



Ubiquitous Service Coordination – Ubiquitous SOA – Concept and Applications

Michiharu Takemoto and Makoto Yoshida
NTT Network Service Systems Labs.



Proposal of “Ubiquitous SOA”

Motivation

- Provide new services **immediately**
 - >> Grid/Web service approach
- Provide **dynamic** services
 - >> Select the suitable components dynamically
- Let network components be used **widely**
 - >> Convergence between telecom SOA and ordinary SOA components
- Include emerged network connected components (e.g., appliances, devices, network facilities)
 - >> Ubiquitous components

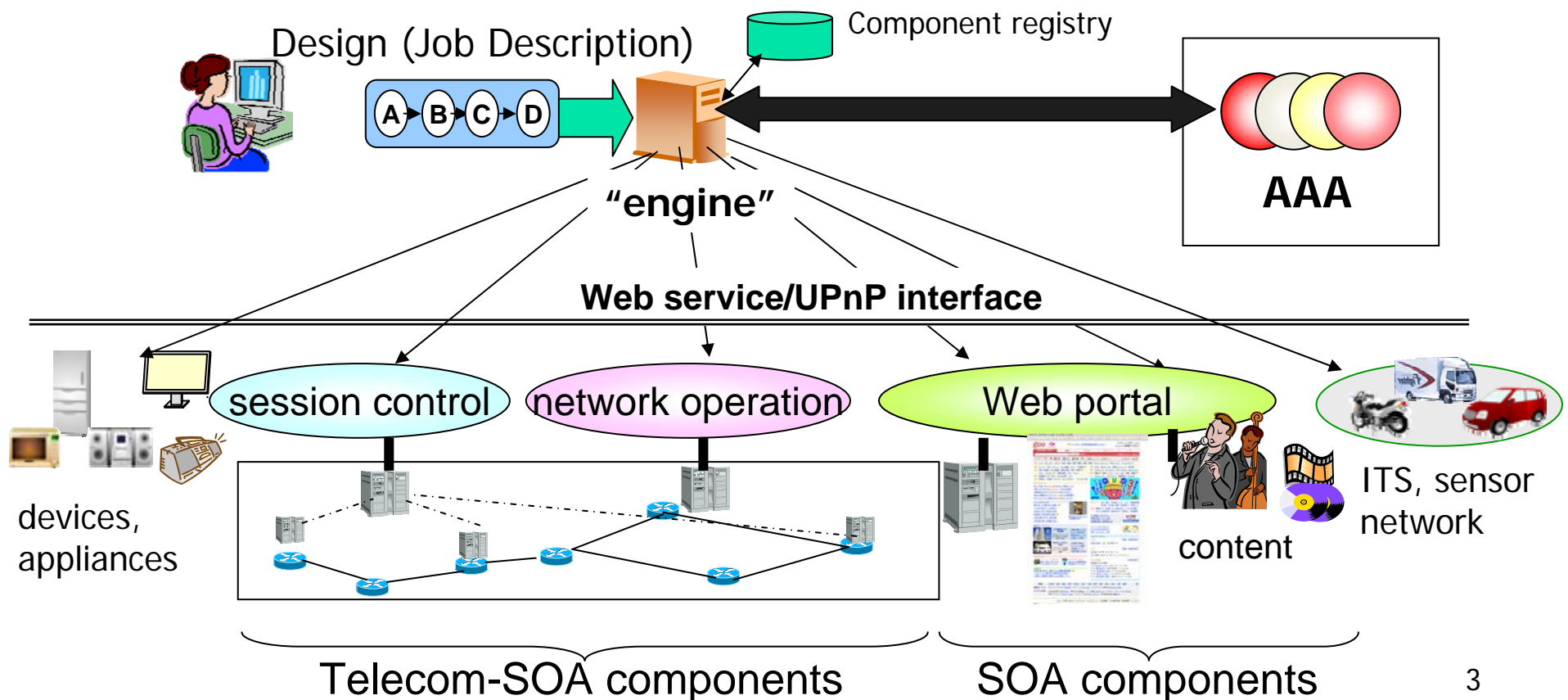
Feature of services

- Dynamic
- Suitable
- Easy to add / remove new components

>> **Proposal of “Ubiquitous SOA”**

Overview of Ubiquitous SOA

- Our ubiquitous service integration platform provides composition capabilities using appliances, sensors, telecom-SOA components as well as ordinary SOA components.
- Network operation functions, session control functions, and ordinary Web page functions are provided directly or through Web-service conversion wrappers.
- Application service providers, and even ordinary people, can provide application services by integrating components in the network.





