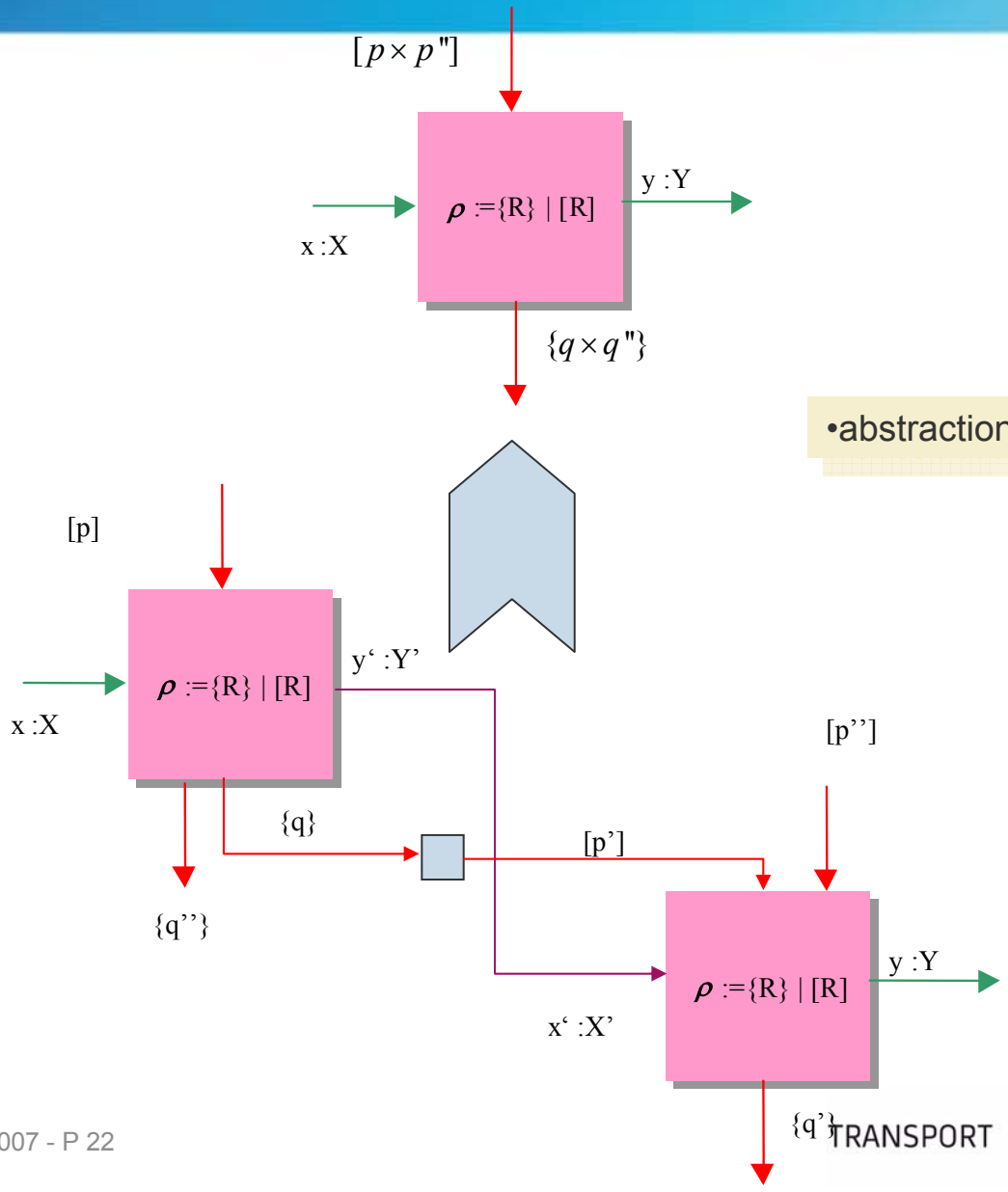
Post condition strengthening

Post'-->post

- A refinement is a task that establish the stronger requirements under the same conditions

