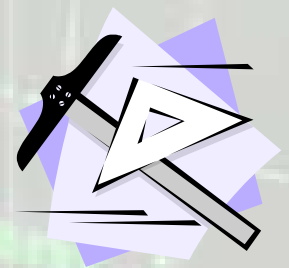
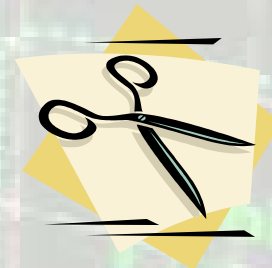


A Service Shaping Approach for Task-based Computing Middleware

Jin Nakazawa

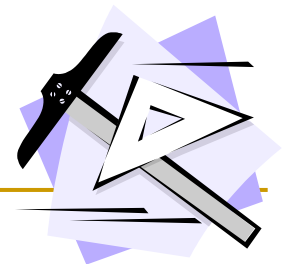
Graduate School of Media and Governance

Keio University, JAPAN



Perspective of This Research

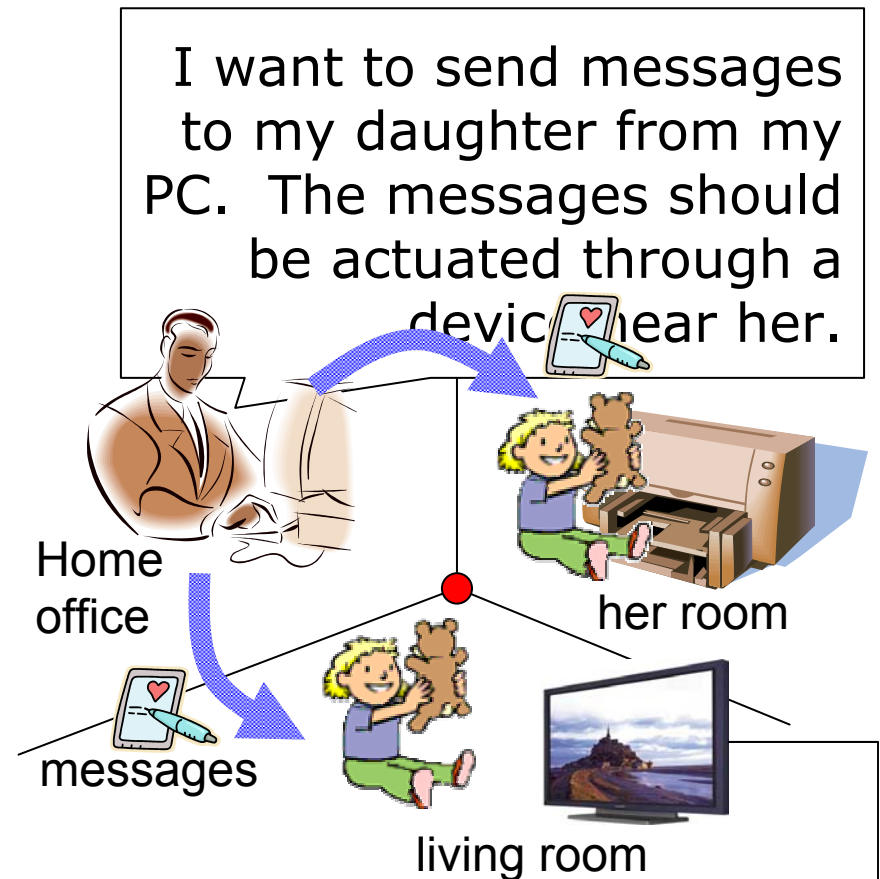
- Physical tools have particular shapes to satisfy their roles in our daily activities.
 - We infer how to use them and how they behave.
 - We can infer substitutive tools from their shapes.
 - Scissors and a knife are different tools, but substitutive to cut a paper since they have sharp edges in common.
- This research aims at representing such shapes for services to describe their substitutability.



Example of Service Substitution

- a TV substituting a printer (1)-

- Sending messages to whatever actuator near a user.
 - Usually, she is in her own room.
 - The messages are printed out.
 - When she moves to the living room, the messages are displayed on the TV.
- **Describing similarity of service shapes**
 - Aiming at describing abstract tasks based on such service substitutability.



