



DSCCL: A Language to support Dynamic Service Composition

Caroline Funk, LMU Munich

Jelena Mitic, Christoph Kuhmünch, Siemens AG
Corporate Technology

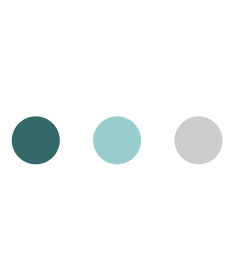
SIPE '06

June 29, 2006, Lyon, France

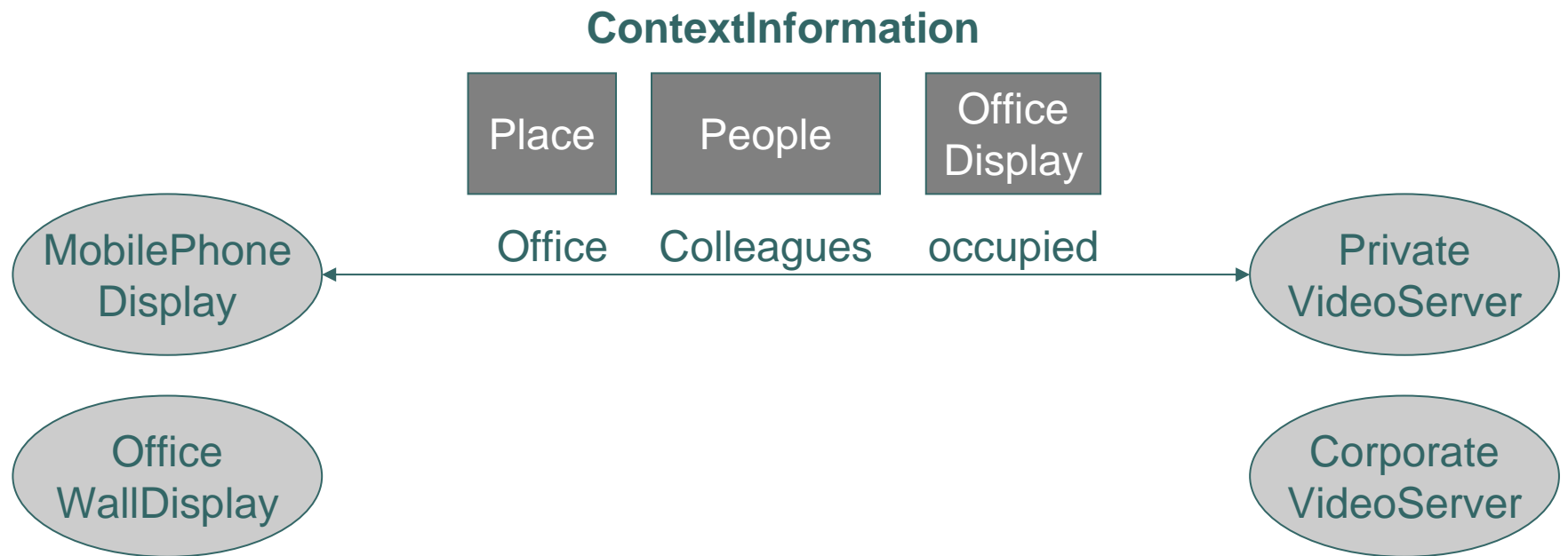


Outline

- Scenario
- Main Requirements
- Related Work
- DSCL Dynamic Service Composition Language
- Implementation
- Status and Outlook



