Feature Interactions in Integrated Services of Networked Home Appliances
-An Object-Oriented Approach-

Presenter: Hiroshi Igaki

Masahide Nakamura, Hiroshi Igaki and Ken-ichi Matsumoto
Nara Institute of Science and Technology
Japan
Home Network System (HNS)

- Development of domestic network infrastructure technology
  - Zigbee, Bluetooth, X-10\textsuperscript{[18]}
- Networked Home Appliances
  - ECHONET\textsuperscript{[3]}, DLNA\textsuperscript{[2]}

HNS environment is widely developed
Examples of HNS application

- Health monitor
  - Monitor health status of home user from medical institution\(^{[17]}\)

- Remote control
  - Control home appliances from outside \(^{[11]}\)

- HNS Integrated Services
  - Orchestrate different home appliances\(^{[13]}\)

The HNS integrated service is known as a next-generation HNS application.
Examples of HNS integrated services

**Coming-Home Service**

1. When the user comes home, lights are turned on.
2. The lights are adjusted with using the illuminometer.

**DVD Theatre Service**

1. User switches on a DVD player.
2. TV and Speaker are turned on in DVD mode.
3. Lights are adjusted in DVD mode.
Feature Interactions on Home Network System

Feature Interaction between DVD Theater Service and Coming Home Service

**DVD Theater Service**
- When a user switches on the DVD player, the TV and the Speaker are turned on in DVD mode, the blind is closed, the brightness of the lights is minimized.

**Coming Home (Light) Service**
- When a user comes home, the light gets bright.
Objective and Approach

Objective
- **Formalize** feature interaction problems between the HNS integrated services

Approach
- **Define HNS model** (every appliance and HNS integrated service)
  - Model each appliance as an **object** consisting of **properties** and **methods**.
  - Model **environmental factor**.
Assumption on Appliance

- Each networked appliance should satisfy the following conditions.

- **Condition C1:**
  - Each appliance has a set of APIs to drive features.

- **Condition C2:**
  - The APIs can be executed by external software agents via network.
Each appliance has appliance status and its application interface.
- Regarded as an object consisting of properties and methods

Some properties are referred or updated from outside by method invocation
→ characterize each method as a pair of pre-condition and post-condition

\[
\begin{align*}
\text{Light.setBlightness}(\text{int degree}): & \quad \text{pre: } \text{Power} = \text{ON} \\
\text{Light.setPower}(\text{tPower mode}): & \quad \text{post: } \text{BrightnessSetting} = \text{degree} \\
& \quad \text{post: } \text{power} = \text{mode}
\end{align*}
\]

\text{Appliance} = (\text{appliance property, method, pre, post})
Environment Model

Appliance affects external environment (room brightness, etc..)
- Model environment property (sound volume, room temperature..) as global properties in HNS.

Environment properties are indirectly referred or updated by appliances
→ Whether methods reads or writes environment properties or not.

Light.setBlightness(int degree): **Environment Read**: nothing  **Environment Write**: Brightness
**Definition of HNS**

HNS = ( Appliances, Environment)

### Appliance property

<table>
<thead>
<tr>
<th>Appliance</th>
<th>Property</th>
<th>PropertyType</th>
<th>Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>AirConditioner</td>
<td>Power</td>
<td>{ON,OFF}</td>
<td>setPower()</td>
</tr>
<tr>
<td></td>
<td>TemperatureSetting</td>
<td>unsigned int (°C)</td>
<td>setTemperature()</td>
</tr>
<tr>
<td>Thermometer</td>
<td>Power</td>
<td>{ON,OFF}</td>
<td>setPower()</td>
</tr>
<tr>
<td></td>
<td>CurrentTemperature</td>
<td>unsigned int (°C)</td>
<td>getTemperature()</td>
</tr>
<tr>
<td>Speaker</td>
<td>Power</td>
<td>{ON,OFF}</td>
<td>setPower()</td>
</tr>
<tr>
<td></td>
<td>Input</td>
<td>{TV, DVD}</td>
<td>setInput()</td>
</tr>
<tr>
<td></td>
<td>Channel</td>
<td>{2, 5.1}</td>
<td>setChannel()</td>
</tr>
<tr>
<td></td>
<td>VolumeSetting</td>
<td>unsigned int (dB)</td>
<td>setVolume()</td>
</tr>
</tbody>
</table>