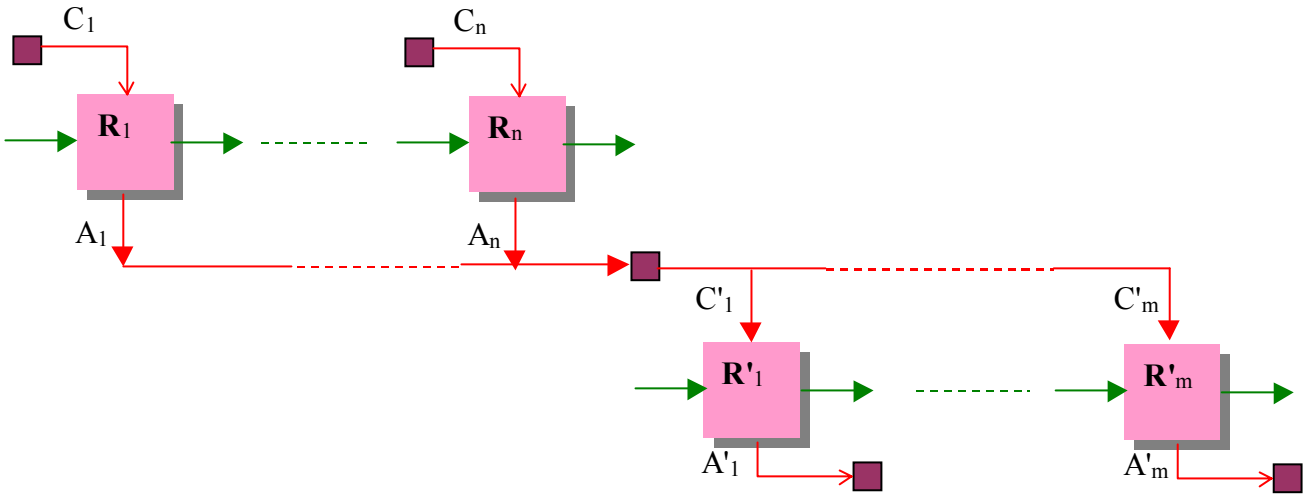


Tasks abstraction



•abstraction of task schedule

Task composition rule



$$\bigcup_i A_i \Rightarrow \bigcup_j C_j$$

$$\times_i A_i \Rightarrow \times C_j$$

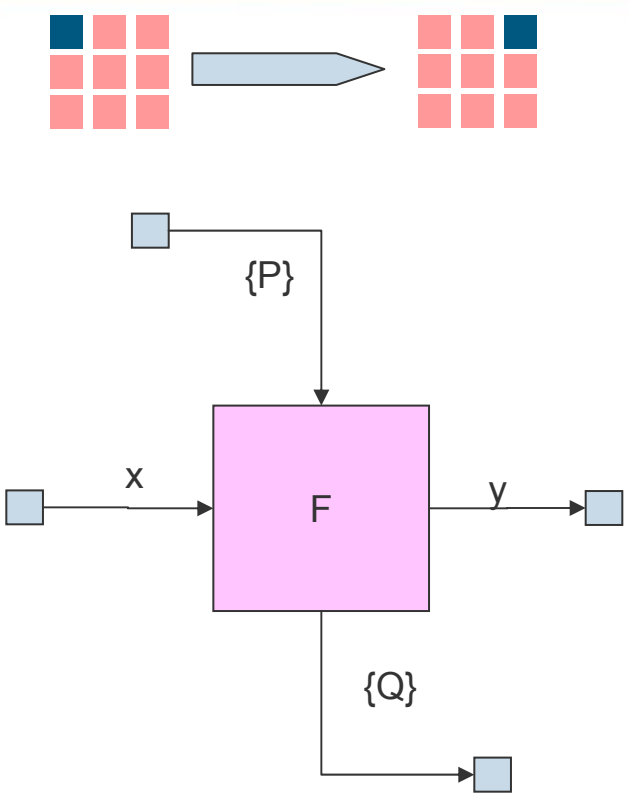
$$\bigcap_i A_i \Rightarrow \bigcup C_j$$