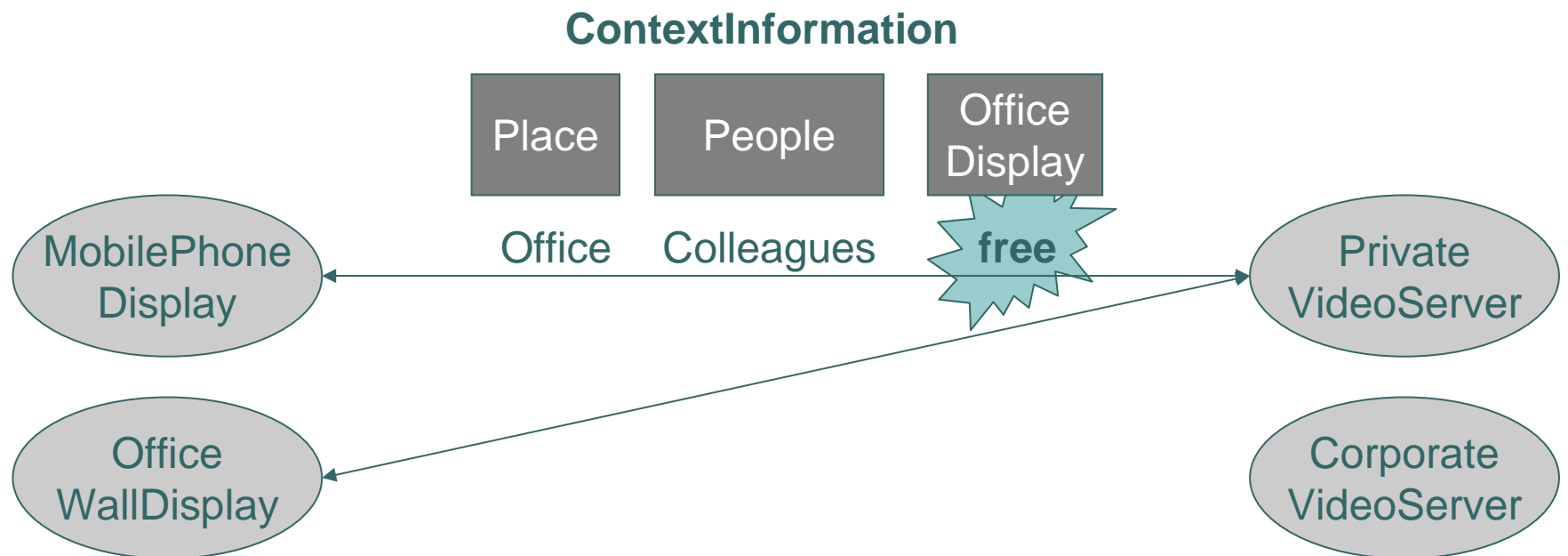
Scenario – VideoShowService



Preferences: use larger display



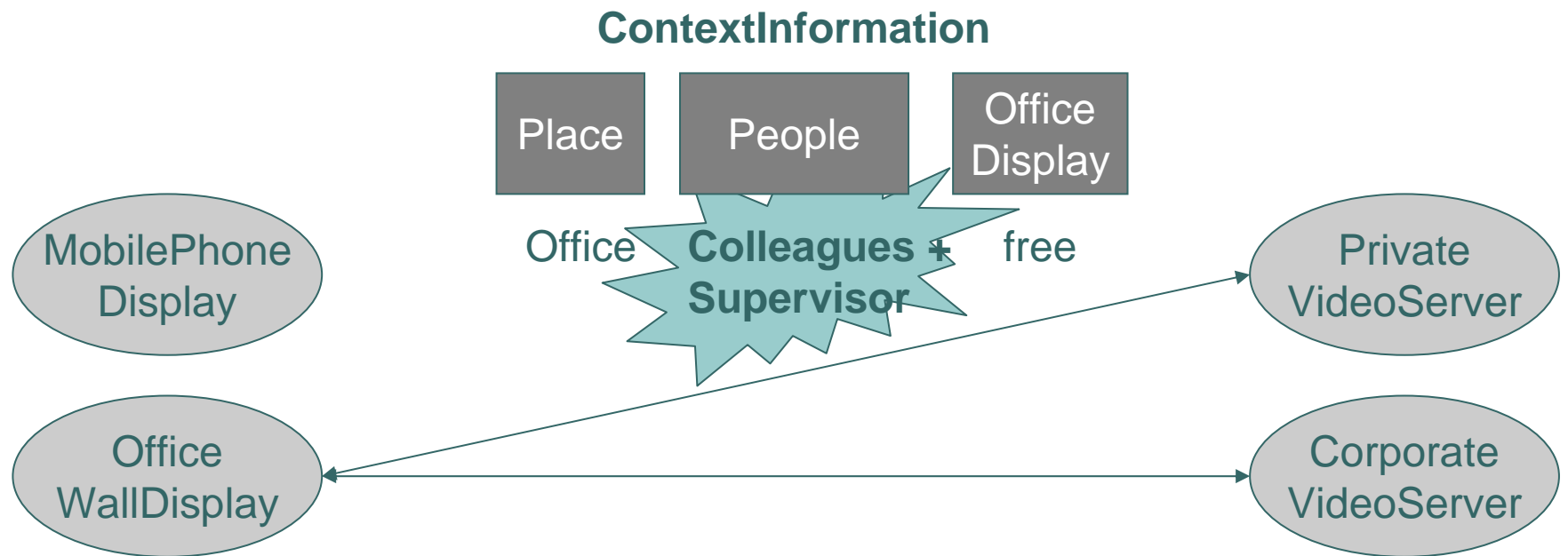
Scenario – VideoShowService



Preferences: use larger display



Scenario – VideoShowService



Preferences: use larger display



Requirements of a Composition Language

- Dynamic Service Composition
 - With respect to availability of services
 - Usage of equivalent services
- Context-Awareness
 - Instantiation according to the situation
 - Selection of services based on the user's context
- Personalization
 - Selection of services based on user preferences
 - Composition based on user preferences



Related Work

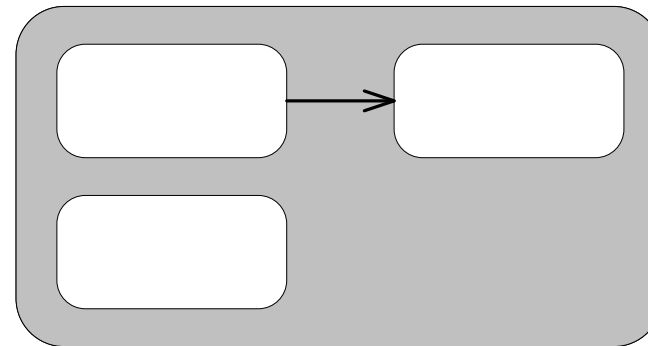
- Web Services: BPEL, WSCI,...
- Semantic Web: OWL-S, WSMO,...
- Pervasive Computing: Aura, PCOM,...

Characteristics:

- binding to types or instances
 - no dynamic composition
 - no usage of equivalent service types
- usage of service specific language constructs
 - no extensibility
- description independent of criteria
 - no context-awareness integrated
 - no personalization