### Environment property

<table>
<thead>
<tr>
<th>Appliance</th>
<th>Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>AirConditioner</td>
<td>Temperature</td>
</tr>
<tr>
<td>Thermometer</td>
<td>Temperature</td>
</tr>
<tr>
<td>Speaker</td>
<td>Volume</td>
</tr>
</tbody>
</table>

### Appliance Method

<table>
<thead>
<tr>
<th>Appliance</th>
<th>Method</th>
<th>Pre-Condition</th>
<th>Post-Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>AirConditioner</td>
<td>setPower(tPower onoff)</td>
<td>Power=onoff</td>
<td></td>
</tr>
<tr>
<td></td>
<td>setTemperature(tTemperature temp)</td>
<td>Power='ON'</td>
<td>TemperatureSetting=temp</td>
</tr>
<tr>
<td>Thermometer</td>
<td>setPower(tPower onoff)</td>
<td>Power=onoff</td>
<td></td>
</tr>
<tr>
<td></td>
<td>getTemperature()</td>
<td>Power='ON' (\land) CurrentTemperature=*</td>
<td></td>
</tr>
<tr>
<td>Speaker</td>
<td>setPower(tPower onoff)</td>
<td>Power=onoff</td>
<td></td>
</tr>
<tr>
<td></td>
<td>setInput(tInput spInput)</td>
<td>Power='ON'</td>
<td>Input=spInput</td>
</tr>
<tr>
<td></td>
<td>setChannel(tChannel spChannel)</td>
<td>Power='ON'</td>
<td>Channel=spChannel</td>
</tr>
<tr>
<td></td>
<td>setVolume(tVolume spVolume)</td>
<td>Power='ON'</td>
<td>VolumeSetting=spVolume</td>
</tr>
</tbody>
</table>
HNS integrated service scenarios

- **SS 1: Auto-TV Service** - When the user turns on the TV, the speaker's is automatically adjusted for the TV mode.
- **SS 2: DVD Theater Service** - When a user switches on the DVD player, the TV, speaker is set in DVD mode, and the blind and the light make the room dark.
- **SS 3: Coming Home Light Service** - When the door (sensor) notices that the user comes home, the light is automatically turned on based on current room brightness (illuminometer).
- **SS 4: Coming Home Air Conditioning Service** - When the door notices that the user comes home, the air-conditioner is turned on based on current temperature (thermometer).
- **SS 5: Ringing and Mute Service** - When the telephone rings, the volume of the speaker is muted.
- **SS 6: Blind Service** - When sun light is available, the blind is opened.
- **SS 7: Sleep Service** - When the user goes to bed or goes outside, all appliances are turned off.
### Definition of service scenarios

Define HNS service scenarios as **sequence of methods (API calls)**

#### Auto-TV Service

Auto-TV Service - When the user turns on the TV, the speaker's is automatically adjusted for the TV mode.

<table>
<thead>
<tr>
<th>Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1.TV.setPower(ON)</td>
</tr>
<tr>
<td>1.2.TV.setInput(TV)</td>
</tr>
<tr>
<td>1.3.Speaker.setPower(ON)</td>
</tr>
<tr>
<td>1.4.Speaker.setInput(TV)</td>
</tr>
<tr>
<td>1.5.Speaker.setChannel(2)</td>
</tr>
<tr>
<td>1.6.Speaker.setVolume(60)</td>
</tr>
</tbody>
</table>

#### DVD Theatre Service

When a user switches on the DVD player, the TV, speaker is set in DVD mode, and the blind and the light make the room dark.

