FONDAZIONE
BRUNO KESSLER

Service Composition and Planning

Paolo Traverso
Bruno Kessler Foundation (FBK)
Center for Information Technology – IRST
Trento, Italy
traverso@fbk.eu
http://cit.fbk.eu/traverso

Agenda

FONDAZIONE BRUNO KESSLER

- Part I: Introduction to Software Services
- Part II: The Automated Composition Problem
- Part III: An Approach to Service Composition
- Part IV: A New Vision for the Internet of Services

Part I: Services …

- Services
  - business model from products to services …
  - services are used, they are not owned

Software Services …

FONDAZIONE BRUNO KESSLER

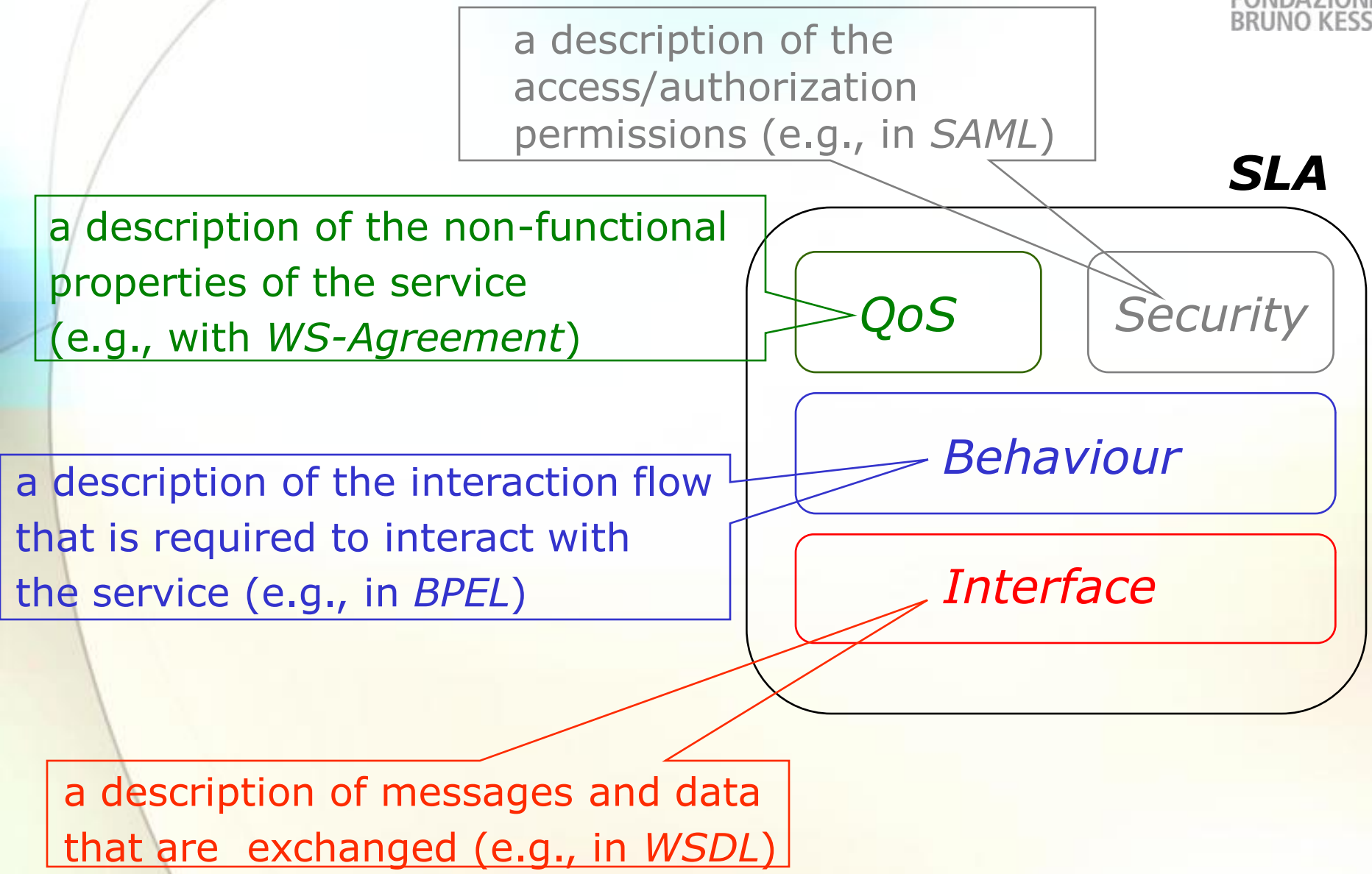
- Services
  - business model from products to services …
  - services are used, they are not owned
- Software services
  - software components that can be used …
  - … but are not owned
- Service-oriented applications
  - constructed by composing and configuring software services…
  - … most often provided by “third parties”
  - … software that is not under control

Key issues in software services

FONDAZIONE BRUNO KESSLER

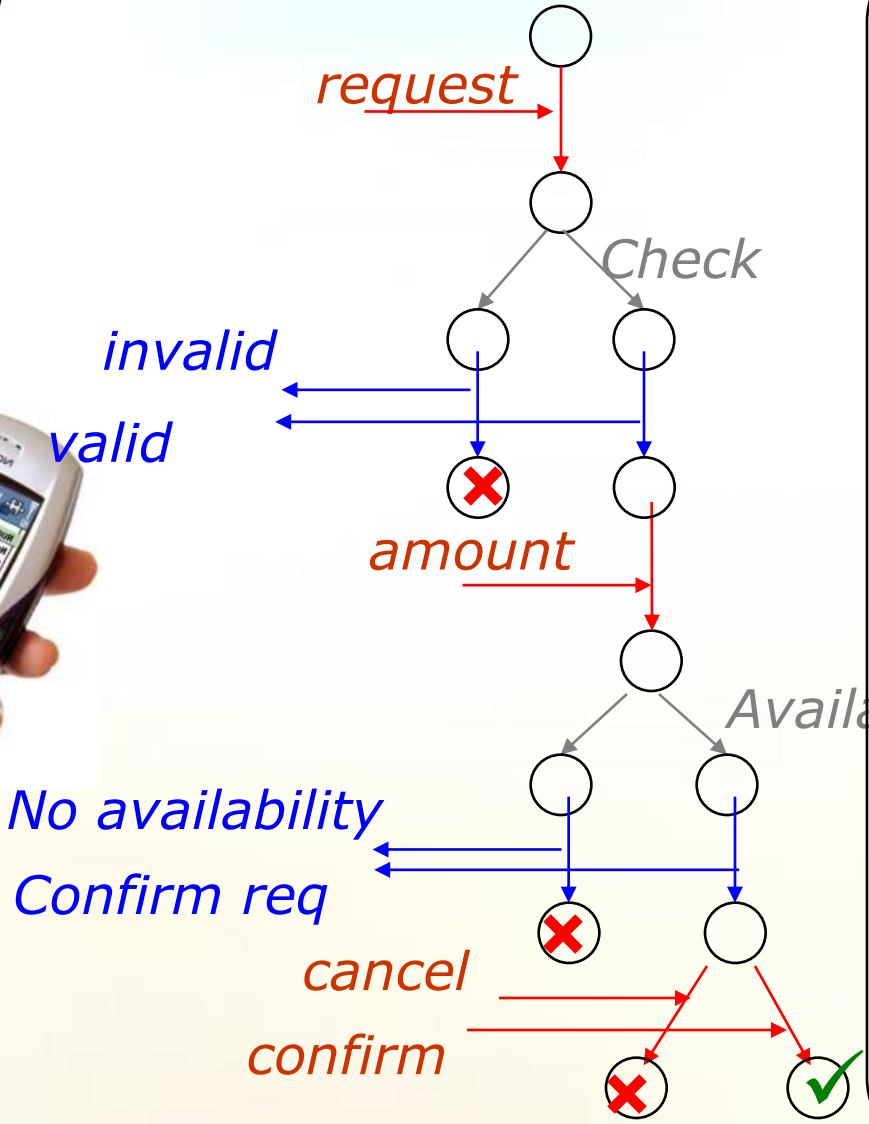
- Service Level Agreements:
  - to describe services to the rest of the world
- Service Composition:
  - to construct new services
- Service Monitoring & Adaptation:
  - to trigger the evolution of services and make them evolve