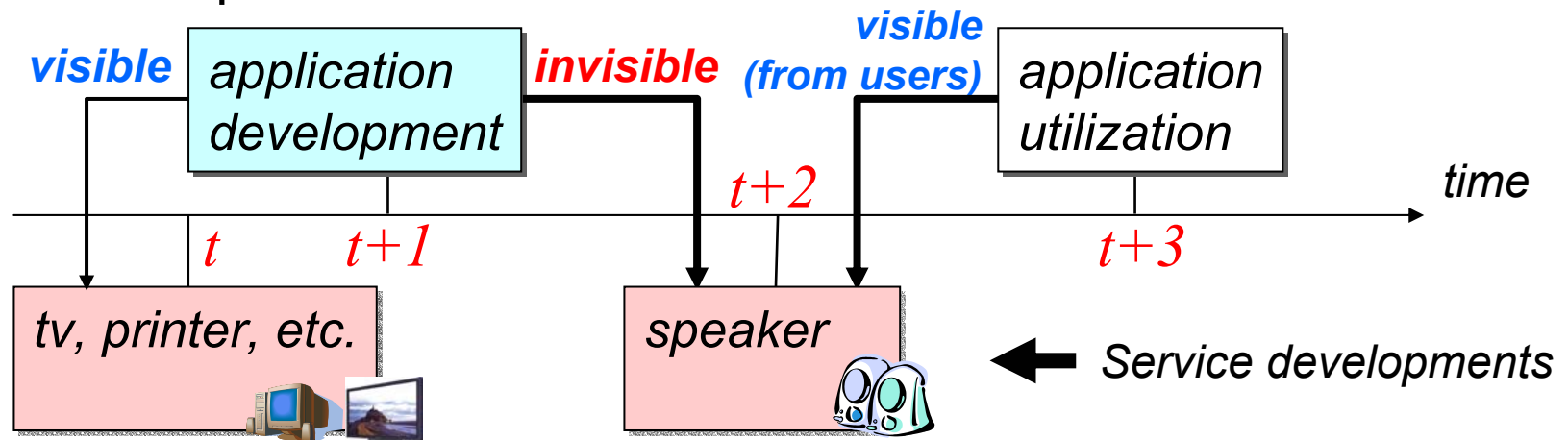
TV substituting the printer. ←

Outline

1. Overview of our service framework: Galaxy
 - services, applications, and service lookup
 2. Problem Description
 - Hidden services problem
 3. Approach
 - Service Shaping
 4. Implementation and Application
 - Current implementation and its performance
 5. Summary
-