Key technologies

Dynamic Service Composition

We would like to provide many types of dynamic services using huge number of components, which are dynamically selected.

- Design-interpreter (“engine”)
 - > selects the components
 - > resolves the interfaces
 - > invokes the interfaces.
- Meta-data in components (Semantic Web Service based)
- Meta-level design (job description) (Semantic Web Service based)

Component example: network resource management facility (ex. 2)

State Handling

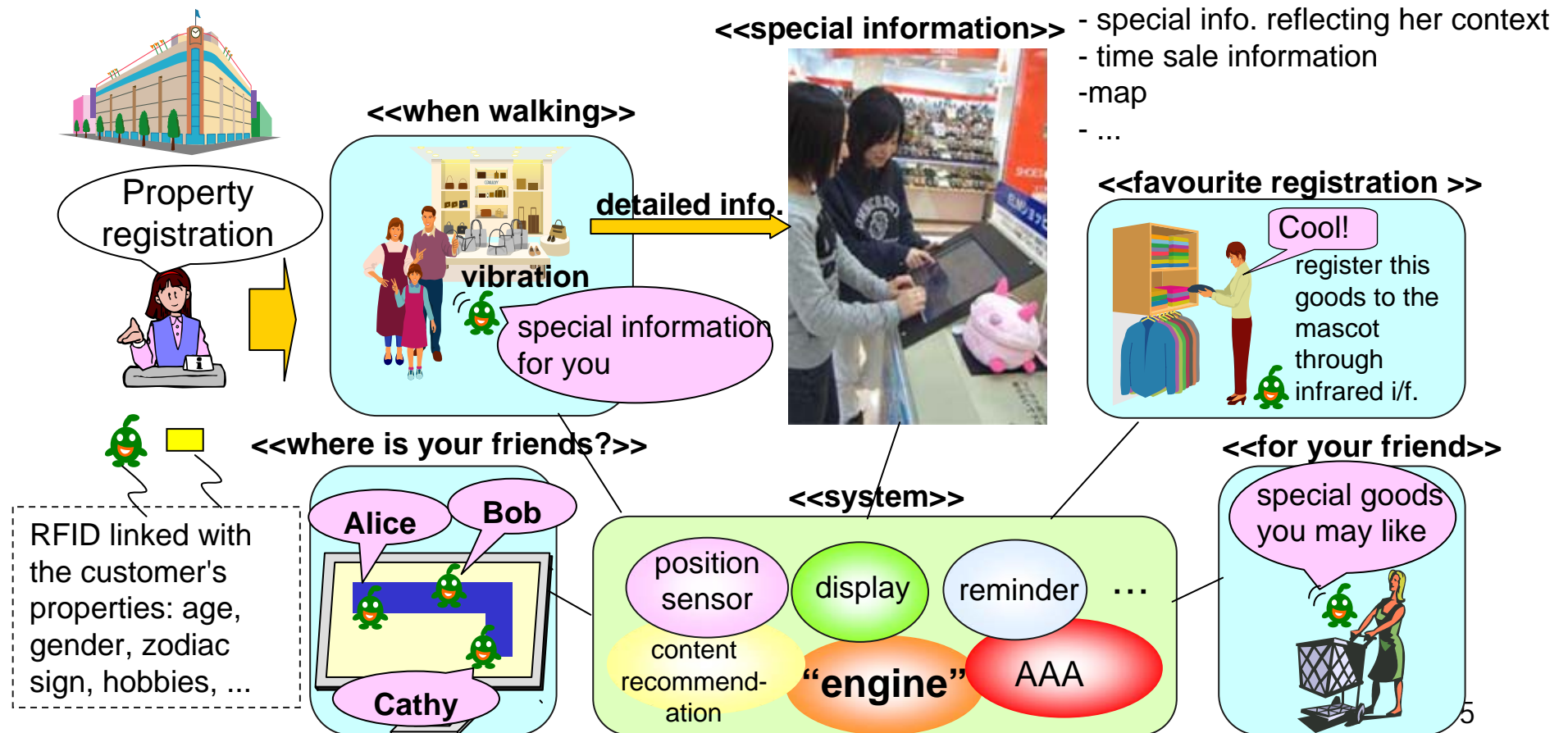
???