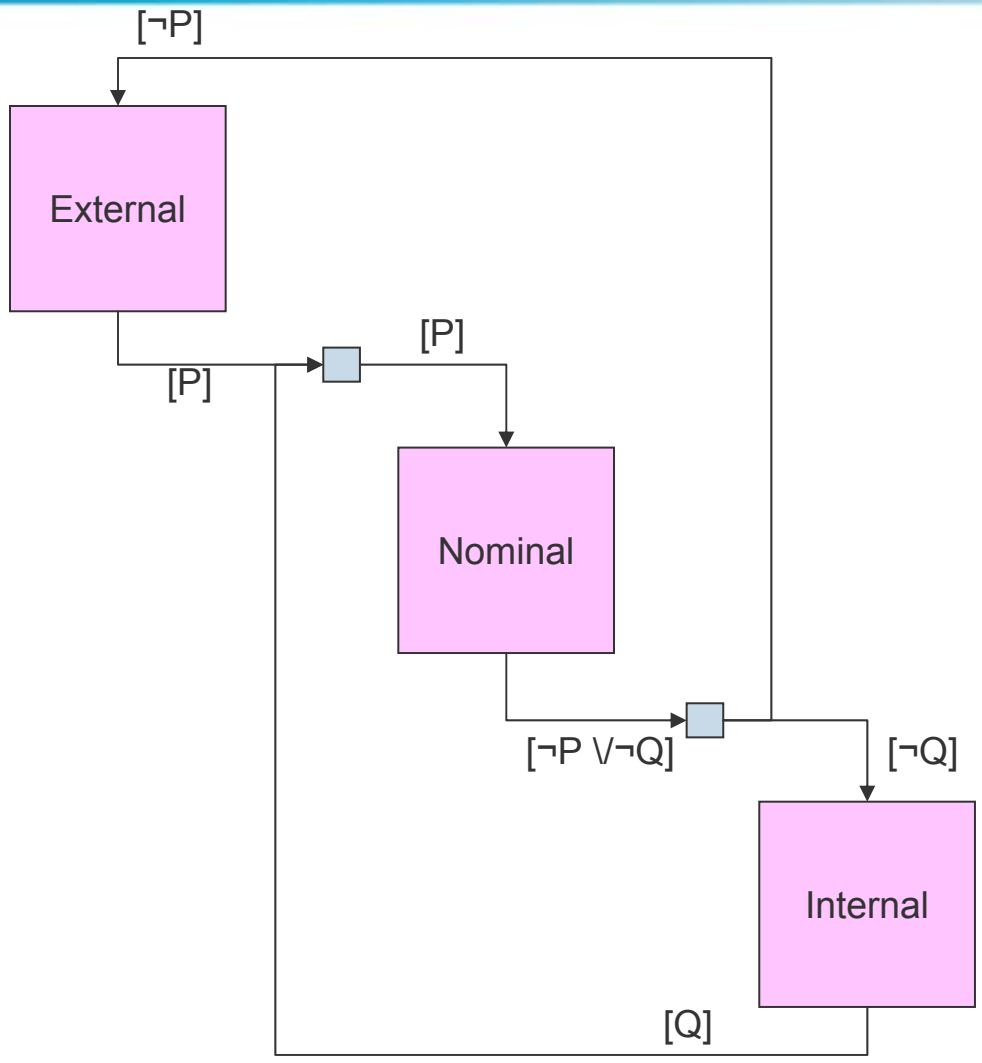
At least one task

$$\bigcap_i A_i \Rightarrow \bigcap_j C_J$$

No untriggerable task

Function versus Mode of operation



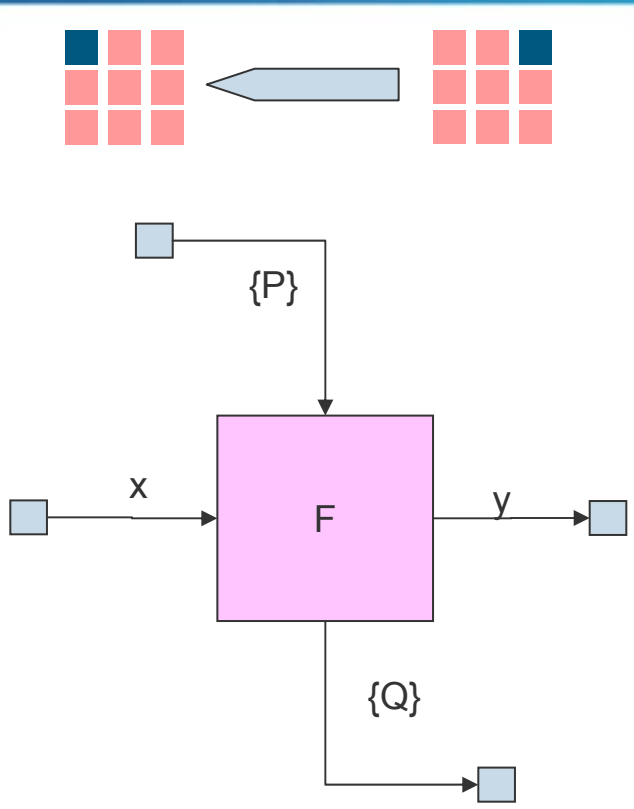


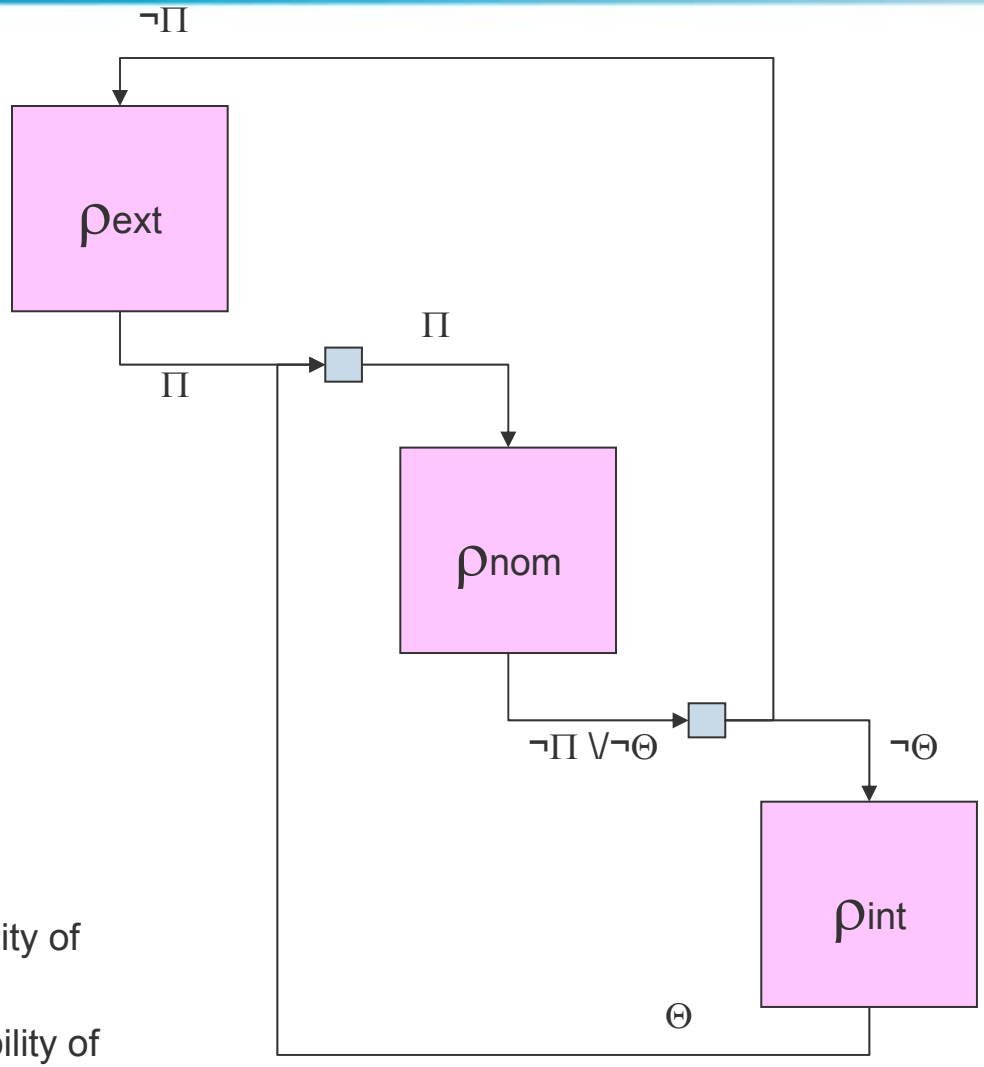
Nominal: $P \rightarrow Q$

External: $\neg P \rightarrow Q \vee \neg Q$

Internal: $P \rightarrow \neg Q$

RAM analysis using Probabilistic reasoning





If ¬Π and ¬Θ are respectively the probability of occurrence of

event **¬P** and **¬Q** then **ρnom** is the availability of function **F** in its nominal mode

CONCLUSIONS

- System theory is basically a modeling theory
- A formal semantic of the system vision is reachable
- System Engineering guaranteed by proof is theoretically possible
- Actually investigating GAME Semantic applied to System/Environment concurrent interaction (existence of a winning strategy)

www.alstom.com