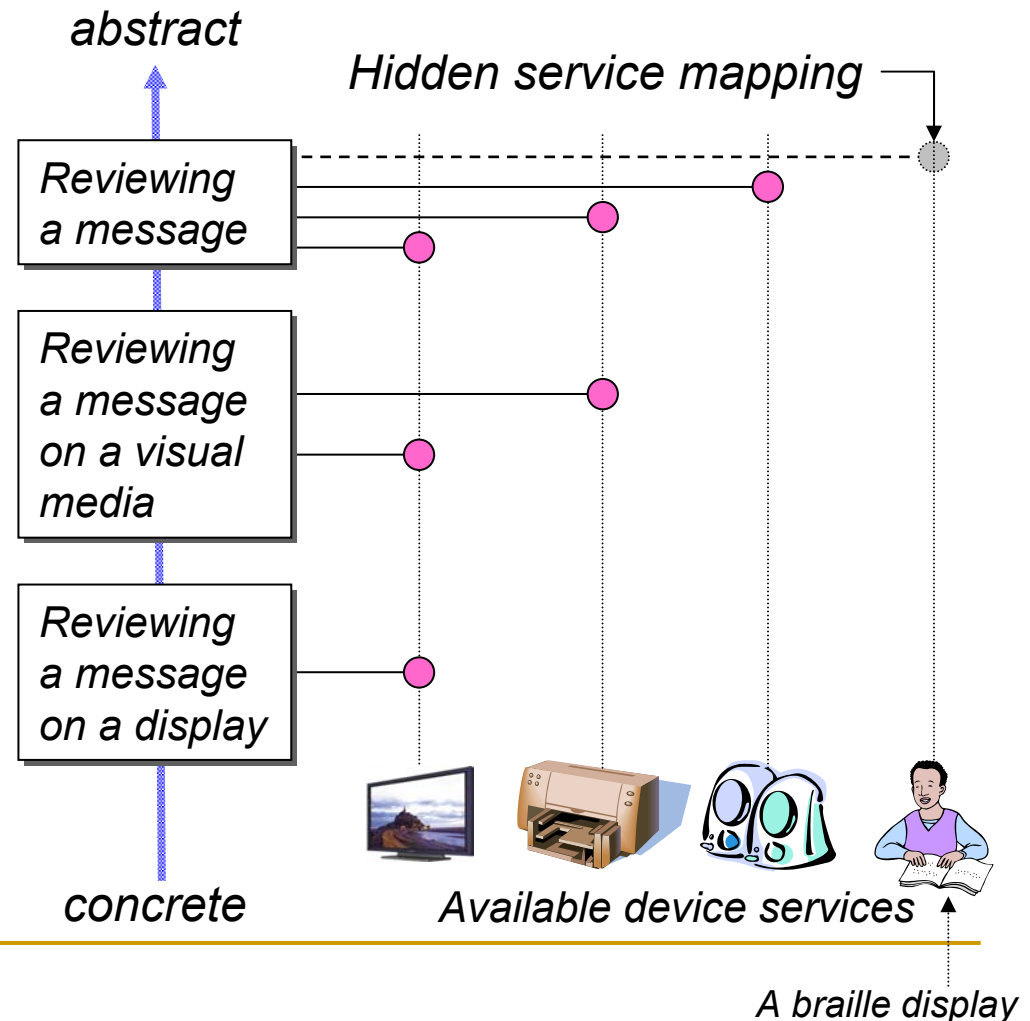
Hidden Services Problem in Type-based Service Management

- Time gap between application development and service deployment
 - Hides services which exist when utilizing an application, but doesn't when developing it.
 - Users should modify the application to utilize the newly developed services.



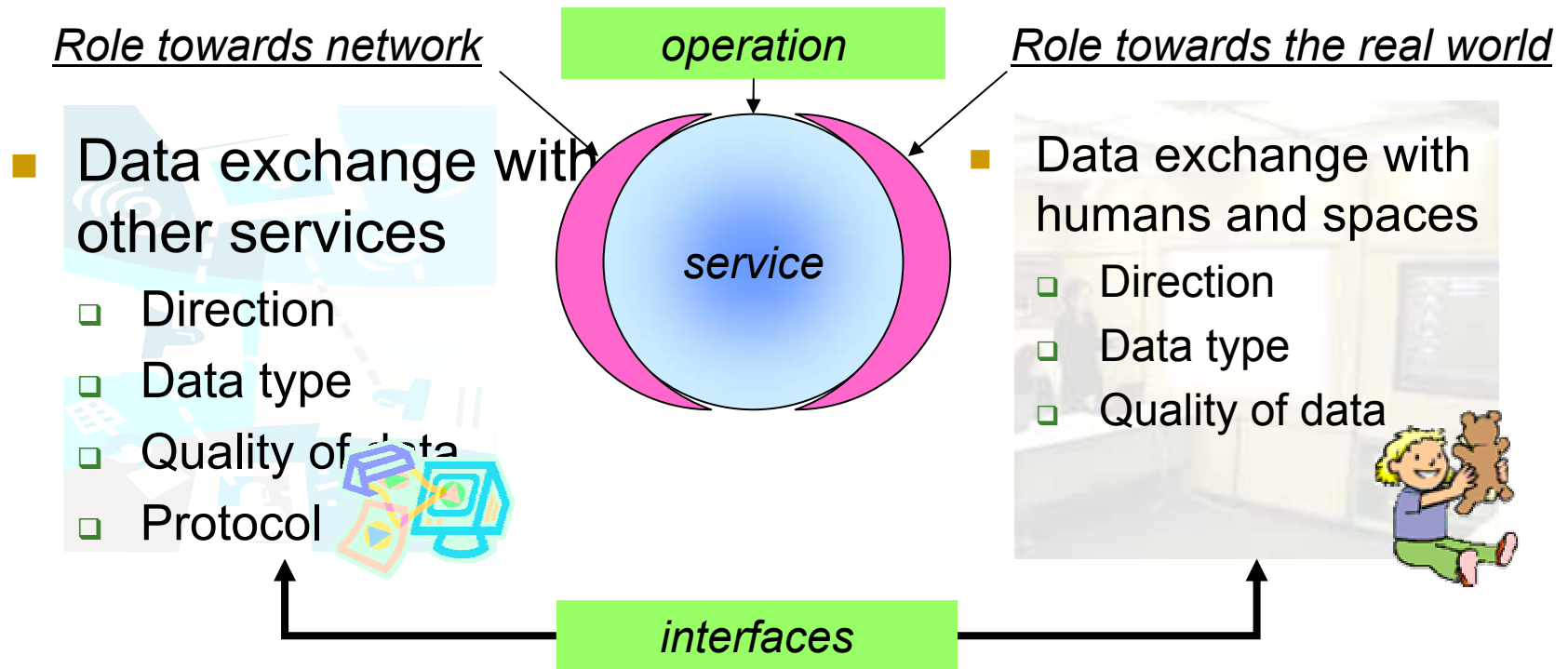
Hidden Services Problem in Type-based Service Management (2)

