Conrad Thiede, Christian Tominski, Heidrun Schumann
University of Rostock, Germany
IV09, Barcelona, July 16, 2009

Service-Oriented Information Visualization for Smart Environments

Smart Environments

- **Standard environments:**
  - Interconnected computing and output devices
  - No proactive support of users
- **Smart environments:**
  - Additional sensor devices to monitor environment and its inhabitants
  - Additional software components to analyze monitoring data and to predict future user actions
  - Proactive support by adaptation to the current situation

Smart Meeting Room

Specific focus of our research:

Smart Meeting Room

- Multiple stationary devices
  - Computing: servers, desktop PCs
  - Output: projectors, canvases, panel displays
  - Environmental: lights, blinds, air conditioning, …
  - Sensory: cameras, motion sensors, …
- Users can bring mobile devices
  - Notebooks, PDAs, projectors, …

Ad-hoc character of environment is core challenge

Smart Meeting Room

The Smart Lab at University of Rostock

Smart Meeting Room & InfoVis

- Smart meeting room and information visualization are connected in two ways

1. InfoVis can help in exploring the data generated by sensory devices and prediction software
2. Smart meeting room can serve as an environment for InfoVis applications

Motivation

- **Goal:** Utilize device ensemble, communication infrastructure, and predictive software components of smart meeting room to drive information visualization
- **Possible scenarios:**
  - Use one canvas for overview, another canvas for detail
  - Overview via projector and details on users' displays
  - Use multiple canvases for comparison tasks
  - Proactively adapt visualization to predicted user intentions

Motivation

- **Challenge:** Adapt the visualization process to the requirements and constraints of dynamic ad-hoc character of smart meeting room
- **Problem:** Classic InfoVis approaches target single machine with static output device only

- **Technical basis needed to tackle challenge:** Loosely coupled software components that generate visualizations in a joint effort

Technical Basis

- Two alternatives for InfoVis in smart env.:
  - Agent-based models (ABM)
    - Agents: operate autonomously to accomplish tasks
    - No higher-level control mechanism needed
  - Service-oriented architectures (SOA)
    - Services: simple, independent, and flexible entities
    - Rely on higher-level control mechanism
- Which one to use?

Service Oriented Architecture

- We chose SOA, because
  - As application context changes in smart meeting room, so do visualization requirements
  - Incorporating multiple application contexts in the high-level control mechanism of a SOA is more suitable than maintaining all the semantic overhead in all involved agents

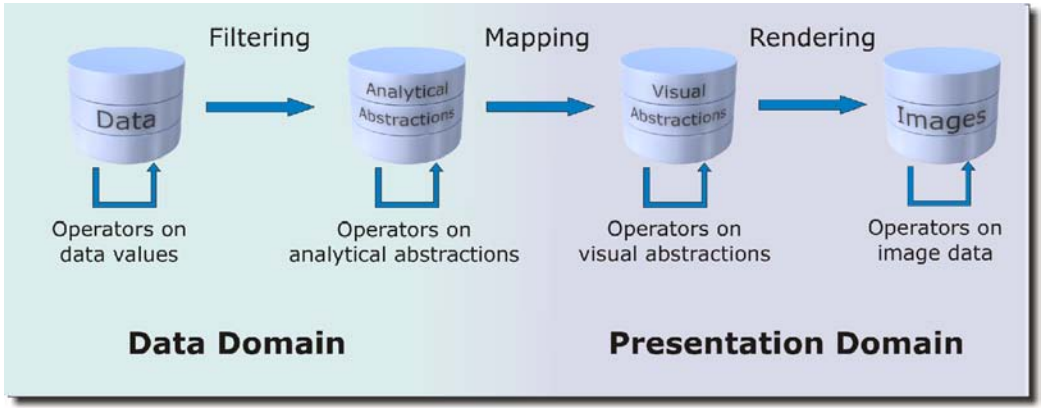
Service Oriented Architecture

Three key questions have to be investigated:

1. What services are required for information visualization in smart meeting rooms?
2. How can services be linked and adapted dynamically at runtime to accomplish visualization tasks given the environment's current situation?
3. How are services invoked to create visual output?

