Reflective Architectures for adaptive information systems

International Conference on Service Oriented Computing 2003

A. Maurino
S. Modafferi
B. Pernici

Trento 17-12-2003

1

Index

► Reflective principle

► Distribution channel model

► Reflective architecture

► Adaptive strategies

► Future Work

2

Reflection Principle

• Appropriate metadata allow the architecture to know its state.

• By using the reflection principle the platform is able to evaluate, modify itself for the satisfaction of required features.

• We use the reflection principle to evaluate and modify the distribution channel along which the service is delivered

3

Distribution Channel Model
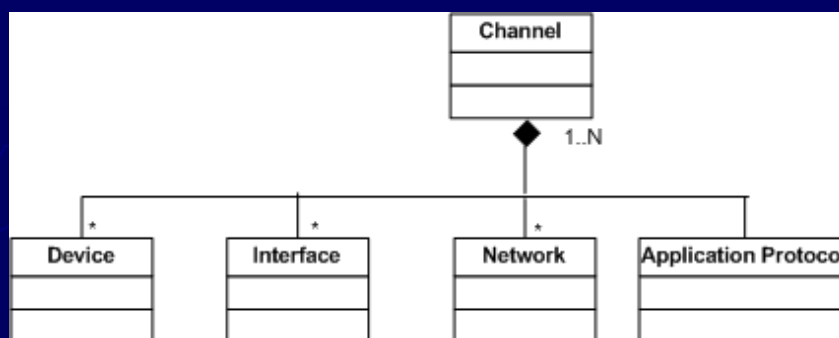
• Each logical channel is composed by a set of tuples

• Each logical channel has a common instance of at least a component
(i.e. all Internet Channel tuples have *http* as Application Protocol)
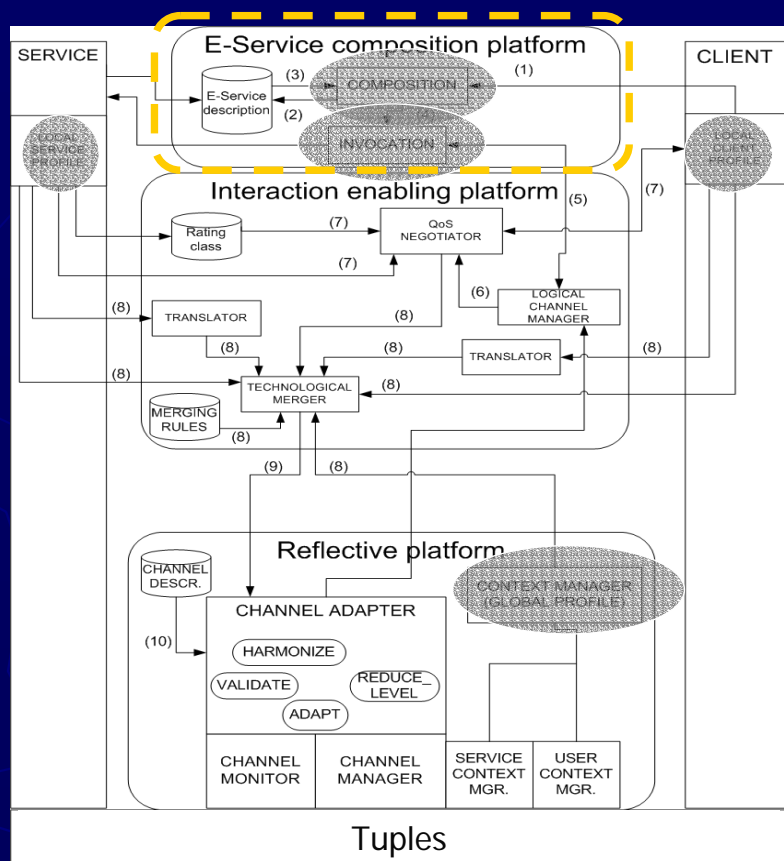
4

General Architecture

- modules are placed in the platform or in service/user device according to capability.

- Each profile is composed by a local and a global part.