- Abstraction gaps between applications and services.
 - Hides a service that
 - an application programmer doesn't suppose to
 - but actually able to satisfy the requirement.
 - Users need to check services supposed in the application, and modify it if some services in the user's environment are not included in it.



Service Shaping

- Defines the role of a service explicitly with
 - Interface: capable input / output against the network / the real world
 - Operation: behavior against input from the network / the real world.



Basic Idea

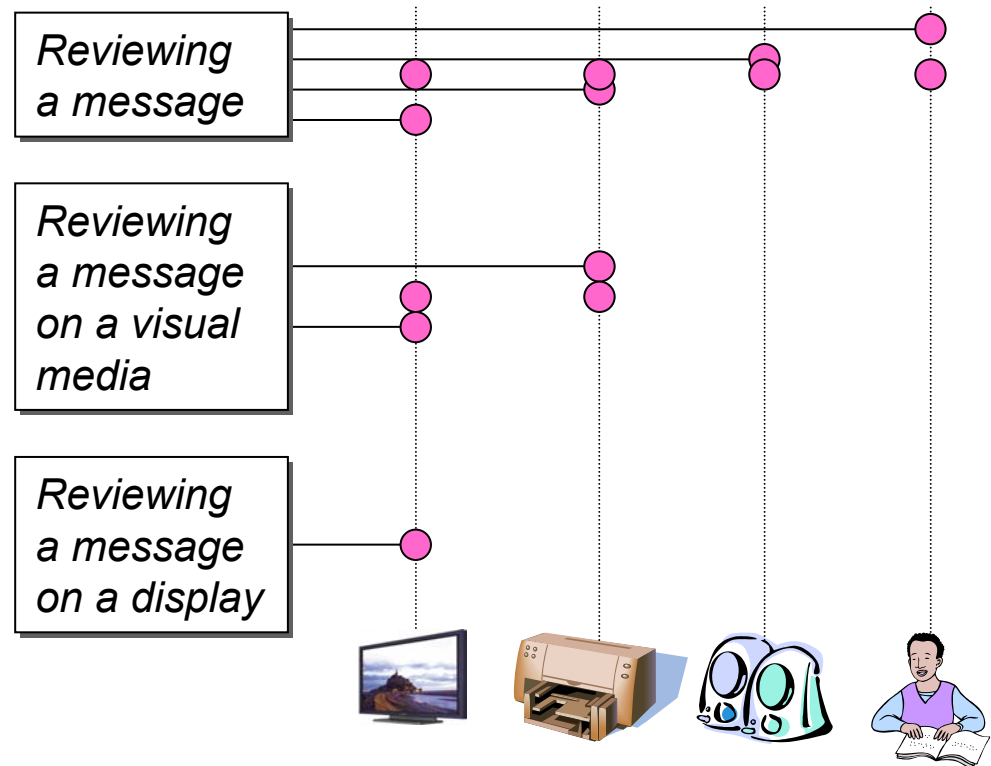
- Each application has a concern on service finding.

- It crosscuts multiple different services.