Very important. But we do not have any appropriate solution
No discovery nor selection facilities using the states.

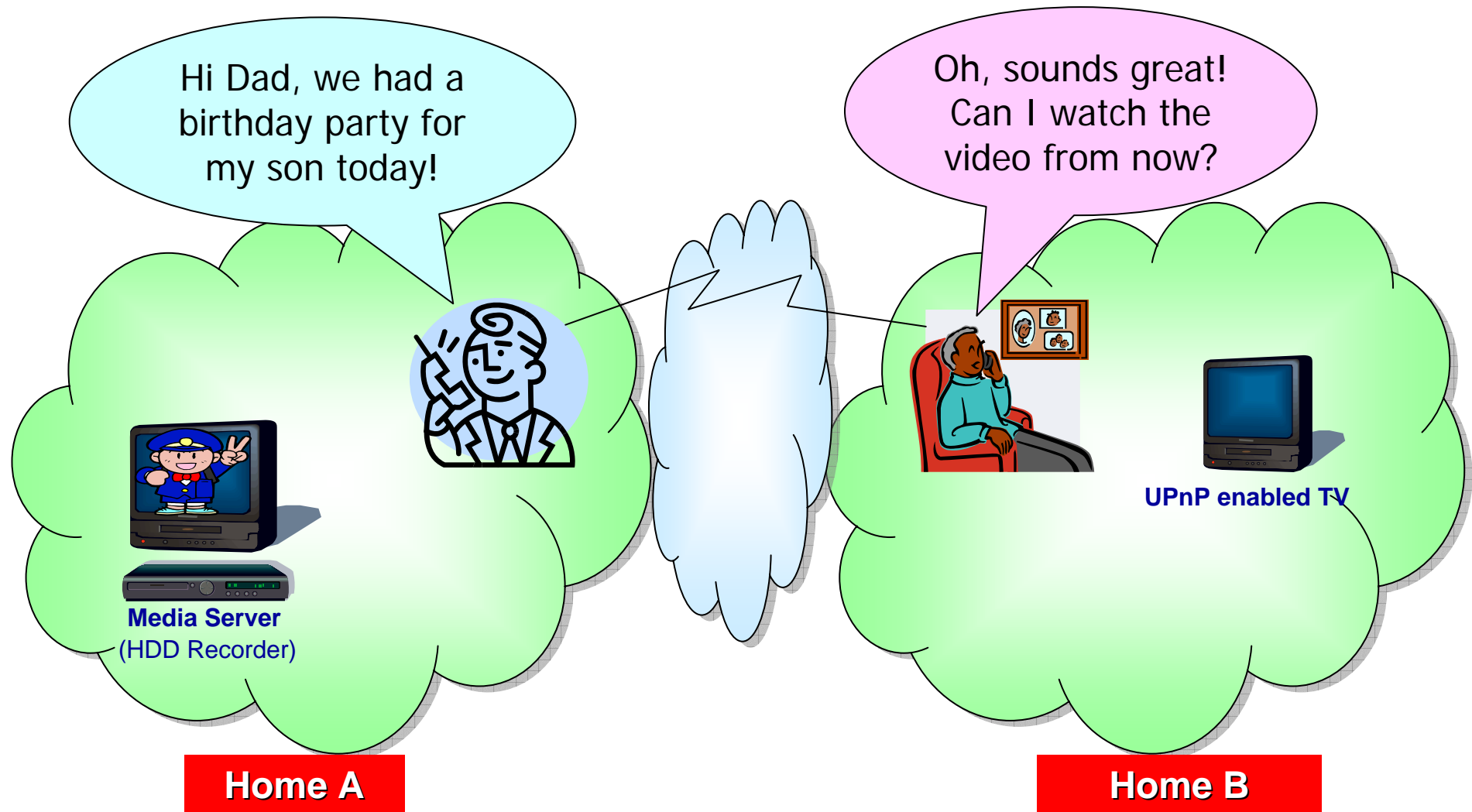
Component example: network resource management facility