E-Service Composition Platform

- Chooses e-service according to user request

- Invokes the chosen service

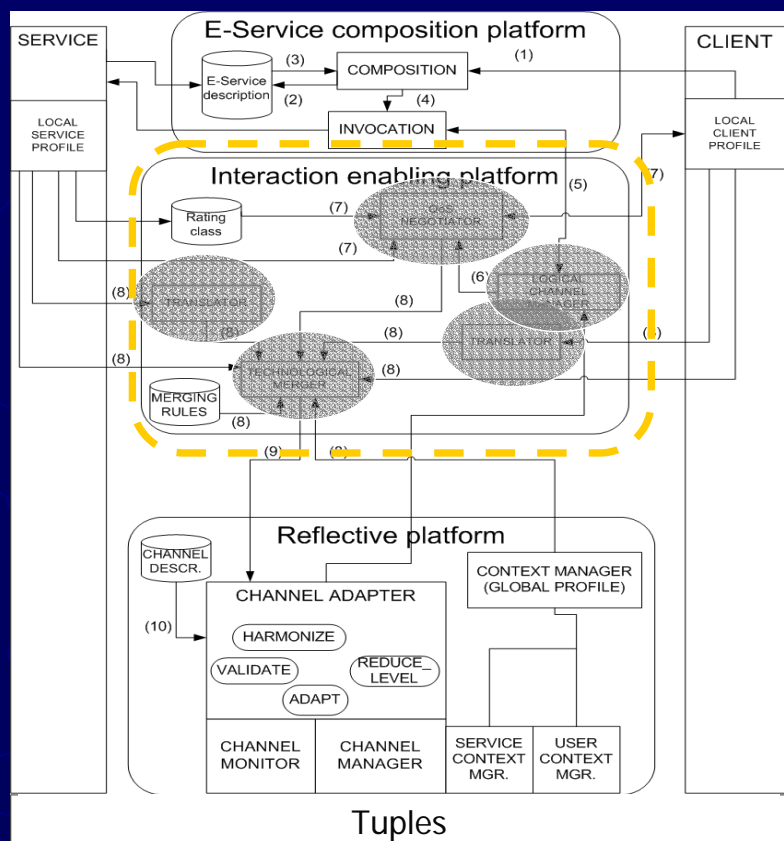---

General Architecture

Interaction Enabling Platform

- chooses the best tuple for service delivery

- determines QoS levels acceptable for the user

- translates logical constraints into technological ones

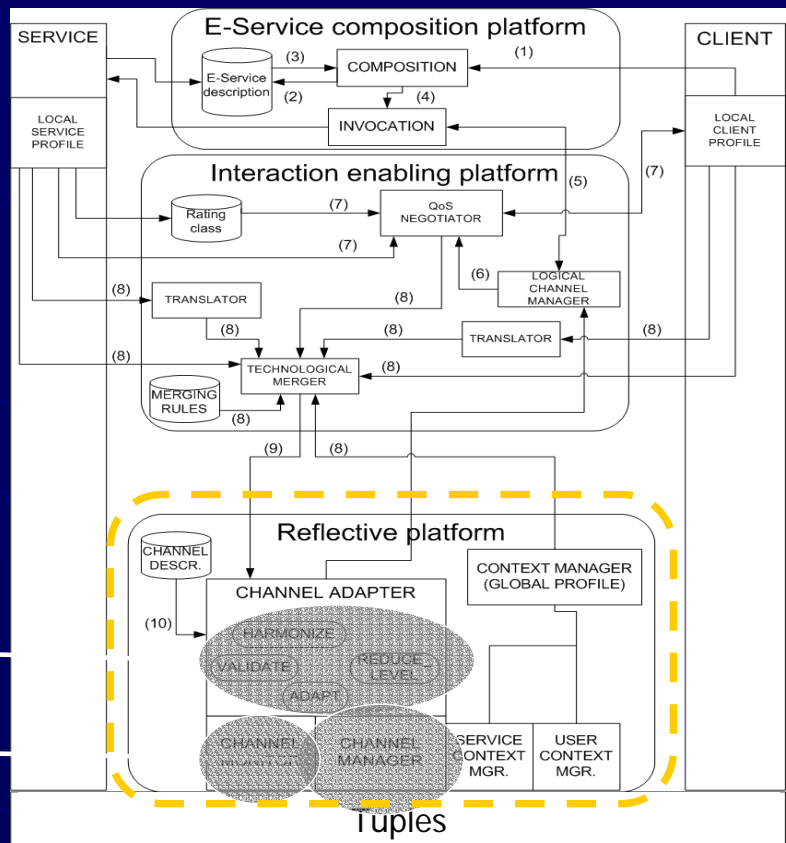- merges service/user/context constraints

General Architecture

Reflective Platform

- Works on a given tuple

- Input:
QoS levels with their constraints

- Attempts channel adaptation

- Monitors the channel during service provisioning

First phase

Second phase

Adaptive Strategies

Service								
Channel1					Channel2			
D_1	NI_1	N_1	AP_1		D_5	NI_1	N_1	AP_3
D_1	NI_2	N_1	AP_1		D_5	NI_2	N_1	AP_4
D_2	NI_2	N_1	AP_1					
D_3	NI_3	N_2	AP_1					
D_4	NI_3	N_2	AP_1					