→ similar part of services

- Our approach

- Enabling applications to specify crosscutting concerns (aspects) with part of service shapes.
- Adopting open standards for decreasing programmer-specific definitions.



Role towards Networks

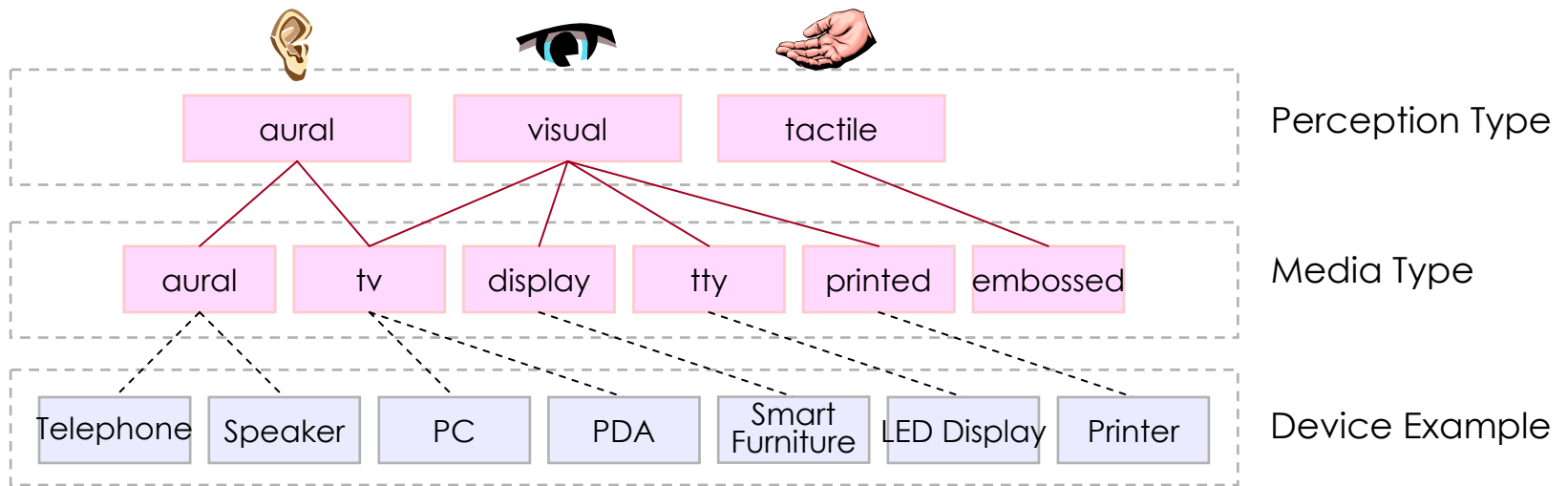
- How a service interacts with other services.
 - Direction : Whether the service sends or receives a data
 - Protocol : Standardized protocol names
 - E.g. RTP for audio and video, HTTP for various text formats, SOAP for controlling appliances
 - Data Type
 - Types are defined with **standardized** format names, i.e. **MIME types**, not with programmer defined types.
 - Media data: text/html, audio/raw, video/mpeg
 - Sensor data: gml, SensorML, ...
 - **Events: event/***
 - Quality of Data
 - Should also be defined with a standardized format, but still under investigation.
-

Role towards the Real World

- How a service interacts with humans and spaces.
 - Direction
 - Whether the service sends or receives a data
 - Data Type
 - Defined with extended form of a standard, which is adopted in CSS2(W3C)
 - Perception (CSS2: W3C)
 - aural, visual, tactile
 - Media Type (CSS2: W3C)
 - aural, braille, embossed, handheld, print, projection, screen, tty, tv
 - Service Specific types (CSS2 and CC/PP: W3C, RFC2534)
 - Display: size, color depth, refreshing rate...

