Fault Tolerance Support of Smart-M3 Application on the Software Infrastructure Level

Ivan Galov, Dmitry Korzun

Petrozavodsk State University
Department of Computer Science

Supported by project # 1481 from the basic part of state research assignment # 2014/154 of the Ministry of Education and Science of the Russian Federation, project # 14-07-00252 of the Russian Fund for Basic Research, project 14.574.21.0060 (RFMEFI57414X0060) of Federal Target Program “Research and development on priority directions of scientific-technological complex of Russia for 2014–2020”.

16th FRUCT Conference
October 30, 2014, Oulu, Finland
Smart-M3 platform

- Semantic information brokers (SIBs) maintain smart space content in low-level RDF triples
- Application consists of several knowledge processors (KPs) running on various devices
- An agent sharing ad-hoc knowledge across numerous domains
  - join, leave
  - insert, update, remove
  - (un)subscribe
- Interaction between KPs is implemented using publish/subscribe mechanism
Software Infrastructure of Smart-M3 Applications

Software infrastructure: the means for application operation

Infrastructural KPs are responsible for service construction and delivery

Infrastructural KP deployment options:

1. Clustering near the SIB
2. Device-aware location
3. Server-oriented location (non-SIB)
Example: SmartRoom system

- Holding conferences, meetings, and lectures
- Personalized interaction with room participant
- Participating using mobile devices

SIB is deployed either locally or on a remote machine
- Conference-service: conference runtime management
- Presentation-service: slide show of current speaker
- Agenda-service: visual activity agenda
- Content-service: storage of participants’ materials
Application Dependability

Dependable application: “reliance can justifiably be placed on the service it delivers” (M.R. Lui et al., 1996)

- Availability: readiness for usage.
- Reliability: continuity of service.
- Safety: nonoccurrence of catastrophic consequences on the environment.
- Confidentiality: nonoccurrence of unauthorized access.
- Integrity: nonoccurrence of improper alterations of information.
- Maintainability: ability to undergo repairs and evolutions.

Fault tolerance: application is capable to deliver its services in the presence of faults
Smart-M3 Application Dependability

Lacks of reliability and integrity
Subscription is the most failure-sensitive operation

Failure reasons:

1. SIB: software error (freezing, crashing); lost subscription connection with KPs; data loss (overload)

2. Wireless network: subscription connection breaks; data packets loss

3. Infrastructural KP: lost network connection with the SIB; software error (crashing)
Solutions

1. Content-service: persistent storage of volumetric data
   - presentations
   - images
   - audio
   - video

2. Mechanisms for fault tolerance of subscription
   - Subscription control mechanism: active regular checks for subscribed data
   - Restart/reconnection: KP reestablishes network connection with the smart space
Subscription Control

- Failure situations:
  - subscription indication was lost
  - subscription connection was disrupted

- Proactive check (subscription) is augmented with active checking

- If subscribed data are changed — failure has occurred

- Challenges:
  - how to detect information changes
  - no clear criterion on identifying breaks in subscription
  - check intervals

- Usage notification model: subscribe on data about changes, not on changing data
  I. Galov and D. Korzun, “Notification model for Smart-M3 applications”, NEW2AN/ruSMART 2014, LNCS 8638

- Not implemented yet: also crucial for mobile clients (A. Vdovenko)
Restart/Reconnection

- Restart: application is shut down and launched again
- Reconnection: application is still running but connection to SIB is re-established

Usage in SmartRoom:

<table>
<thead>
<tr>
<th>Infrastructure element</th>
<th>Restart</th>
<th>Reconnection</th>
</tr>
</thead>
<tbody>
<tr>
<td>SIB</td>
<td>+ (auto)</td>
<td>–</td>
</tr>
<tr>
<td>Conference-service</td>
<td>+ (auto)</td>
<td>–</td>
</tr>
<tr>
<td>Agenda-service</td>
<td>+ (manual)</td>
<td>+ (manual)</td>
</tr>
<tr>
<td>Presentation-service</td>
<td>+ (manual)</td>
<td>+ (manual)</td>
</tr>
<tr>
<td>Content-service</td>
<td>+ (manual)</td>
<td>–</td>
</tr>
</tbody>
</table>

Restart implementation: launching on the server computer with help of Ubuntu Upstart init daemon (configuration files):

```
redsbid → sib-tcp → conference-service
```
Mechanisms Deployment

Mechanisms:
SC – subscription control
RS – restart
RC – reconnect

Infrastructure of Smart-M3 application

Content-service

computer network
Performance Evaluation: Content-service

Server computer: Ubuntu Linux, Intel Xeon 2.30GHz, 4GB RAM

Content-service file (10M) upload time on the web server:

- Apache: 1.814 sec. (standard deviation is 0.014)
- nginx: 1.842 sec. (standard deviation is 0.030)
Performance Evaluation: Restart/Reconnection

- Server computer: Ubuntu Linux, Intel Xeon 2.30GHz, 4GB RAM (SIB, Conference-service, Content-service)
- Personal computer: Windows, Intel Core 2 Quad 2.40GHz, 8GB RAM (Presentation-service, Agenda-service)
- Time between service terminating and starting (until readiness) was measured

<table>
<thead>
<tr>
<th>Infrastructure element</th>
<th>Restart (sec)</th>
<th>Reconnection (sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SIB</td>
<td>1.045</td>
<td>—</td>
</tr>
<tr>
<td>Conference-service</td>
<td>2.023</td>
<td>—</td>
</tr>
<tr>
<td>Agenda-service</td>
<td>1.350</td>
<td>0.134</td>
</tr>
<tr>
<td>Presentation-service</td>
<td>0.230</td>
<td>0.075</td>
</tr>
<tr>
<td>Content-service</td>
<td>2.193</td>
<td>—</td>
</tr>
</tbody>
</table>
Conclusion

- Software infrastructure for the case of Smart-M3 applications
- Solutions for Smart-M3 applications fault tolerance:
  - Content-service to control volumetric factual data
  - Mechanisms for subscription control (fault detection) and recovery (when fault has happened)
- Experimental evaluation for SmartRoom system

Thank you for attention

E-mail: galov@cs.karelia.ru