Prototype 1: shopping support system (SI)

The **ubiquitous shopping support system** notifies customers of the existence of information appropriate for them in their current situations (**context**), such as time, place, their properties, and preferences in a shopping mall. A mascot with active/passive tags, radio i/f, and infrared i/f, vibrates when there is something interesting in a specific area in the mall.



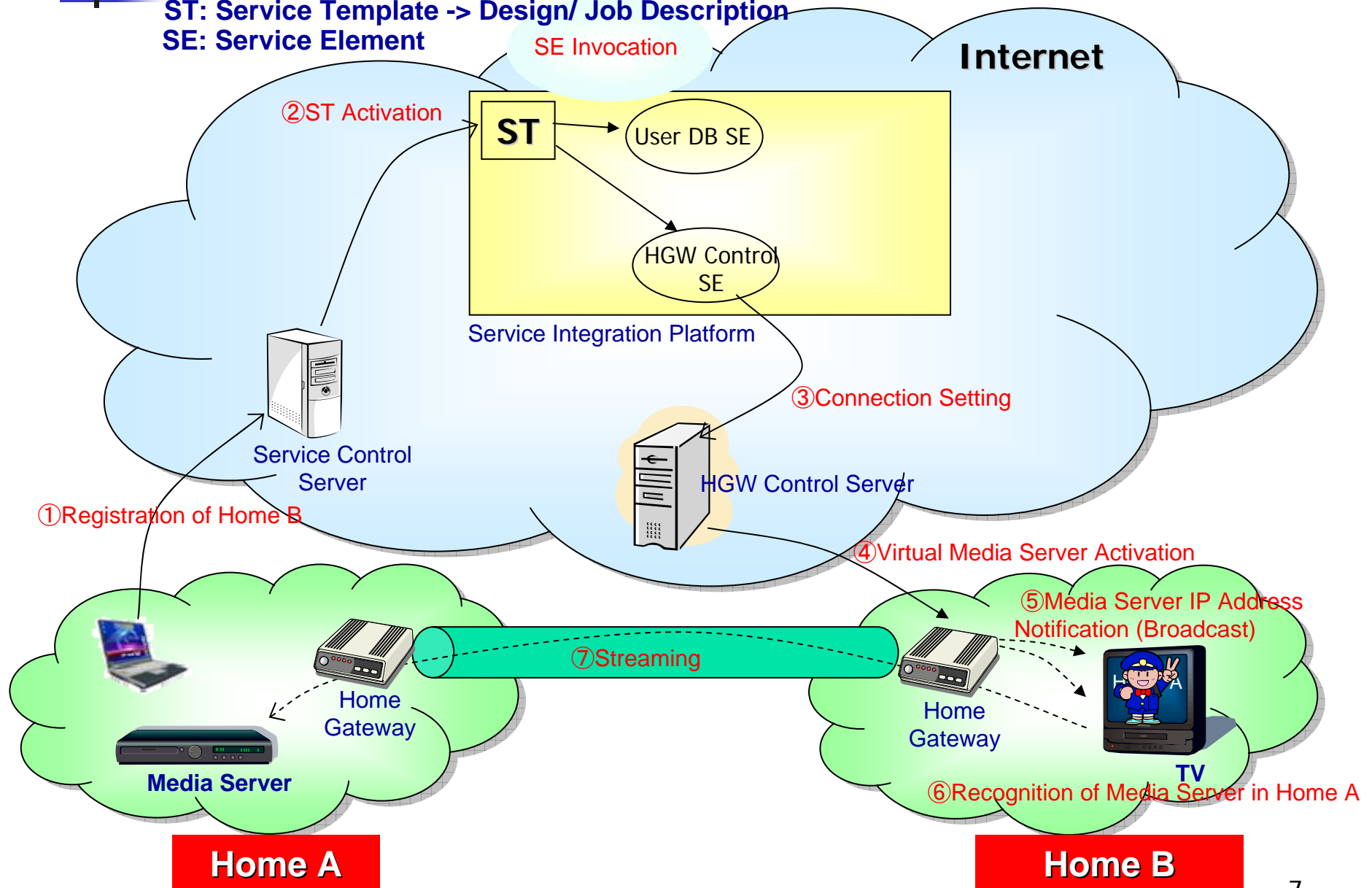
Prototype 2: Personal Home Video Sharing