Role towards the Real World (2)

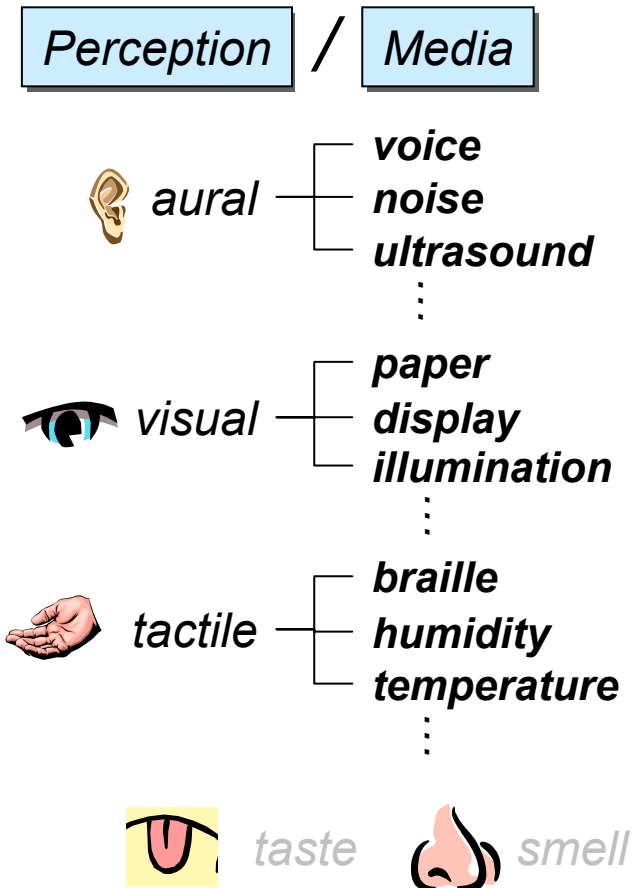
- The original media description in CSS2
 - Limited to describing actuator devices.
 - Able to be optimized more.



Role towards the Real World (3)

■ Summary of our extension

- Added sensible environmental information to represent sensors.
 - illumination
 - humidity
 - temperature, etc.
- Removed a redundant type
 - “aural/tv”
 - replaceable with “visual/display”+ “aural/aural”