<table>
<thead>
<tr>
<th>Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1.DVD.setPower(ON)</td>
</tr>
<tr>
<td>2.2.TV.setPower(ON)</td>
</tr>
<tr>
<td>2.3.TV.setInput(DVD)</td>
</tr>
<tr>
<td>2.4.Blind.setPower(ON)</td>
</tr>
<tr>
<td>2.5.Blind.setGate(Close)</td>
</tr>
<tr>
<td>2.6.Light.setPower(ON)</td>
</tr>
<tr>
<td>2.7.Light.setBrightness(50)</td>
</tr>
<tr>
<td>2.8.Speaker.setPower(ON)</td>
</tr>
<tr>
<td>2.9.Speaker.setInput(DVD)</td>
</tr>
<tr>
<td>2.10.Speaker.setChannel(5.1)</td>
</tr>
<tr>
<td>2.11.Speaker.setVolume(80)</td>
</tr>
</tbody>
</table>

#### Coming Home Service

When the door (sensor) notices that the user comes home, the light is automatically turned on based on current room brightness (illuminometer).

<table>
<thead>
<tr>
<th>Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1.Door.getDoorStatus()</td>
</tr>
<tr>
<td>3.2.Illuminometer.setPower(ON)</td>
</tr>
<tr>
<td>3.3.Illuminometer.getBrightness()</td>
</tr>
<tr>
<td>3.4.Light.setPower(ON)</td>
</tr>
<tr>
<td>3.5.Light.setBrightness(600)</td>
</tr>
</tbody>
</table>
Formalization of interaction

A pair of services s1 and s2 interact. <=>
A pair of methods from s1 and s2 conflict.

- Appliance Interaction
  - Direct conflict among methods on the same appliance
- Environment Interaction
  - Indirect conflict among appliances via the HNS environment
Appliance Interaction

Two methods from different services try to access the same appliance property on the same appliance.

SS1: Auto-TV Service
1.1. TV.setPower(ON)
1.2. TV.setInput(TV)
1.3. Speaker.setPower(ON)
1.4. Speaker.setInput(TV)
1.5. Speaker.setChannel(2)
1.6. Speaker.setVolume(60)

SS2: DVD Theater Service
2.1. DVD.setPower(ON)
2.2. TV.setPower(ON)
2.3. TV.setInput(DVD)
2.4. Blind.setPower(ON)
2.5. Blind.setGate(Close)
2.6. Light.setPower(ON)
2.7. Light.setBrightness(50)
2.8. Speaker.setPower(ON)
2.9. Speaker.setInput(DVD)
2.10. Speaker.setChannel(5.1)
2.11. Speaker.setVolume(80)
Environment Interaction

Two methods from different services try to access different appliance, which results in a conflict on an environment property.

1. Door.getDoorStatus()
2. Thermometer.setPower(ON)
3. Thermometer.getTemperature()
4. AC.setPower(ON)
5. AC.setTemperature(26)
6. Blind.setPower(ON)
7. Blind.setGate(Open)

SS4: Coming Home Service (Temperature)
SS6: Auto-Blind Service

Environment Interaction
Case Study

Detect interactions between the methods of SS1~SS7

Input

- 7 HNS integrated service scenarios
  - Sequence of appliance methods

- 10 appliances
  - Appliance property, Appliance method, pre-condition, post-condition

- 3 environment properties
  - Environment property, R/W function

Output

- All possible pairs of appliance methods that cause appliance or environment interactions
## Case study: Appliance Interaction

43 appliance interactions

<table>
<thead>
<tr>
<th></th>
<th>SS1</th>
<th>SS2</th>
<th>SS3</th>
<th>SS4</th>
<th>SS5</th>
<th>SS6</th>
<th>SS7</th>
</tr>
</thead>
<tbody>
<tr>
<td>SS1</td>
<td>(1.2,2.3) (1.4,2.9) (1.5,2.10) (1.6,2.11)</td>
<td>(1.6,5.3)</td>
<td>(1.6,5.3)</td>
<td>(1.1,7.2) (1.2,7.2) (1.3,7.4) (1.4,7.4) (1.5,7.4) (1.6,7.3) (1.6,7.4)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SS2</td>
<td>(2.7,3.5)</td>
<td>(2.11,5.3) (2.5,6.2)</td>
<td>(2.1,7.1) (2.2,7.2) (2.3,7.2) (2.4,7.9) (2.5,7.8) (2.5,7.9) (2.6,7.7) (2.7,7.6) (2.7,7.7) (2.8,7.4) (2.9,7.4) (2.10,7.4) (2.11,7.3) (2.11,7.4)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SS3</td>
<td></td>
<td></td>
<td>(3.2,7.5) (3.3,7.5) (3.4,7.7) (3.5,7.6) (3.5,7.7)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SS4</td>
<td></td>
<td></td>
<td></td>
<td>(4.2,7.9) (4.3,7.9) (4.4,7.8) (4.5,7.8)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SS5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(5.3,7.3) (5.3,7.4)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SS6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(6.1,7.11) (6.2,7.10) (6.2,7.11)</td>
<td></td>
</tr>
<tr>
<td>SS7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### SS1 vs SS2