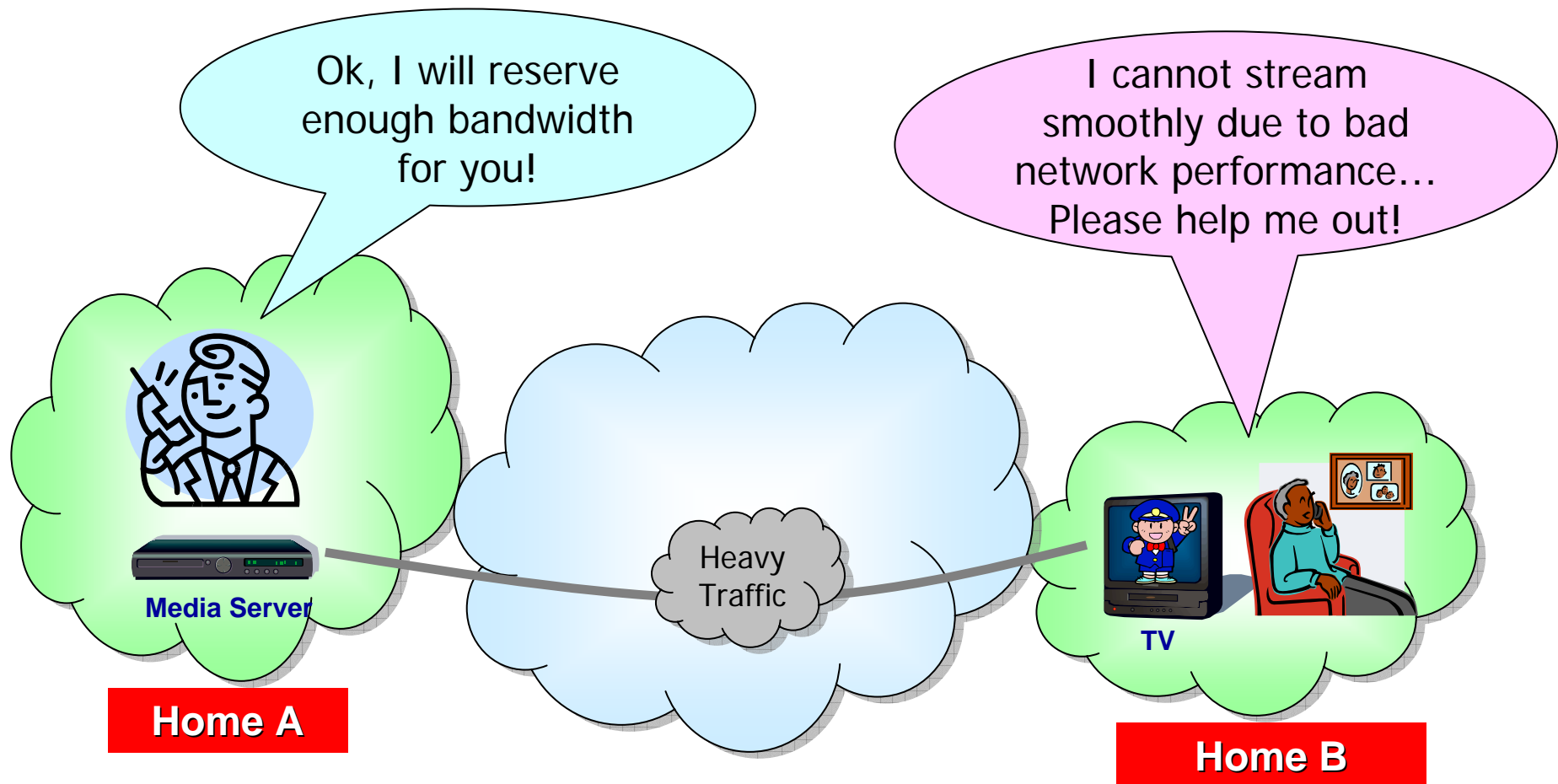
Video Sharing using UPnP Enabled Devices