Service Level Agreements (SLA)



Example: Behaviour SLA

The interaction flow for an on-line bank payment (e.g. in BPEL)



BANK
ON-LINE
SERVICE

Outline

1. Service Level Agreements:
   - to describe services to the rest of the world
2. Service Composition:
   - to construct new services
3. Service Monitoring & Adaptation:
   - to trigger the evolution of services and make them evolve

Some work in automated composition

FONDAZIONE BRUNO KESSLER

Automata-based e-service coordination:

- Formal framework for composing e-services from behavioral descriptions given in terms of automata.

Hull, Benedikt, Christophides. PODS 2003

Satisiability-based e-service coordination:

- decision procedures for satisfiability are used to address the problem of e-coordination

Berardi, Degiacomo, Mecella, Lenzerini. ICSOC 2004, 2005

QoS-aware service composition:

- genetic algorithms for the optimal QoS estimation

Canfora, Di Penta et al. ICSOC 2004, ICWS 2005

The Automated Composition Problem

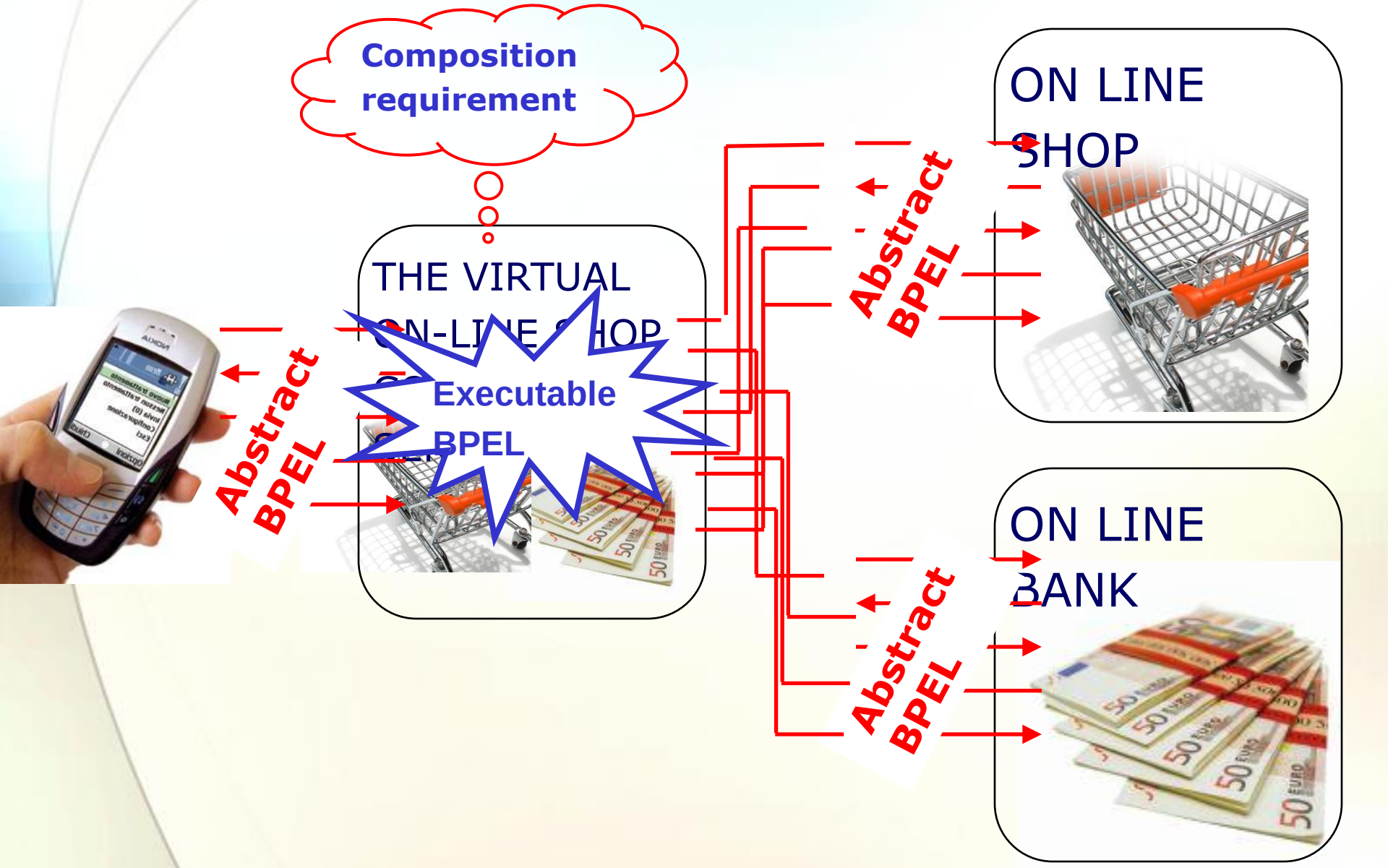
THE VIRTUAL ON-LINE SHOP COMPOSED SERVICE