Services

- Starting point:
  Data state reference model (Chi, 2000)



- Operators in Chi's model are implemented as services that can be interconnected

Services

We differentiate between:

- Service interfaces:
  - Basic functionality
  - Describe input and output of services
  - Hide implementation details
- Service implementations:
  - Concrete implementations of some interface
  - Several implementations can coexists to cope with different requirements (e.g., output devices)

Service Linking and Adaptation

- In dynamic environments, visualizations can hardly be compiled in advance
- Instead, visualizations need to be composed at runtime, depending on available resources

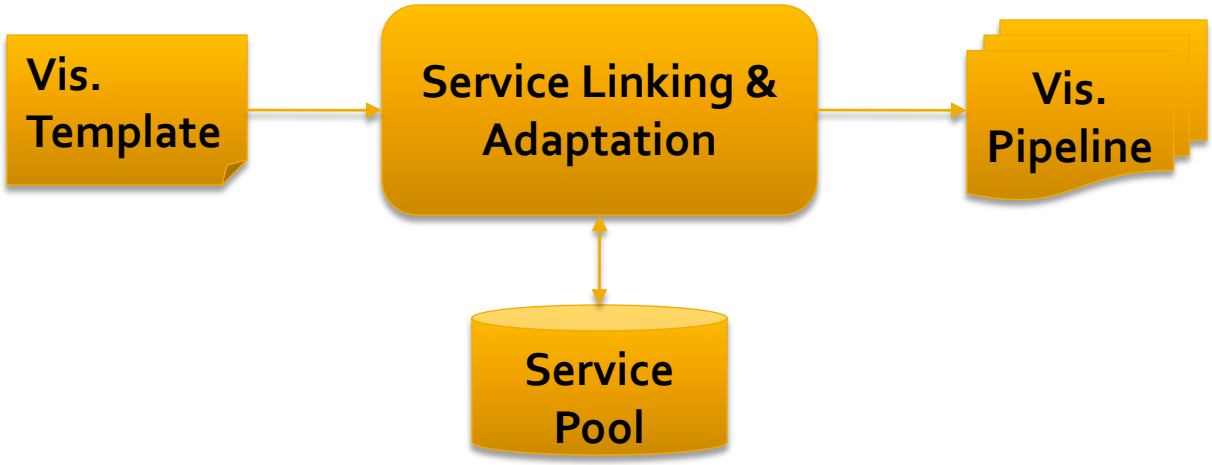
- **Visualization templates**
  - Definition of visualization pipeline based on service interfaces (not implementations)
  - Evaluation of template and binding of available service implementations at runtime

Service Linking and Adaptation

Service adaptation to device characteristics at runtime:

- Selective binding of service implementations
- Inherently adaptive service implementations
- Adaptation of visualization template