ST: Service Template -> Design/ Job Description

SE: Service Element



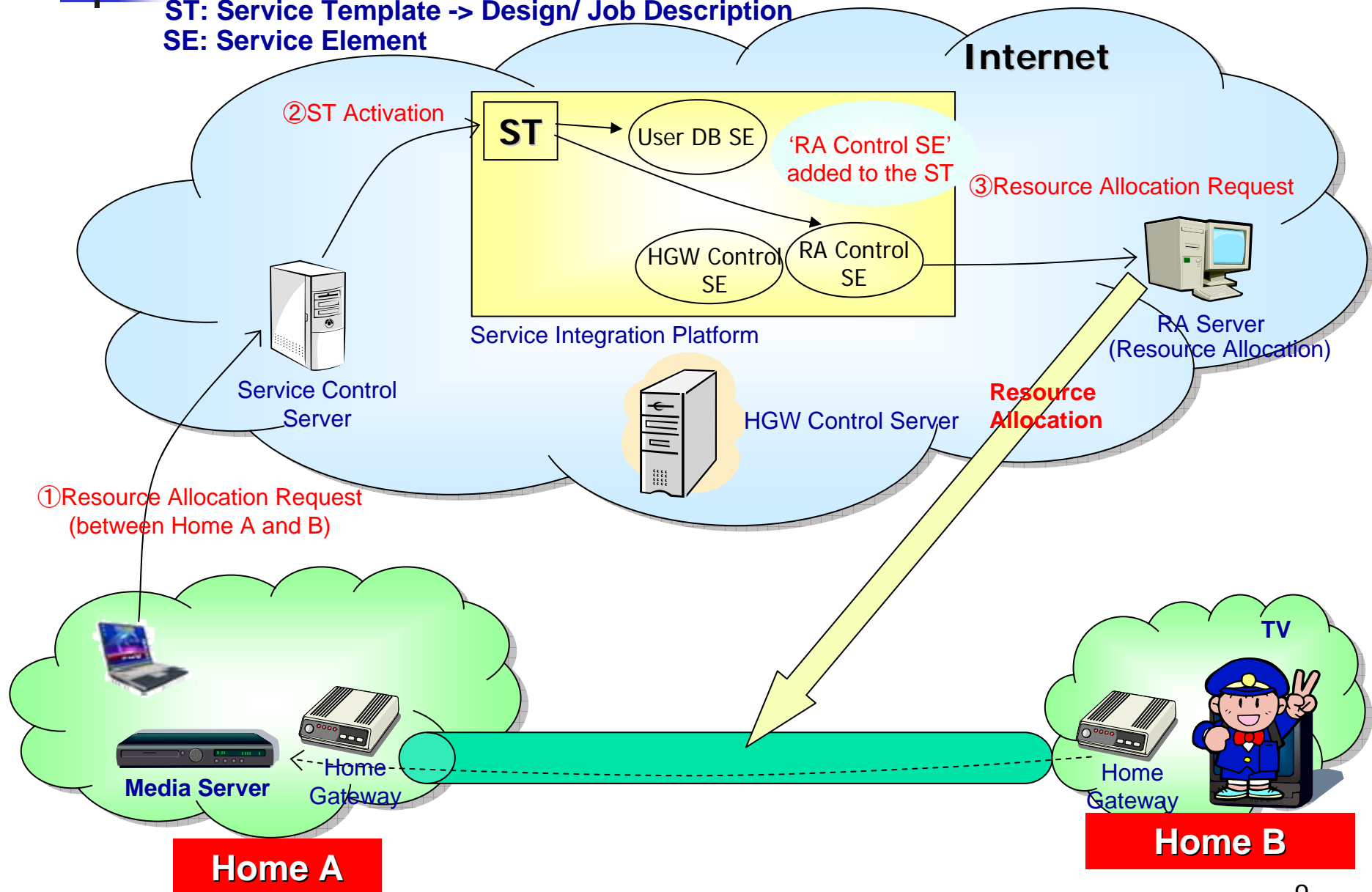
Video Sharing with Good Quality?