ON LINE SHOP

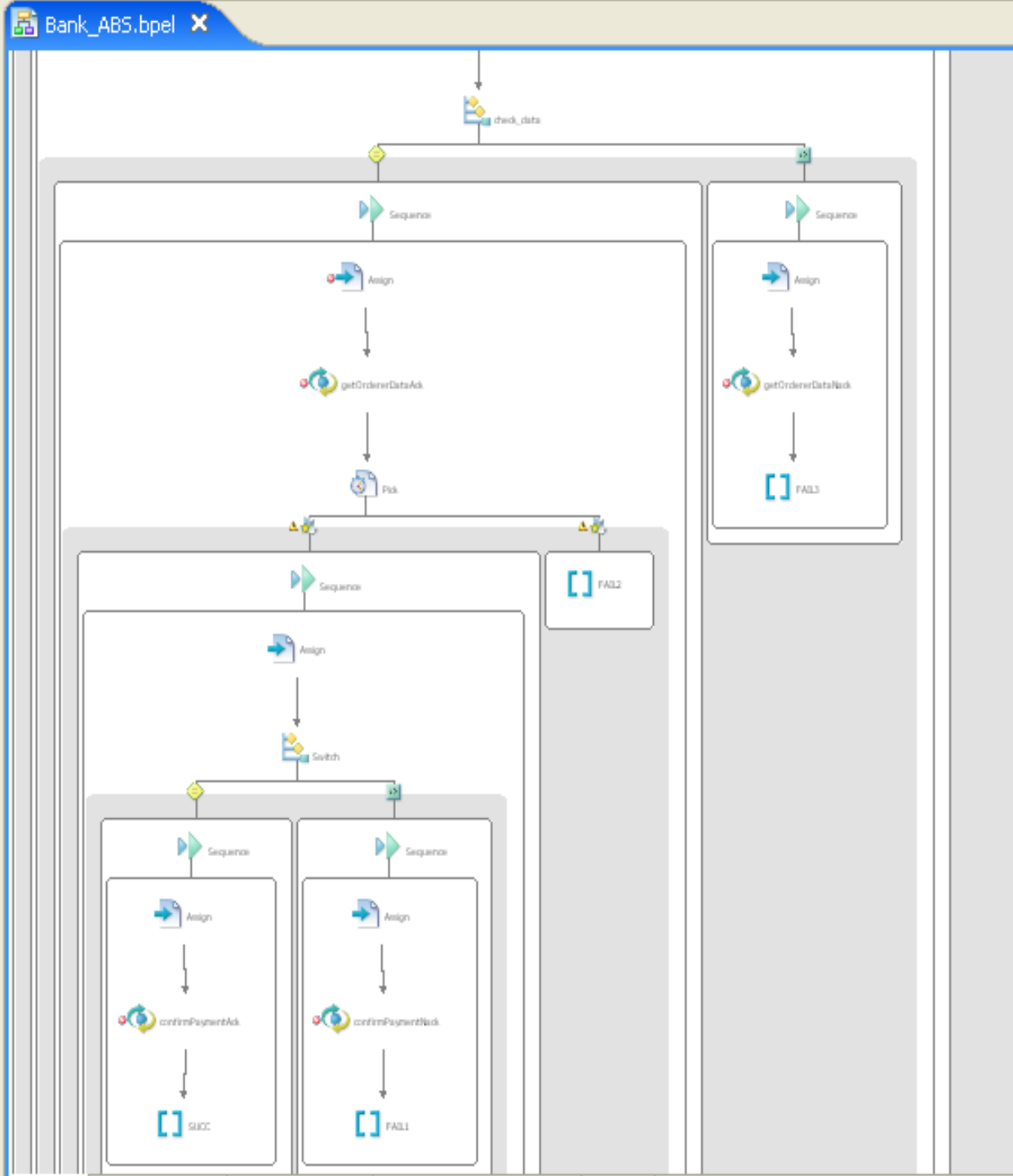
ON LINE BANK

Marconi, Pistore, Traverso. ISWC 2004, IJCAI 2005, SEMF 2006

The Automated Composition Problem



Abstract BPEL



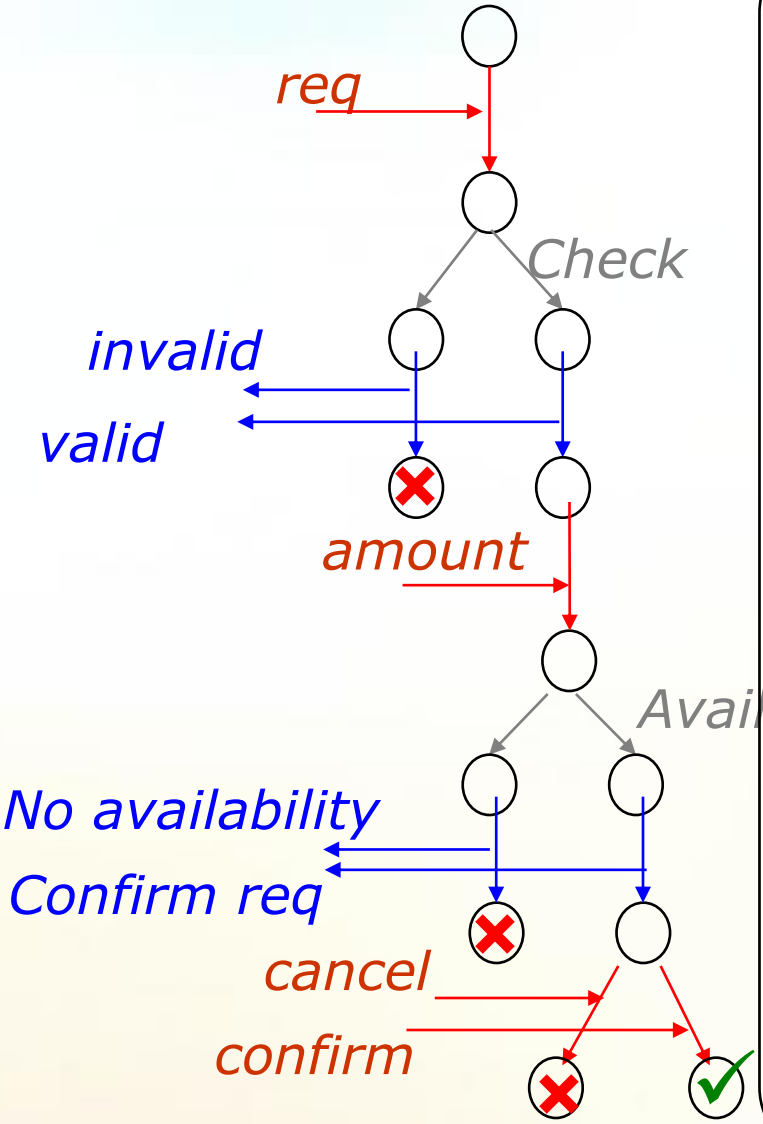
From Abstract BPEL to State Transition Systems

Abstract BPEL -> State Transition Systems

Input actions I (reception of messages)

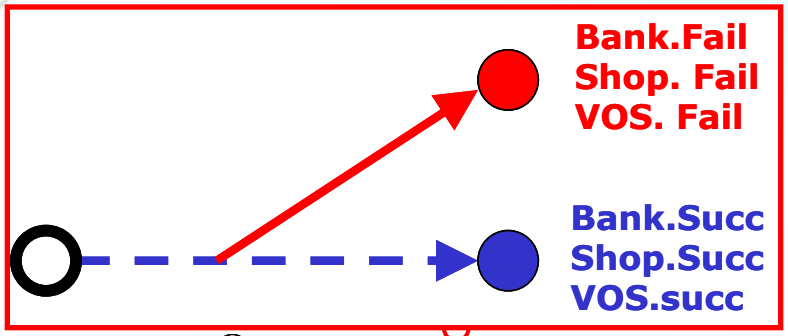
Output actions O (message sent)