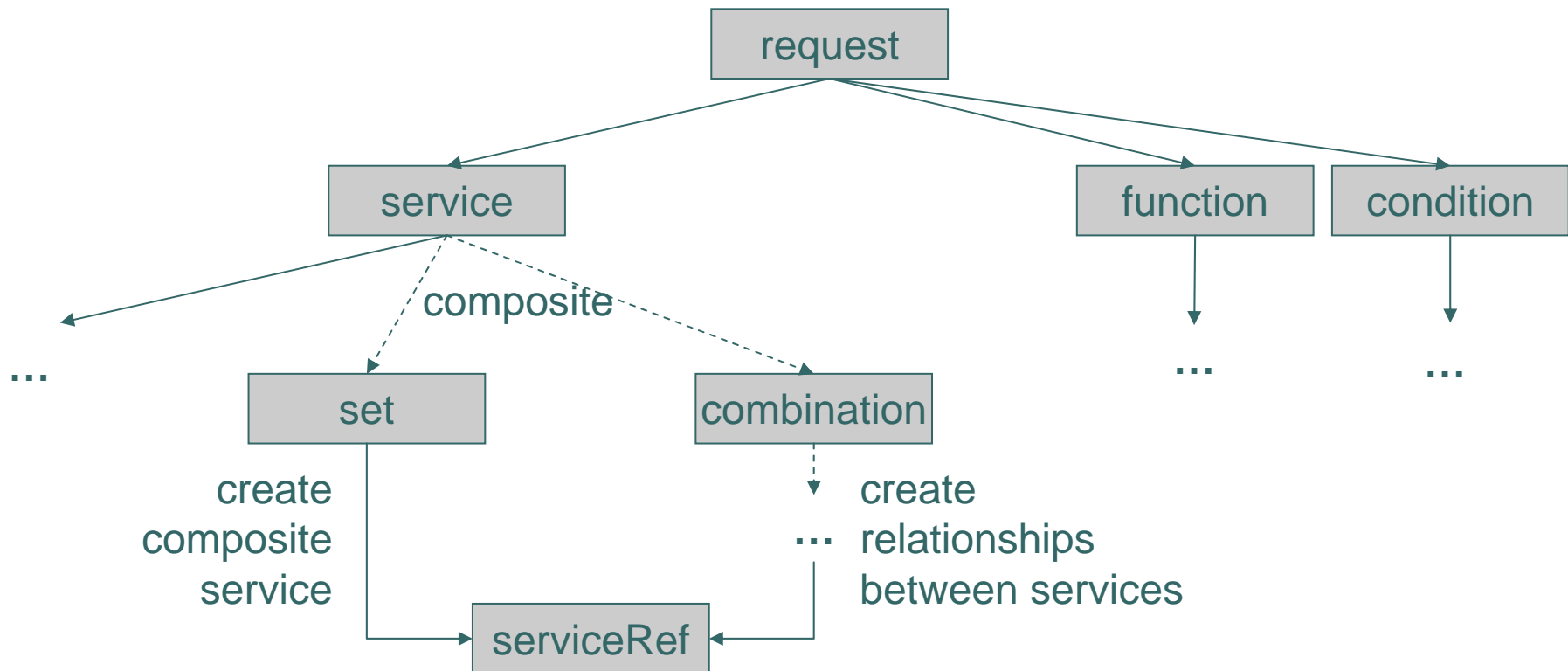
DSCCL Dynamic Service Composition Language

- Goal: description of context-aware personalized composite services allowing for dynamic adaptation
- Basis: ontology describing existing service types and relationships
- XML-based
- Description of (non coherent) directed graph of services



