Context Mediation as a Linked Service
MaaS architecture

Pierre De Vettor[1], Michaël Mrissa[1], Carlos Pedrinaci[2]
Laboratoire d'InfoRmatique en Image et Systèmes d'information
UMR5205 CNRS/Université de Lyon, 69622 Villeurbanne Cedex, France
e-mail: pierredevettor@gmail.com, michael.mrissa@liris.cnrs.fr
Knowledge Media Institute
The Open University, Milton Keynes, MK7 6AA, UK.
e-mail: c.pedrinaci@open.ac.uk

Background and Motivation

Problem
- Lack of semantics in classical Web services exchange
- Data representation specific to each country (or community)
- Data from business workflows needs to be adapted

Background
- Use of context to solve heterogeneous data interpretation
- Linked services / Minimal Service Model (MSM) / iServe platform [1]

Goals

- Attaching additional information about context to input and output data in services description
- Runtime discovery of suitable mediation services to be invoked when data flows
- Design of a MaaS architecture for on-the-fly mediation and injection of mediation services

Using context to solve heterogeneities

- Service annotation
  - Performed at design time
  - Based on the Minimal Service Model
- WSDL-like : a `msm:Service` is defined as
  - a set of `msm:Operation` that have for input and output:
  - some `msm:MessageContent` where each piece of data is represented by:
  - a `msm:MessagePart`
- The MSM model allows using the `sawsdl:modelReference` attribute
- With `modelReference` we attach each `MessagePart` to
  - a concept from a domain ontology e.g. `qudt:QuantityOfPrice`
  - a list of contextual properties, e.g. `qudt:CurrencyUnit`, `qudt:EUR`, ...

- Domain ontology based on Quantities, Unit, Physical dimensions and Data types ontologies (`http://www.qudt.org/`)
  - QUDT describes a value as a `QuantityValue` with the following properties:
    - a `Quantity` (e.g. Length)
    - a `Unit` (e.g. Meter)
    - a `NumericValue`
    - an `Uncertainty`
- Proof of concept: A unit mediation service
  - Integrated to iServe for discovery
  - Takes as input : a `qudt:quantityValue` and a `qudt:unit`

Mediation as a Service architecture

- Architecture (accessible as a service)
  - Built around a mediator service (WES)
  - Takes as input (1):
    - a workflow representation
    - input data of the composition
- Operation of the WES
  - WES retrieves MSM description of each service (step 2)
  - Execution of the workflow by orchestrating different services (steps 3, 5 & 7)
  - Extraction of contextual sensitivity between each invocation
  - When conversion is required, the WES sends a *discovery request* to iServe to find a relevant mediation service (steps 4 & 6)
  - This mediation service is invoked to perform the conversion and the WES continues the execution of the composition

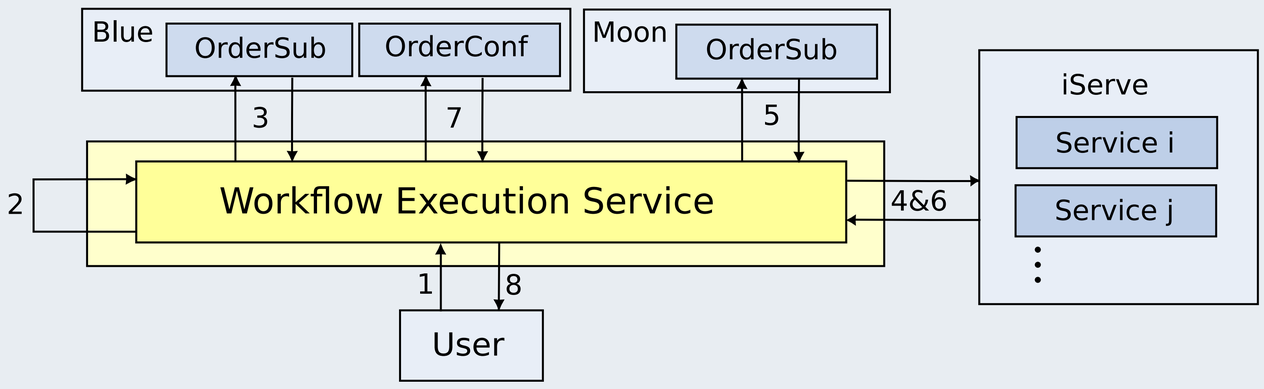

Figure: Overview of our MaaS architecture

Conclusion

- Strengths of our approach
  - Respect of service oriented paradigm
    - Loose coupling
    - Service reuse
    - Composition
  - Promotion of the MaaS (Mediation as a Service) architecture
  - Access via a generic linked service interface
  - Independence from workflow language and use of standards

Future Work

- Plugin-based architecture for distributed discovery
- Integrate schema- and concept-level mediation
- Improve our ad hoc workflow representation language
- Provide compatibility with existing workflow languages such as WS-BPEL
- Enable QoS based selection of mediation services

Bibliography

Carlos Pedrinaci and John Domingue.
Toward the Next Wave of Services: Linked Services for the Web of Data.
Journal of Universal Computer Science, 16(13):1694–1719, 2010.