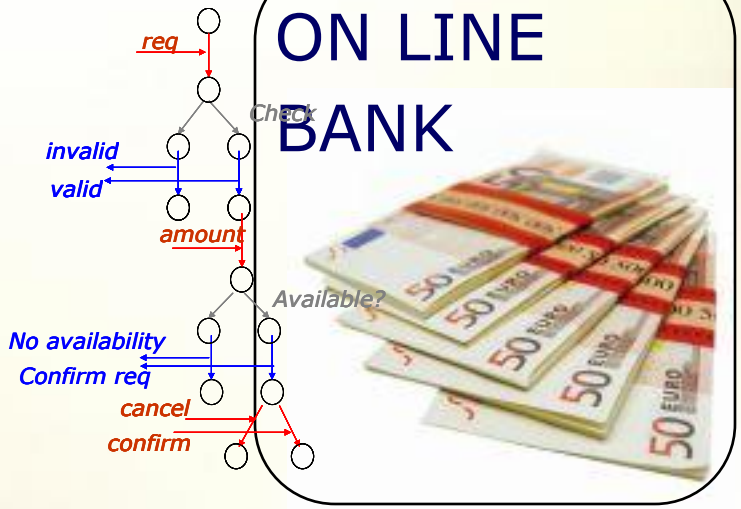
Internal action τ (internal evolutions that are not visible to external services)



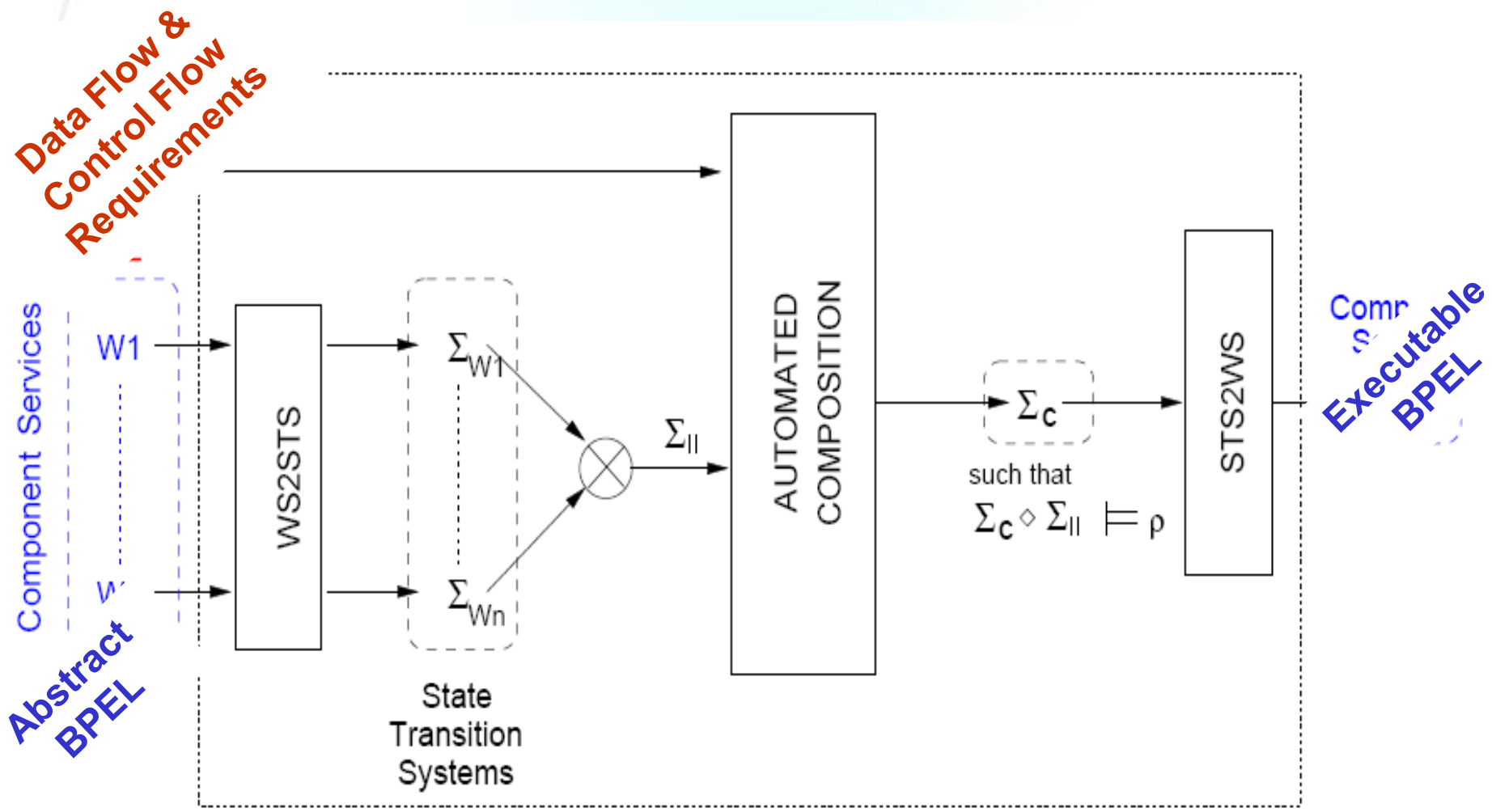
BANK ON-LINE SERVICE

The Automated Composition Problem





The Approach



The Approach

Definition. Let $\Sigma_1 = \langle \mathcal{S}_1, \mathcal{S}_1^0, \mathcal{I}, \mathcal{O}, \mathcal{R}_1 \rangle$ and $\Sigma_2 = \langle \mathcal{S}_2, \mathcal{S}_2^0, \mathcal{O}, \mathcal{I}, \mathcal{R}_2 \rangle$ be two complementary state transition systems. The **controlled system** $\Sigma_1 \Diamond \Sigma_2$ is the STS defined as:

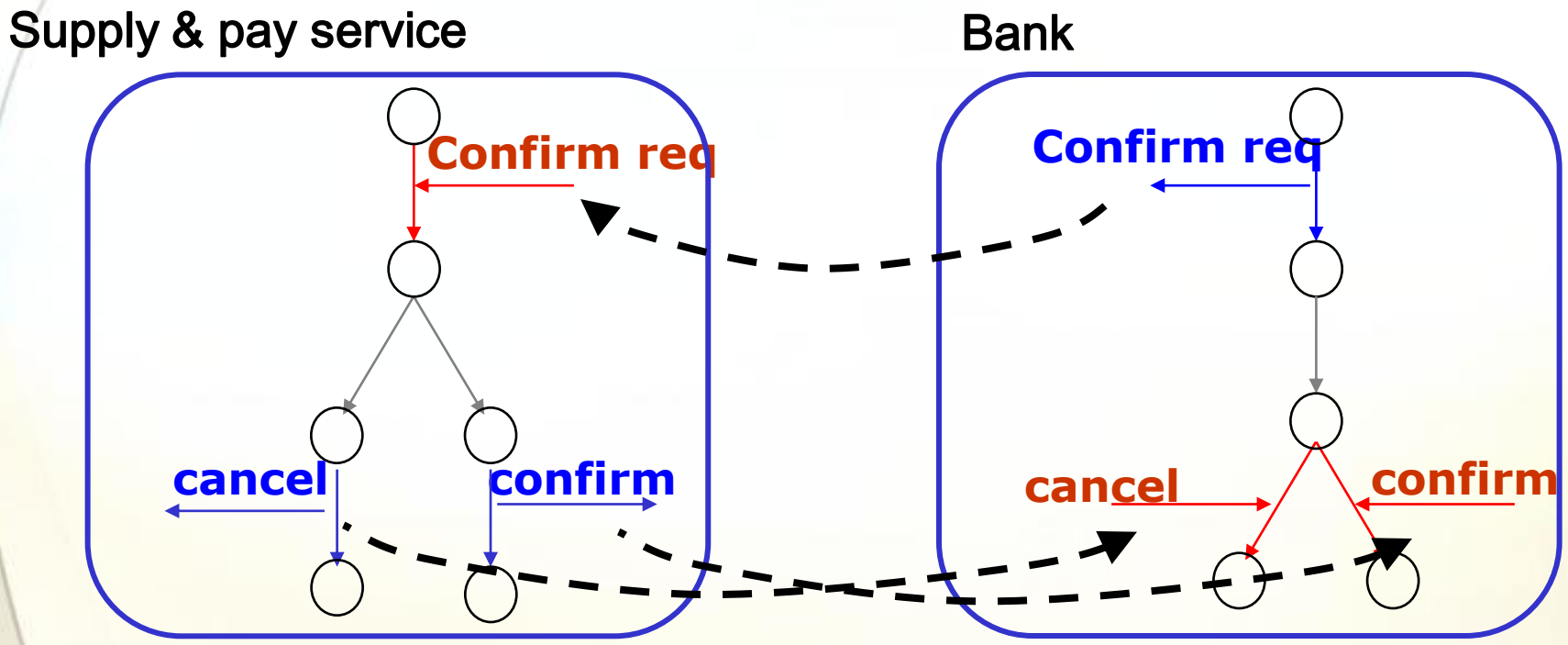
$$\Sigma_1 \Diamond \Sigma_2 = \langle \mathcal{S}_1 \times \mathcal{S}_2, \mathcal{S}_1^0 \times \mathcal{S}_2^0, \emptyset, \emptyset, \mathcal{R}_1 \Diamond \mathcal{R}_2, \rangle$$

where $\langle (s_1, s_2), \tau, (s_1', s_2') \rangle \in (\mathcal{R}_1 \Diamond \mathcal{R}_2)$ if

- $\langle s_1, \tau, s_1' \rangle \in \mathcal{R}_1$ and $s_2 = s_2'$;
- $\langle s_2, \tau, s_2' \rangle \in \mathcal{R}_2$ and $s_1 = s_1'$;
- $\langle s_1, a, s_1' \rangle \in \mathcal{R}_1$ and $\langle s_2, a, s_2' \rangle \in \mathcal{R}_2$ with $a \in \mathcal{I} \cup \mathcal{O}$.

The Approach

The Approach: Controlled System



Deadlock free composition

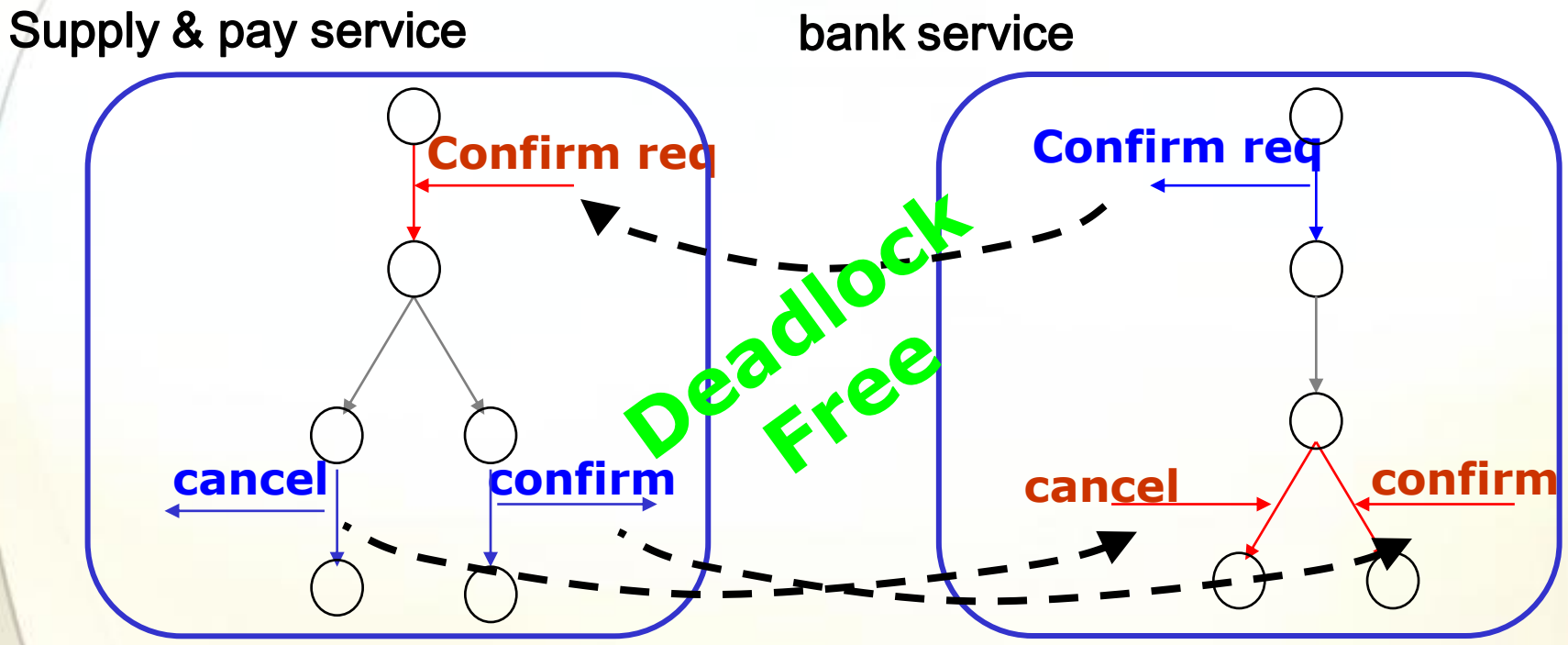
Deadlock-free composition:

Definition. Let $\Sigma_1 = \langle \mathcal{S}_1, \mathcal{S}_1^0, \mathcal{I}, \mathcal{O}, \mathcal{R}_1 \rangle$ and $\Sigma_2 = \langle \mathcal{S}_2, \mathcal{S}_2^0, \mathcal{O}, \mathcal{I}, \mathcal{R}_2 \rangle$ be two STS.
The controlled system $\Sigma_1 \diamond \Sigma_2$ is said to be **deadlock free** if all states $(s_1, s_2) \in \mathcal{S}_1 \times \mathcal{S}_2$ satisfy the following conditions:

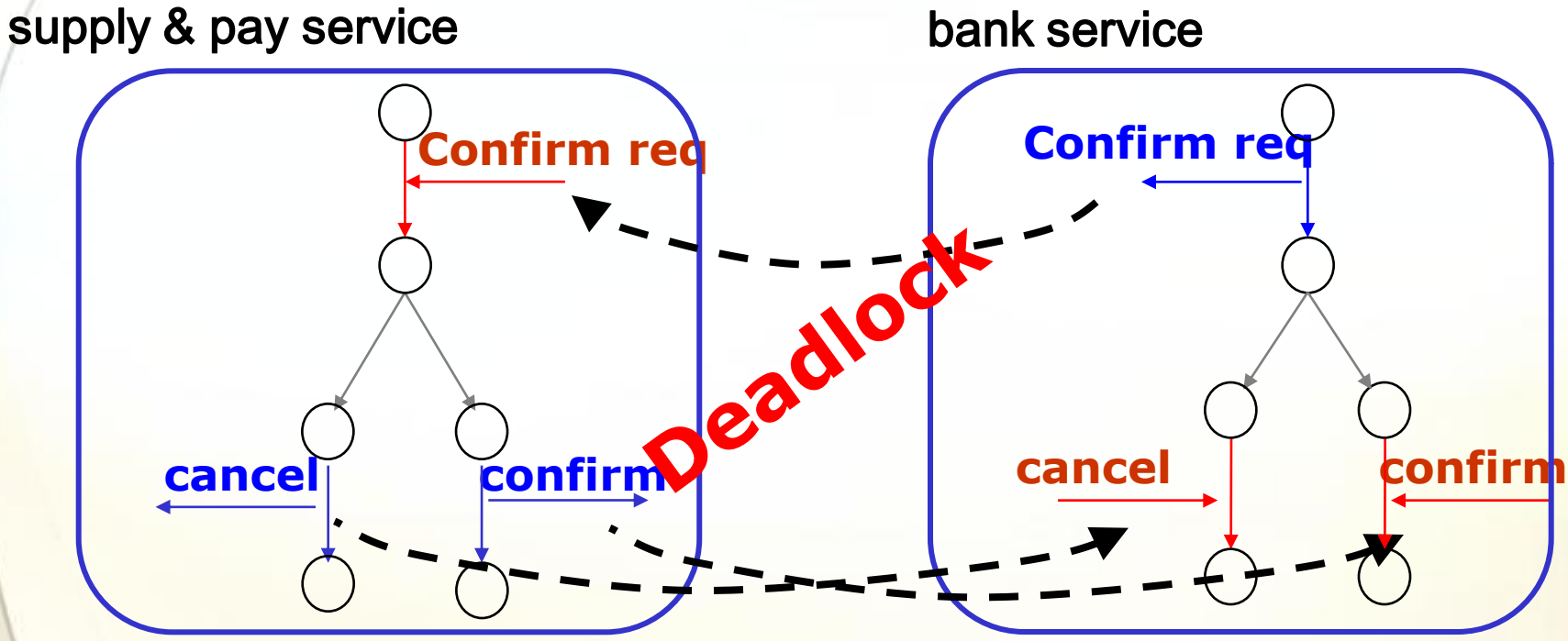
- if $\langle s_1, a, s_1' \rangle \in \mathcal{R}_1$ with $a \in \mathcal{O}$ then there is some $s_2' \in \tau\text{-closure}(s_2)$ such that $\langle s_2', a, s_2'' \rangle \in \mathcal{R}_2$ for some $s_2'' \in \mathcal{S}_2$;
- if $\langle s_2, a, s_2' \rangle \in \mathcal{R}_2$ with $a \in \mathcal{I}$ then there is some $s_1' \in \tau\text{-closure}(s_1)$ such that $\langle s_1', a, s_1'' \rangle \in \mathcal{R}_1$ for some $s_1'' \in \mathcal{S}_1$.

τ-closure(s) denotes the set of states reachable from s performing transitions labelled by τ.

Deadlock free composition



Deadlock free composition



Synthesis: the automated composition problem

Composition of web services:

- Starting from $\Sigma_{\|} = \Sigma_1 \,\|\, \cdots \,\|\, \Sigma_n$ and composition requirement ρ, find a controller Σ_c such that:

  1. requirement ρ is enforced:

$$\Sigma_c \Diamond \Sigma_{\|} \models \rho$$

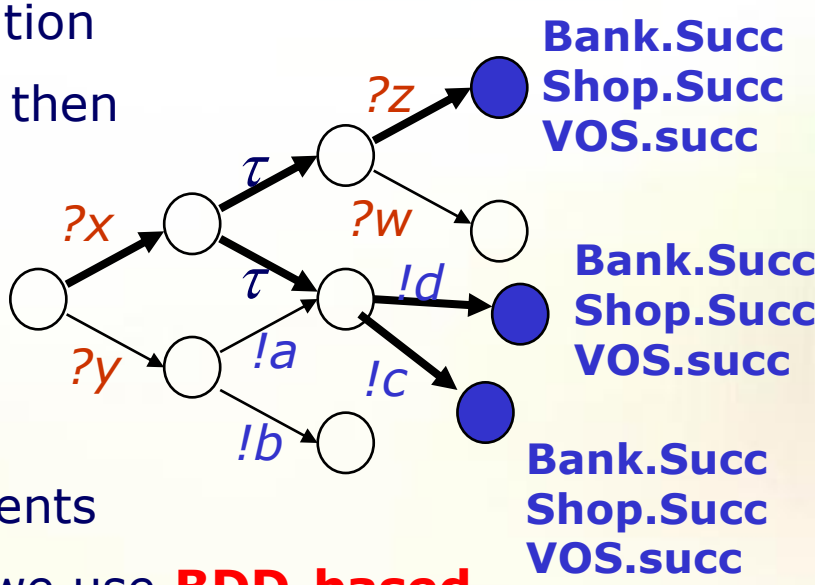
  2. the asynchronous interaction model is respected:

$$\Sigma_c \Diamond \Sigma_{\|} \text{ is deadlock-free}$$

The Composition Algorithm: Intuitions

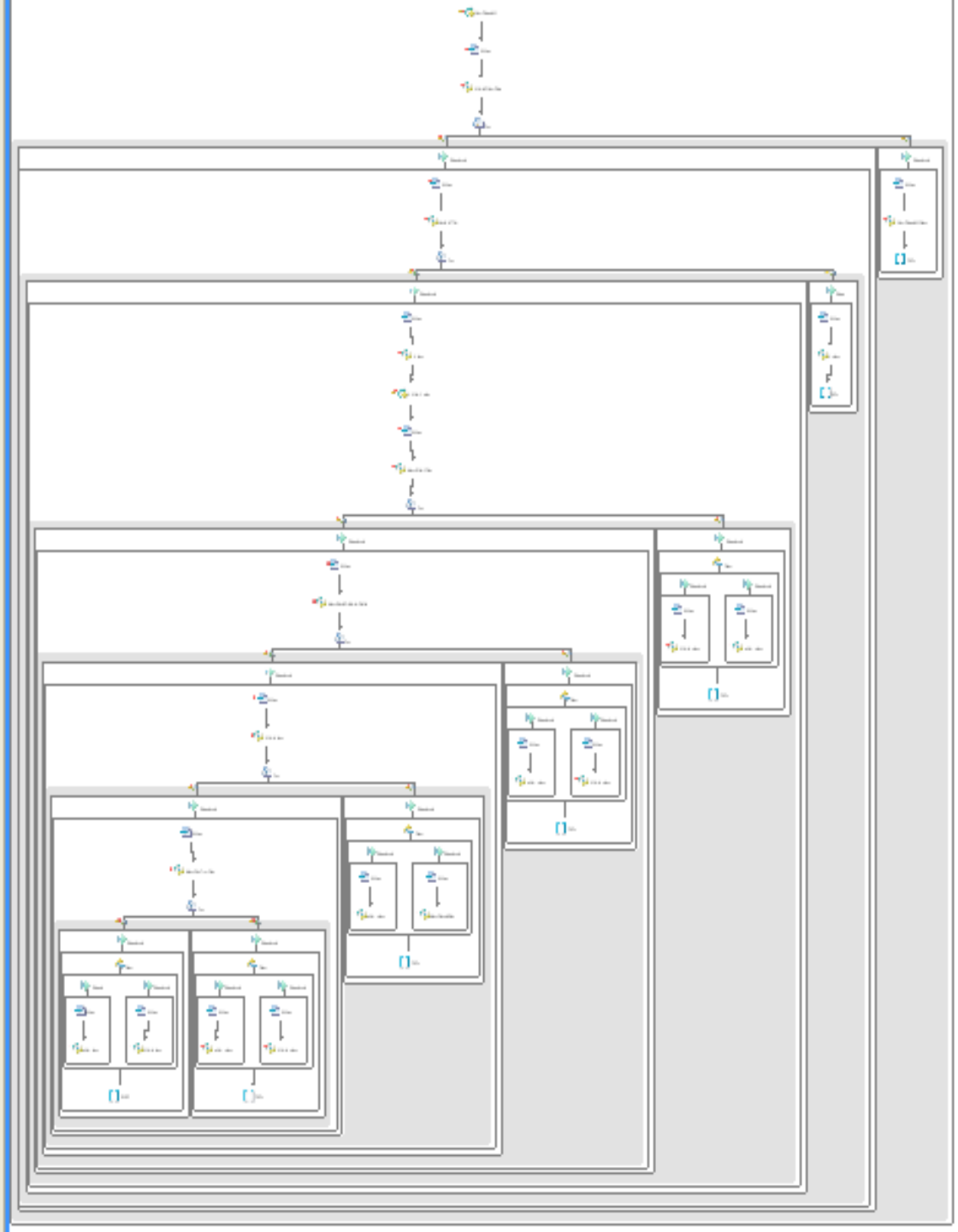
- The **Parallel Product** of the State Transitions Systems (STSs) of Available Interaction Flows (Components + Composed)
- Search the Product STS to satisfy the Composition Requirement
- Find a **subgraph of the Product** STS which satisfies the following conditions (example with **reachability conditions**):
  1. All terminal states satisfy the condition
  2. If a state belongs to the subgraph, then
     - **a. one outgoing input**
     - **b. all outgoing taus**
     - **c. all outgoing outputs**

     belong to the subgraph
  3. remove non deadlock-free components

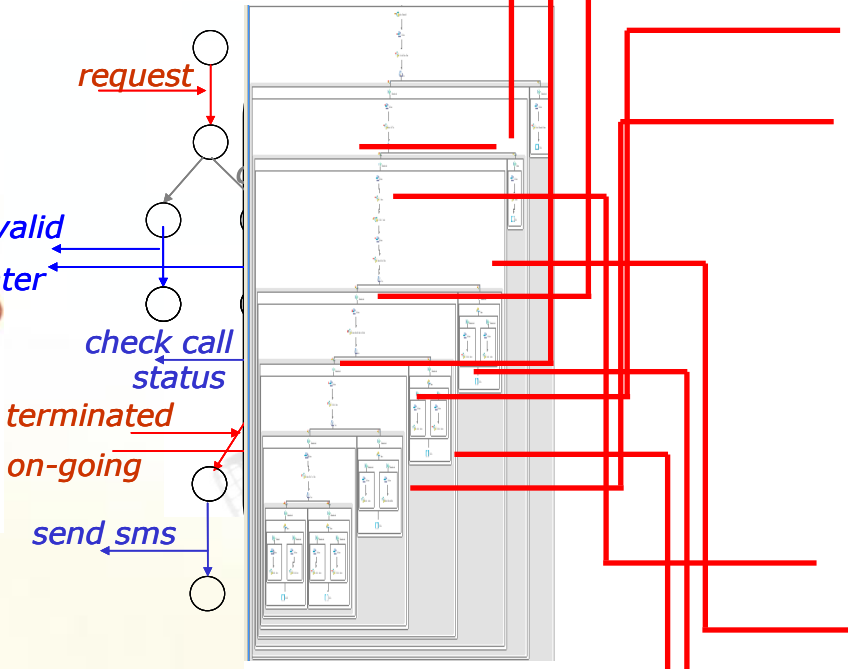


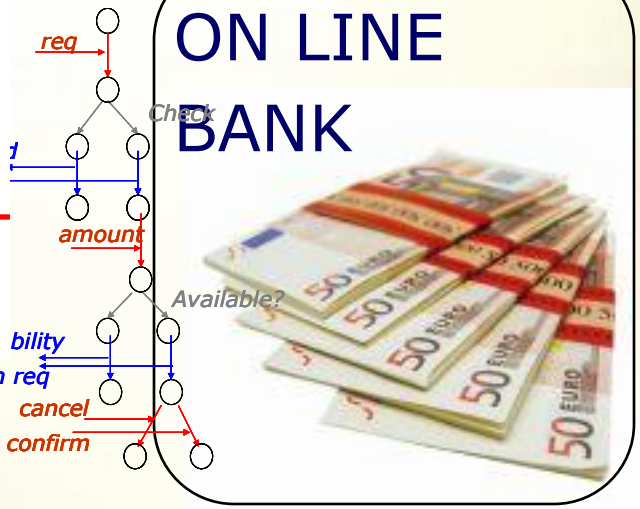
- Product STSs can be extremely large: we use **BDD-based exploration primitives** from the "**Planning as Model Checking**" framework