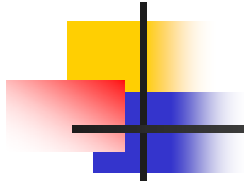


Video Sharing using Resource Allocation

ST: Service Template -> Design/ Job Description

SE: Service Element



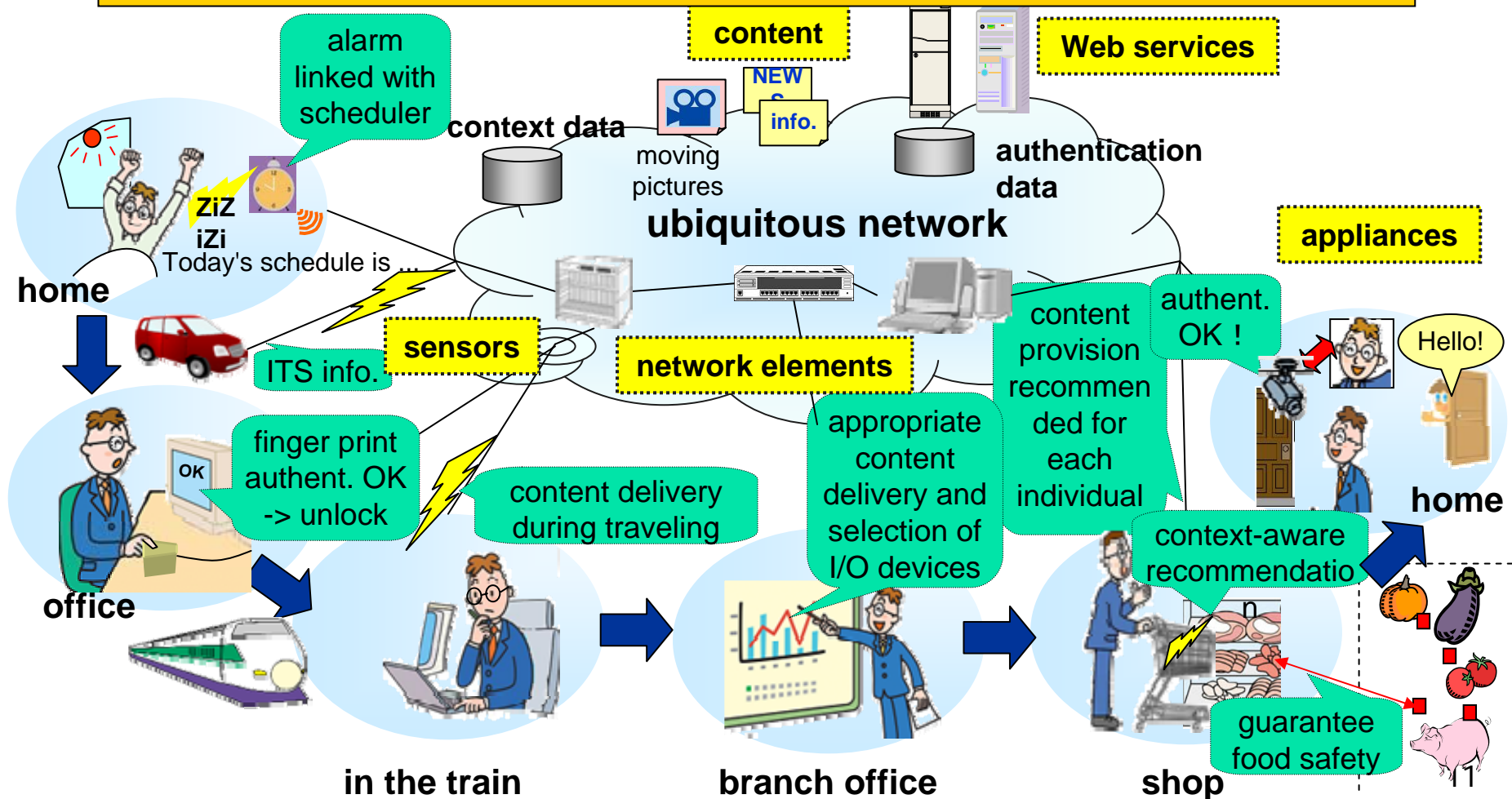


Thanks!

An Image of ubiquitous life

Services provided in daily lives will be automated and personalized by ubiquitous communication and computing technologies. Appropriate functions are provided anywhere and in any situation.

provide functions appropriate for the current time & situation



Adaptive meta-level service scenario

BPEL, often used for legacy system integration, must designate specific operations and formats. Before invocation, specific Web Services satisfying given conditions must be found through the UDDI registry.

```
<sequence>
```

```
  <invoke name="NTT-Kyoto-RentalCar" operation="reserveCar"/>
```

```
</sequence>
```

specific &
unique

In our method, rigid invoke/operation names are not used, but OWL-S based abstract description is used. From several candidates, the most appropriate one is selected considering the user situations, and the current one is switched over to another if user situations are changed.

```
<sequence>
```

```
  <search name="rentalcarSE" attributeName=travelDestination/>
```

```
  <invoke name="rentalcarSE" operation="reserveCar"/>
```

```
</sequence>
```

OWL-S based
description

abstract expression



Abstract description

【abstract scenario】

```
<sequence>
  <invoke name="user property SE" operation="address"/>