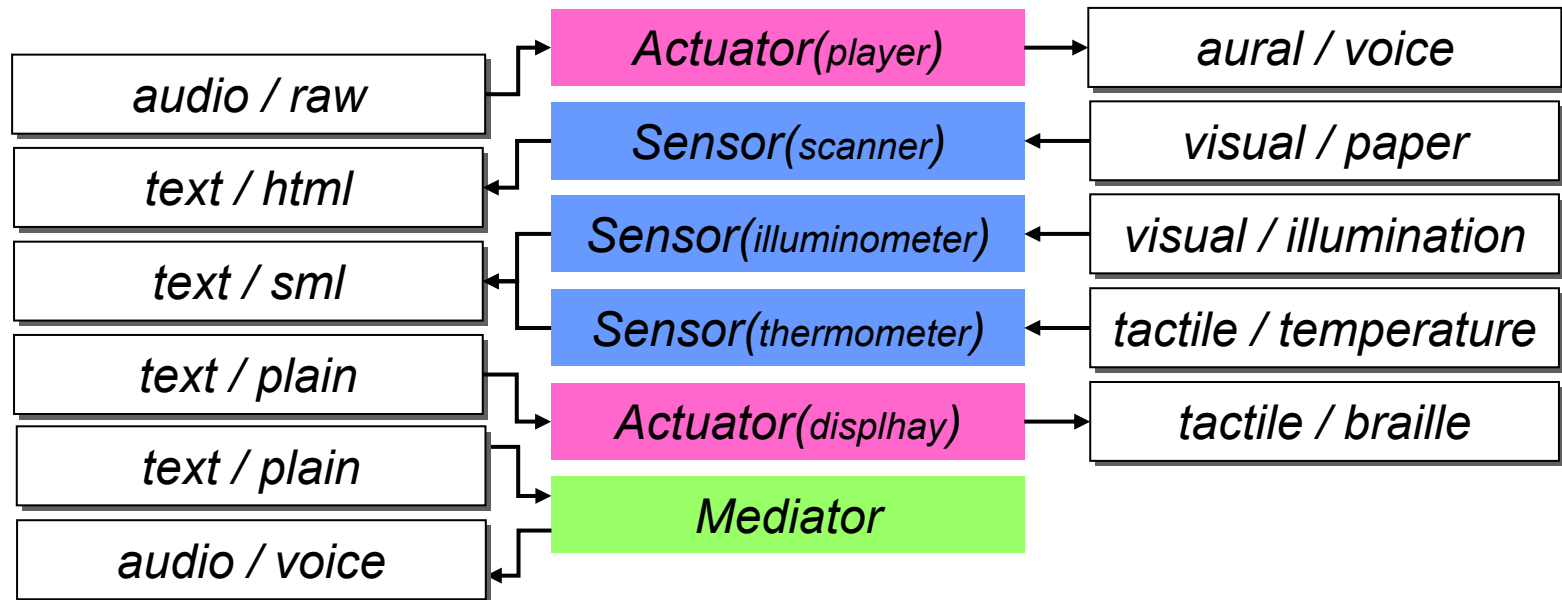


Internal Operation

- Represents a data flow between a network interaction and real world interaction.

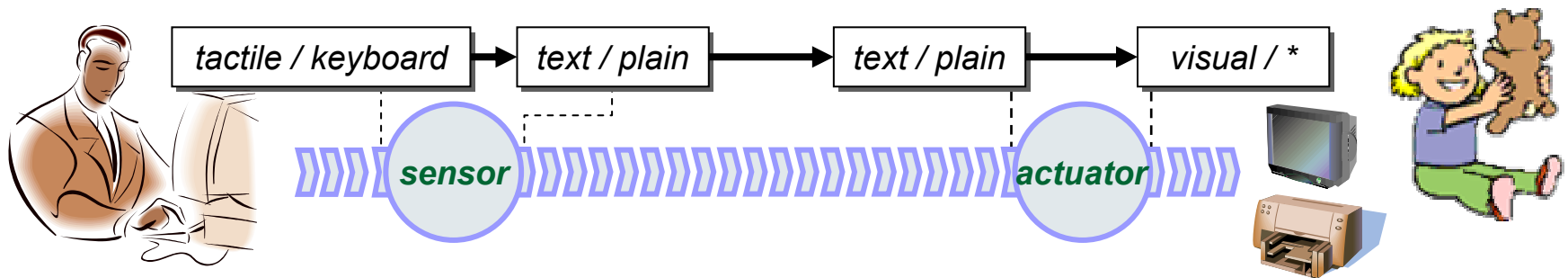
Role towards network

Role towards the real world



Service Shapes in the Scenario

- The scenario can be represented with one cooperation stream.
- The application code would be...



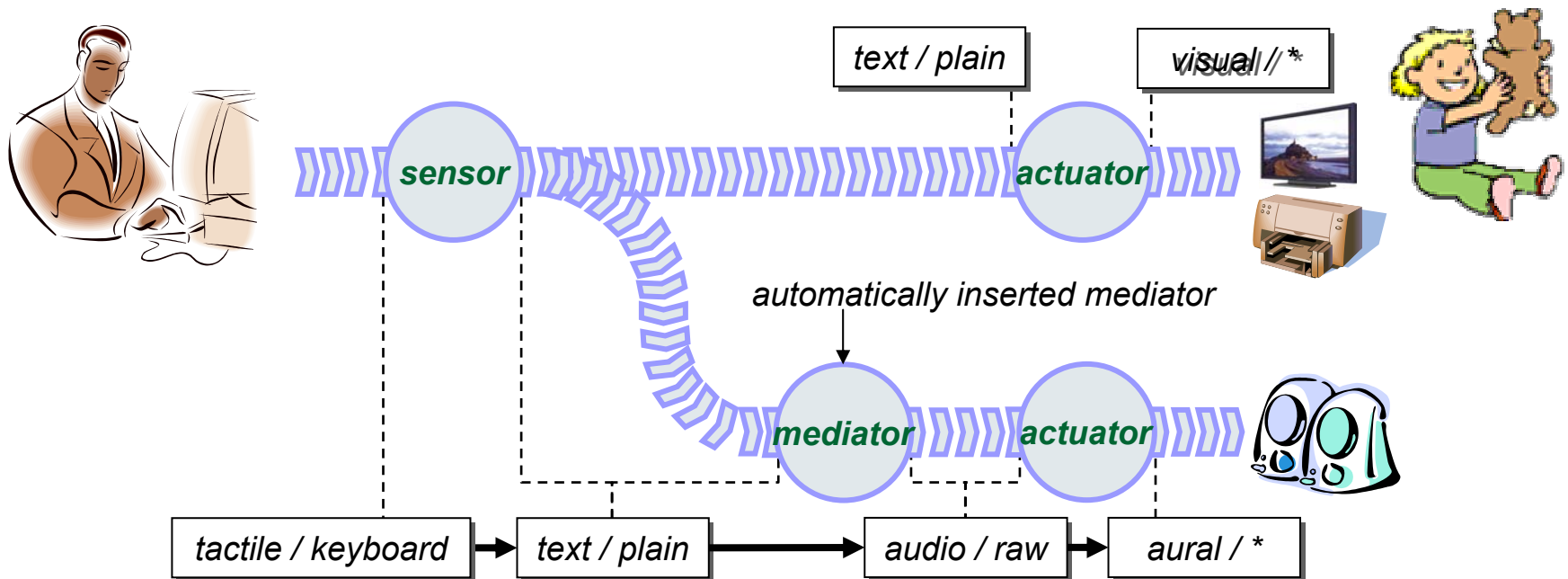
```
Attribute a1 = nearest to Jin;
Attribute a2 = nearest to her;
Source s = find( "tactile / keyboard", "*", a1);
Destination d = find( "*", "visual / *", a2);
connect(source, destination);
```