- (1.2,2.3)
- (1.4,2.9)
- (1.5,2.10)
- (1.6,2.11)
Case study: Appliance Interaction

43 appliance interactions

SS1 vs SS2
(1.2,2.3)(1.4,2.9)
(1.5,2.10)(1.6,2.11)

SS1
Auto-TV Service

1.1.TV.setPower(ON)
1.2.TV.setInput(TV)
1.3.Speaker.setPower(ON)
1.4.Speaker.setInput(TV)
1.5.Speaker.setChannel(2)
1.6.Speaker.setVolume(60)

SS2
DVD Theater

2.1.DVD.setPower(ON)
2.2.TV.setPower(ON)
2.3.TV.setInput(DVD)
2.4.Blind.setPower(ON)
2.5.Blind.setGate(Close)
2.6.Light.setPower(ON)
2.7.Light.setBrightness(50)
2.8.Speaker.setPower(ON)
2.9.Speaker.setInput(DVD)
2.10.Speaker.setChannel(5.1)
2.11.Speaker.setVolume(80)
Case Study:
Environment interaction

24 interactions detected

<table>
<thead>
<tr>
<th></th>
<th>SS1</th>
<th>SS2</th>
<th>SS3</th>
<th>SS4</th>
<th>SS5</th>
<th>SS6</th>
<th>SS7</th>
</tr>
</thead>
<tbody>
<tr>
<td>SS1</td>
<td></td>
<td></td>
<td></td>
<td>(1.6,5.1)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SS2</td>
<td>(2.5,3.3)(2.5,3.5)(2.7,3.3)</td>
<td>(2.5,4.3)(2.5,4.5)</td>
<td>(2.11,5.1)(2.11,5.2)</td>
<td>(2.7,6.2)</td>
<td>(2.5,7.6)(2.7,7.10)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SS3</td>
<td></td>
<td></td>
<td></td>
<td>(3.3,6.2)(3.5,6.2)</td>
<td>(3.3,7.6)(3.3,7.10)(3.5,7.10)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SS4</td>
<td></td>
<td></td>
<td></td>
<td>(4.3,6.2)(4.5,6.2)</td>
<td>(4.3,7.10)(4.5,7.10)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SS5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(5.1,7.3)(5.2,7.3)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

SS2: DVD Theater Service

- Speaker Object: Volume Setting 80
- AC Object: Temp Setting 26
- Thermometer Object: Current Temp 10

SS4: Coming Home (Air Conditioner) Service

- Speaker Object: Volume 4.5
- AC Object: Temp 4.3
- Blind Object: Close

SS2 vs SS4
(2.5,4.3)(2.5,4.5)
Summary

- We have formalized the HNS in an object-oriented fashion.
- We have defined two kinds interactions.
  - Appliance interaction, Environment interaction
- We conducted a case study of off-line interaction detection.
  - 7 integrated services, 10 appliances.
  - 43 appliance and 24 environment interactions found.
**Related Work**

**Resource-centric** approach by Kolberg et. al[9]

- Captures each appliance as a *shared/non-shared resource*.
- Characterizes FI as incompatible access to a resource.
- Easy FI detection/handling, but difficult to capture slight differences on services.

Ours is **service-centric**

- Allows fine-grained treatment of FIs, but complex FI handling mechanism is necessary.
Future Work

- We described only offline interaction detection
  - FI detection method based service scenarios
- Results of the detection enable us to rebuild the service scenarios so that any interaction is avoided
  - limits flexible creation and deployment of the service scenarios
- Runtime interaction detection method is useful
Thank you