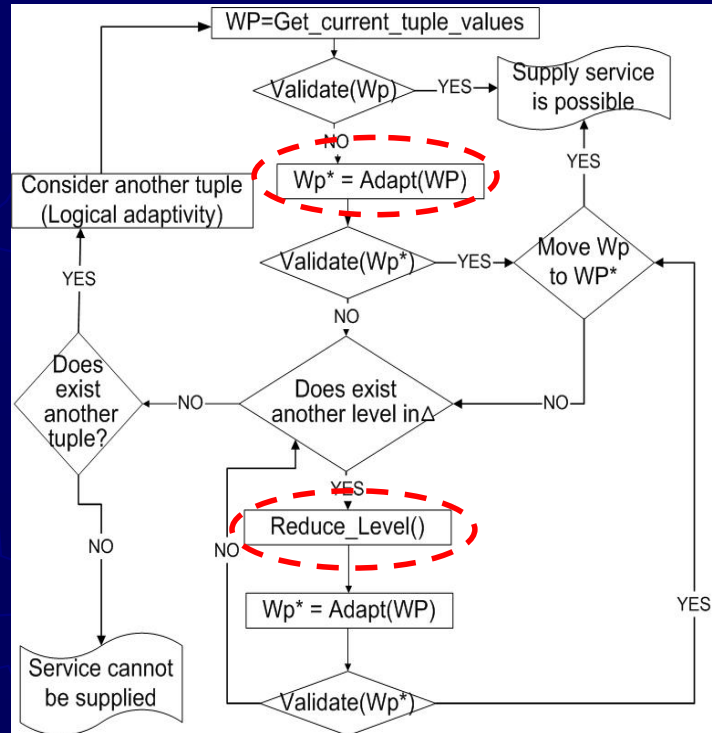Technological adaptability ←

Logical Adaptability

Reflective Platform: Technological Adaptivity

Interaction Enabling Platform: Logical Adaptivity

Multichannel strategies

8

Technological Adaptivity flow-chart

Wp= current working point

Wp*= proposed working point

Δ=set of acceptable QoS levels

Reflective Platform functions

Harmonize functions: combines related technological constraints

Example

Channel TransferRate composed by:
• Transfer Rate of Network
• Transfer Rate of Network Interface

$$\text{Harmonize}_{ni} ::= \begin{cases} V_{min}=0 \text{ Kb/s} \\ V_{max}= \min[\max(\text{TransferRate\_Interface}); \max(\text{TransferRate\_Network})] \end{cases}$$

Reflective Platform functions

Validate function:
evaluates if a working point satisfies all constraints.

$$Validate\ \ (Wp\ )\big|_{Ca\ ,Cc} = \begin{cases} 1 & if\ \ Wp\ \in (Cc\ \cap\ Ca\ ) \\ 0 & otherwise \end{cases}$$
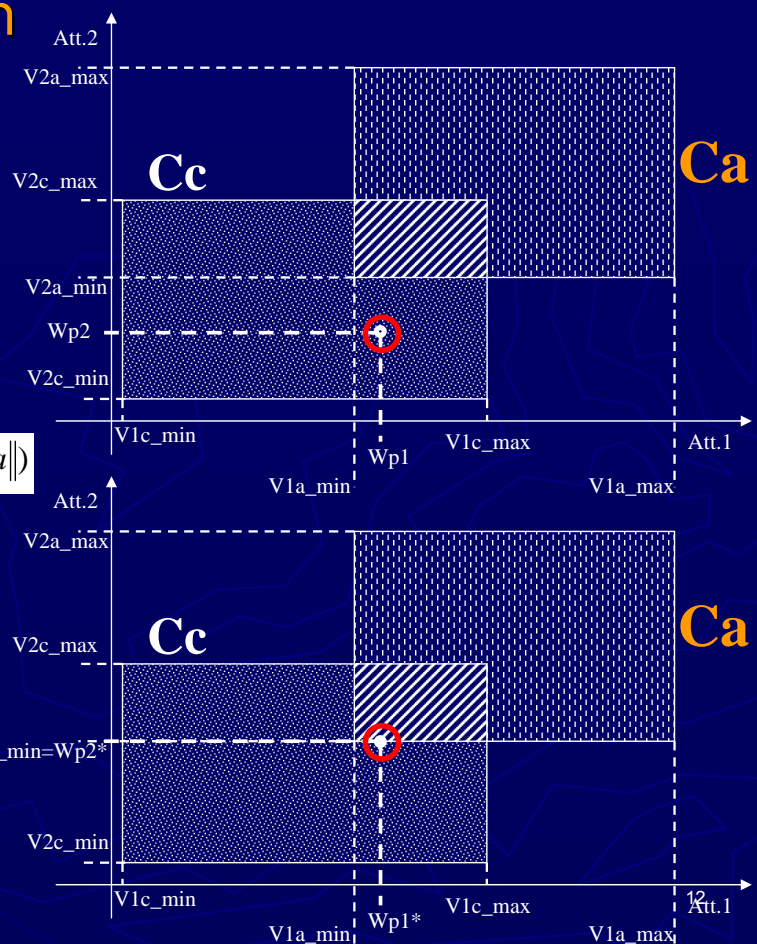
Where