input from the real world the output to the network
input from the network the output to the real world

aspects

Service Shapes in the Scenario

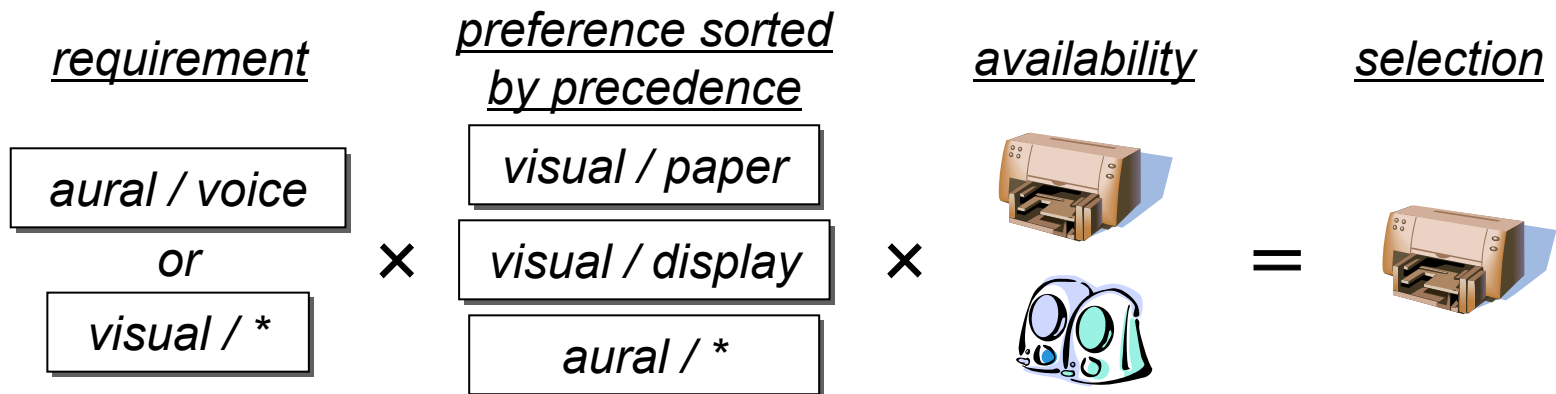
```
Attribute a1 = nearest to Jin;  
Attribute a2 = nearest to her;  
Source s = find( "tactile / keyboard", "*", a1);  
Destination d = find( { "visual / *", "aural / *" }, "*", a2);  
connect(source, destination);
```



Adaptive Service Selection

as an Application

- A circulated application cannot predict configuration of a particular environment.
 - Need for orthogonally evaluate the application's requirement, users' preferences, and service availability for an adaptive service selection.
- Substitutability of services reasoned with service shaping would enhance it.



Summary

- Proposed *service shaping* as a new approach for service modeling.
 - Modeled role of services towards a network and the real world
 - Enabled applications to specify their concerns on both services and human interaction.
 - Adopted open standards for description.
 - Increased possibility of service alternation with decreased programmer-specific definitions.
 - Future Task
 - Other form of shape description
-