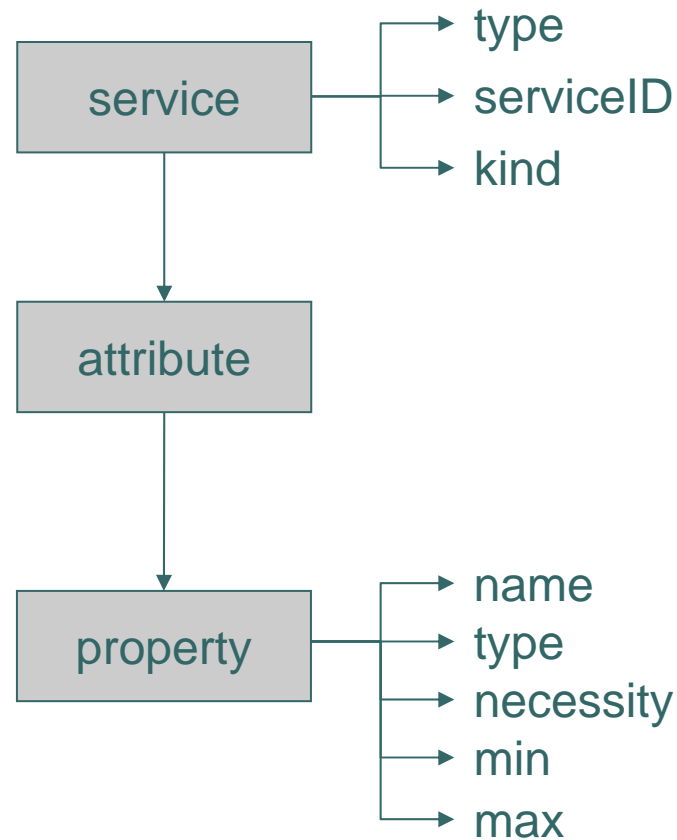


DSCCL Language Overview

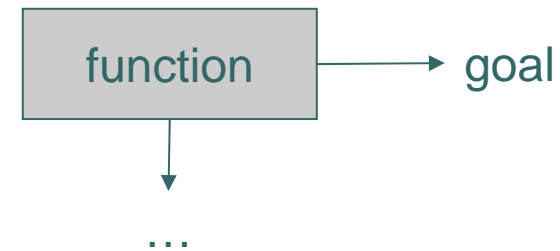
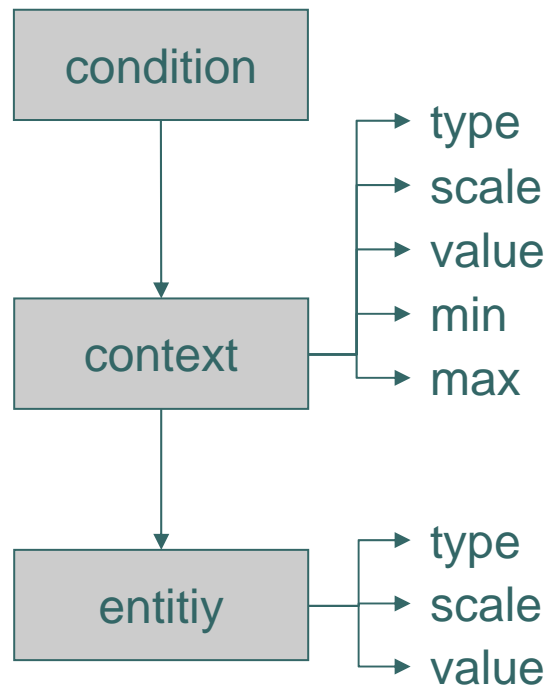




DSCCL Service Specification



DSCCL Conditions and Objective Function



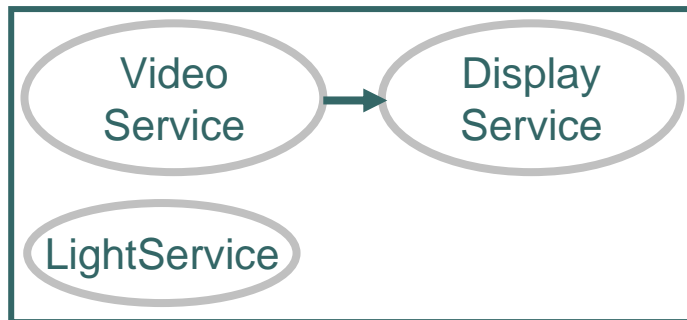
- description of function
- reference to service attributes
- usage of constants

Example

Condition: Location = Lyon, ...

Objective function:

$\min((VS.cost+DS.cost)/2)$



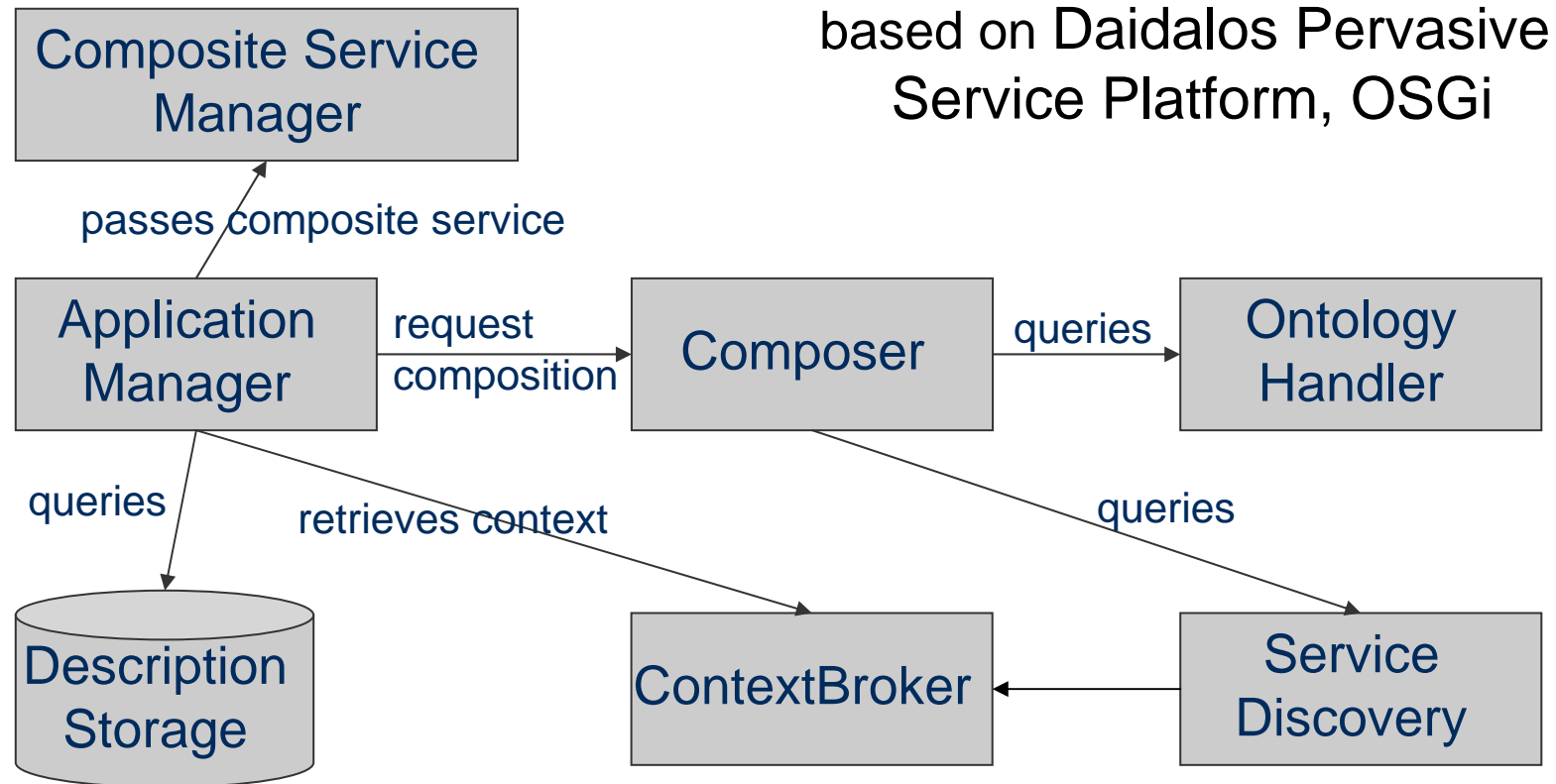
```
<condition>
  <context type=„location“ scale=„city“
    value=„lyon“/> ...
</condition>
```

```
<function goal=„min“>
  <division>
    <addition>
      <functionService ref=„VS“ att=„cost“/>
      <functionService ref=„DS“ att=„cost“/>
    </addition>
    <constant value=„2“/>
  </division>
</function>
```

```
<service type=„LightService“ kind=„single“ id=„LightS“/>
...
<service type=„VideoShowService“ kind=„composite“ id=„VP“>
  <set>
    <serviceRef ref=„V-D-Sequence“ necessity=„mand“>
    <serviceRef ref=„LightS“ necessity=„opt“>
  </set>
</service>
```



Implementation





Status and Outlook

Current Results

- Implementation based on OSGi
- Composition of pervasive services based on DSCCL descriptions

Planned work

- Tool for creating DSCCL documents
- More Elaborated Service Ontology
- Reasoning on adaptation
- Recomposition incl. state transfer



Thank you!