Imaging Services on the Grid as a Product Line: Requirements and Architecture

M. ACHER, Ph. COLLET, Ph. LAHIRE, J. Montagnat

Workshop SOAPL 2008
September 8th
Context: Services for the Grid

**Grid**

*sharing* data, algorithms
computation power, data-intensive

**Workflows for the e-Science Grid**
process chain, pipeline, data flow
reuse and compose (black) boxes

**Implemented as Services**
Requirements Overview

Functional
Format: DICOM
Acquisition Model: MRI
Anatomic Structure: Brain
Noise: Not significant

QoS
Accuracy: 80%
Security: None
Reliability: 50%
Execution time: Not specified

Services for image segmentation
S1
S2

Functional
Format: DICOM
Acquisition Model: MRI
Structure Anatomique: Stomach
Bruit: Dedicated to noisy images

QoS
evaluation: Statistical
Accuracy: 90%
Reliability: 30%

Functional
Format: DICOM
Acquisition Model: MRI
Structure Anatomique: Brain
Bruit: Noise sensitive

QoS
evaluation: Statistical
Accuracy: 85%
Reliability: 50%
Composing Services on the Grid

How to deploy Grid Services?
- needs fine-grained information

How to manage QoS (Quality of Service)?
- such as execution time, availability, reliability, etc.

To give information to ...
- workflow engine, software architect, scheduler

Our position: a variability problem!
From Service to Product Line (1)
From Service to Product Line (2)
Functional Variability

extract: inputs

Magnetic Resonance Imaging
Acquisition Model
MRI = MRI T2
Resolution
Spatial Resolution
Dimension = 2D
color = B&W
Noise = none
Anatomic Structure = brain
Format = DICOM
QOS Variability

How to **caracterize**

How to **measure**

How to **compute**
QOS description : example

Metric
measurable = true
unit = %
comparable = true
type = numeric

Dimension
accuracy = high
time = any
...

Computation
dynamic = true
rely_on = output
accuracy = good

Measurable

QoS Property

Metric

Value Type

Comparable

Unit

% = false

Numeric

Operator

<=

Accuracy

Dynamic

Output

% = true

Conditions

Output

Output

Output

Output
Segmentation: refining classification

QoS depends on application domain:
- goal of segmentation
- body region
- imaging protocol

“A particular segmentation may have high performance in determining the volume of a tumor in the brain on an MRI image,
... but may have low performance in segmenting a cancerous mass from a mammography scan of a breast”

Dimensions: time and space complexity, accuracy, robustness, precision, specificity, sensibility

Interdependency between QoS and Computation of QoS:
- costly but precise
- quick but uncertain
- evaluation has a QoS too
Towards SPL: big picture
Towards Service product line

Grid workflow expert

Behaviour + QOS + variability

Service Product Line segmentation service

Service Product Line registration service

Service Metamodel
QoS Metamodel
Grid Metamodel

Service Repository

Medical imaging computation expert

Workflow

One Workflow
An MDE Approach

**Equipping Service/Workflow with meta information**
- A common core (QOS & service metamodels)
- Specific branches

**Building the SPL**
- Describing a generic Domain-Specific service / workflow
- Specifying composition protocol of one service
- Allow to address different workflow
- Includes also variability

**Approach**
- Model Driven Engineering (MDE)
- Platform independent, abstraction
- Model transformation and/or model composition
An MDE Approach

Model-Driven Engineering

Workflow? Service Composition

Model abstraction of services

Model-Driven Engineering

Workflow?

Service Composition

eHealth domain

Instance of the SPL

transformation

Platform dependent

GRID Engine
On-going Work

- QoS multi-views
  - experts collaboration
  - from end users to services

- How to infer a SPL?

- Derivation process
  - who for the reasoning process?
  - heuristics needed

- From Service to workflow
From Service to Workflow

- Grid workflow expert
- Service Repository
- Behaviour + QOS + variability
- Service Metamodel
- QoS Metamodel
- Medical imaging computation expert
Questions?