  <search name="transportation SE" attributeName="address"/>
    <copy from="home" to="destination"/>
    <copy from="user@location" to current position"/>
  <invoke name="transportation info SE" operation="obtain transportation URL"/>
  <search name="monitor SE" attributeName="user@location"/>
    <copy from="transportation URL" to="monitor display URL"/>
  <invoke name="monitor SE" operation=monitor display/>
</sequence/>
```

【BPEL scenario】

★train

```
<sequence>
  <invoke name="elmDB" operation="getUserAddress"/>
  <copy from="userAddress" to="deststation"/>
  <copy from="userCurrentPlace" = "startstation"/>
  <invoke name="YamatoLine" operation="getTimeURL"/>
  <copy from="timeURL" to="displayURL"/>
  <invoke name="elm1FMonitor" operation=displayInfo/>
</sequence/>
```

★bus

```
<sequence>
  <invoke name="elmDB" operation="getUserAddress"/>
  <copy from="userAddress" to="deststation"/>
  <copy from="userCurrentPlace" = "startstation"/>
  <invoke name="KonanBus" operation="busTimeURL"/>
  <copy from="bustimeURL" to="displayURL"/>
  <invoke name="elmGateMonitor" operation=displayInfo/>
</sequence/>
```

★flight

```
<sequence>
  <invoke name="elmDB" operation="getUserAddress"/>
  <copy from="userAddress" to="deststation"/>
  <copy from="userCurrentPlace" = "startstation"/>
  <invoke name="JAL" operation="flightTimeURL"/>
  <copy from="flighttimeURL" to="displayURL"/>
  <invoke name="elm2FMonitor" operation=displayInfo/>
</sequence/>
```