•Ca: space of constraints from application (user/server requirements)
•Cc: space of exiting working points

Reflective Platform functions

Adapt function:
proposes a new working point
closest to user constraints set

$$Wp^* = Adapt(Wp)\big|_{(Ca,Cc)} = \arg\min(\|Wp; Ca\|)$$

Reflective Platform functions

Reduce_level function: modify service/user constraints considering a lower QoS level, that is a more relaxed constraints set

$$Ca^{new} = Reduce\_level(Ca)|_{\Delta}$$

Where Δ is the ordered set of QoS level acceptable for a given tuple

Reference Example

Internet Banking	Device					Interface		Network			Protocol		
	Type	Screen resolution	Number of colors	Audio	Input Device	Type	Transfer rate	Type	Transfer rate	Security	Type	Standar-disation	Security
n1	PC	C	C	C	O	Network Card	C	Wired	O	O	HTTP/SSL	O	O
n2	PC	C	C	C	O	Modem	C	Wired	O	O	HTTP/SSL	O	O
n3	TV	O	O	C	O	Modem	C	Wired	C	O	HTTP/SSL	O	O
n4	Mobile Phone	O	O	C	O	GPRS	C	GPRS	O	O	HTTP	O	O
n5	PDA	C	C	C/	O	GPRS	C	GPRS	O	O	HTTP	O	O

C = Controllable attributes   O = Observable attributes

Example

We assume that there exists a service that "*allow users, through Internet, to see in realtime interviews with financial analysts*"

Constraints (Ca):
- Sufficient Channel Bandwidht for real time video-streaming
- Audio On

We suppose that the choosen tuple for service delivery is n4
The following constraints sets are carried out from Interaction Enabling Platform (Ca)

QoS Level 0
- <150 kb/s,*,TransferRate,Interface>
- <150 kb/s,*,TransferRate,Network>
- <On,Audio,Device>

QoS Level 1
- <128 kb/s,*,TransferRate,Interface>
- <128 kb/s,*,TransferRate,Network>
- <On,Audio,Device>

Example

Wp
- <128 kb/s,TransferRate,Interface>
- <128 kb/s,TransferRate,Network>
- <Off,Audio,Device>

Cc_{n4}
- <0,128 kb/s,TransferRate,Interface>
- <0,128 kb/s,TransferRate,Network>
- <Off,On,Audio,Device>

Ca = QoS Level 0
- <150 kb/s,*,TransferRate,Interface>
- <150 kb/s,*,TransferRate,Network>
- <On,On,Audio,Device>

Then Validate(Wp) =0

Wp*=Adapt(Wp)=
- <128 kb/s,TransferRate,Interface>
- <128 kb/s,TransferRate,Network>
- <On,Audio,Device>

Example

Wp*

$\{$
<128 kb/s,TransferRate,Interface>
<128 kb/s,TransferRate,Network>
<On,Audio,Device>

Cc_{n4}

$\{$
<0,128 kb/s,TransferRate,Interface>
<0,128 kb/s,TransferRate,Network>
<On,Audio,Device>

Ca = QoS Level 0

$\{$
<150 kb/s,*,TransferRate,Interface>
<150 kb/s,*,TransferRate,Network>
<On,Audio,Device>

Still Validate(Wp*) =0

Ca^{new} = Reduce_level(Ca)= QoS Level 1 = $\{$ <128 kb/s,*,TransferRate,Interface>
<128 kb/s,*,TransferRate,Network>
<On,Audio,Device>

Then Validate(Wp*) =1

Conclusions & Future Work

- ☐ General reflective architecture
- ☐ Enriched distribution channel model
- ☐ Adaptive strategies

- ☐ E-Service Composition
- ☐ Multichannel
- ☐ Repository description
  - • Mapping rules
  - • Merging rules
- ☐ Tuple choice policy

Question

modafferi@elet.polimi.it

maurino@elet.polimi.it

pernici@elet.polimi.it