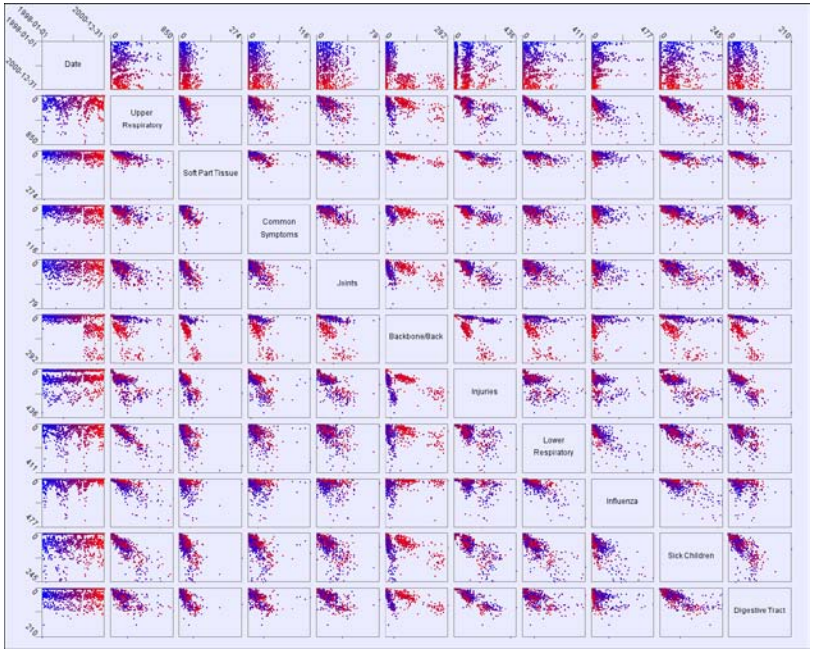
Service Invocation



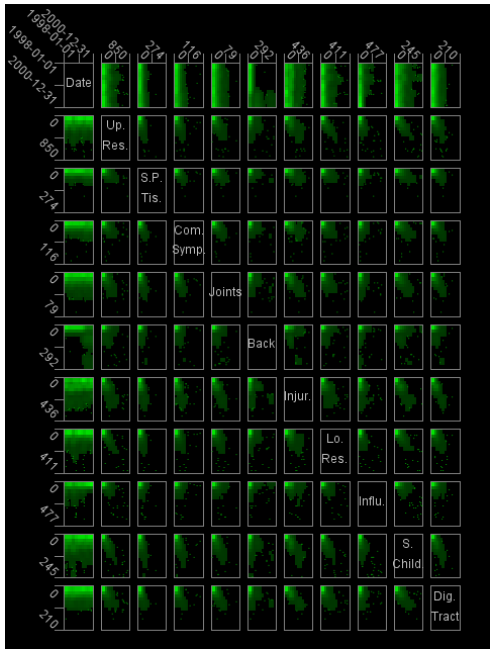
- Service linking and adaptation generates vis. pipeline that is ready to be executed
- Services are invoked in proper order using *Jini*™ and output is transferred to display

Results

Output adapted to device



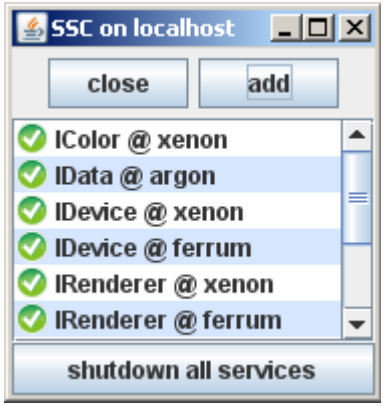
Regular-size implementation



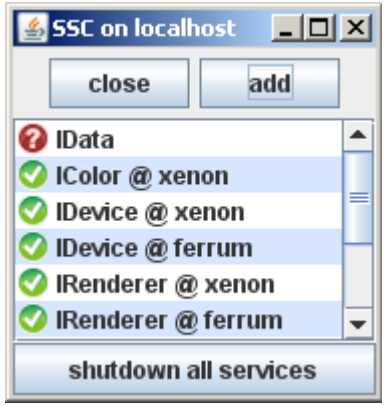
PDA-size implementation

Results

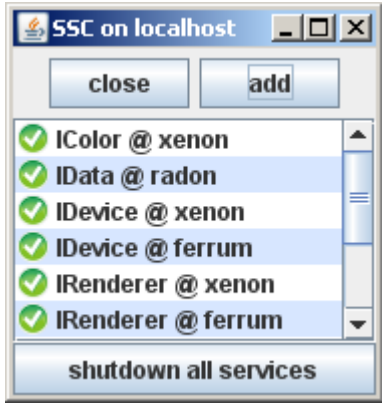
Automatic replacement of vanished service



Data provided by *Argon*



Argon has left the environment



Radon automatically picks up task of providing data

Time

Usage Scenario

<Show video>

Summary & Future Work

- **SOA architecture for information visualization in smart environments**
- **Technical basis to investigate the "really interesting" stuff in the future**
  - Integration of sensors and recognition of user intention
  - More and better adaptation strategies
  - Collaborative information visualization in smart meeting room