The Automatically G



Deployment of Executable BPEL





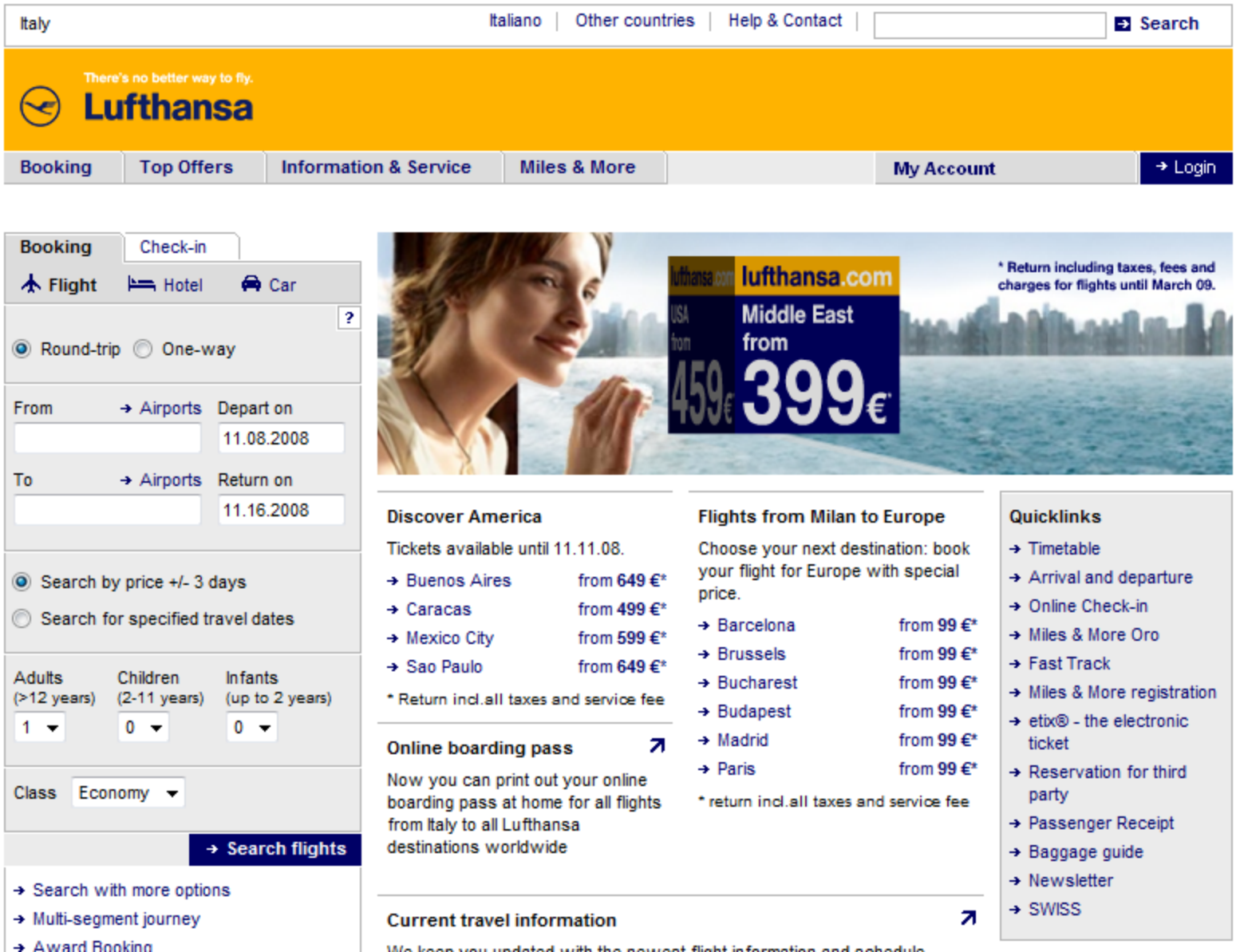
Tools available at www.fbk.eu

Agenda

FONDAZIONE BRUNO KESSLER

- Part I: Introduction to Software Services
- Part II: The Automated Composition Problem
- Part III: An Approach to Service Composition
- Part IV: A New Vision for the Internet of Services

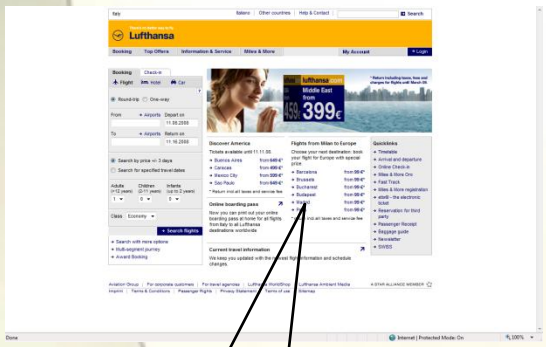
Example: Flight Service



Example: Flight Service

time

context



Book Flight

Flight delay

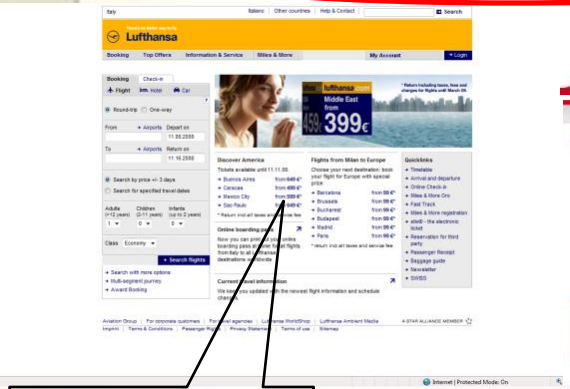
Your bag is on!

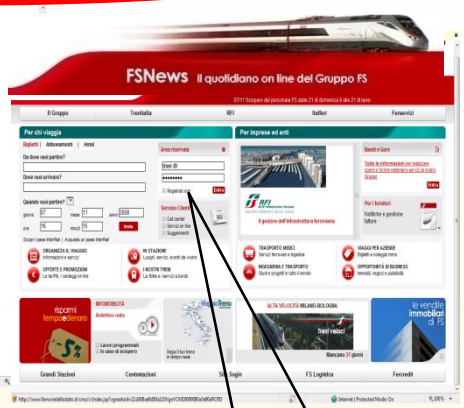
Connecting flights

Example: Service Composition

time

context





Book Flight

Book Train

Buy Movie Ticket

The Vision (one step forward …)

FONDAZIONE BRUNO KESSLER

Real Services are very different from Software Services, for instance with respect to …

→ **… duration:** the time for booking a travel is limited with respect to the duration of the actual travel.

→ **… dynamic:** software services are static and accessible anywhere and anytime; the real services are dynamic and context dependent.

→ **… coupling:** software services are independent and loosely coupled; the real services we use are strongly related.

The Vision (one step forward …)

FONDAZIONE BRUNO KESSLER

The SOC concepts and approaches have to be re-thought:

→ **… monitoring:** from monitoring the execution of software to monitoring the environment where the service operates

→ **… adaptation:** from adaptation among software services to adaptation to service and environment changes and to user's reactions

→ **… composition:** from task/goal driven composition of software to a composition based on how a service relates to core assets for the users

Conclusions

FONDAZIONE
BRUNO KESSLER

Thank you for your attention!