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

Mathieu Acher, Ph. Collet, Philippe Lahire, Johan Montagnat

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

Mathieu Acher*, Philippe Collet*, Philippe Lahire*, Johan Montagnat*
*Université de Nice Sophia Antipolis, laboratoire I3S - CNRS UMR 6070
Polytech Nice - Sophia, 930 Route des Colles, BP 145, F-06903 Sophia Antipolis Cedex, France
acher@polytech.unice.fr, Philippe.Collet@unice.fr, Philippe.Lahire@unice.fr, johan@i3s.unice.fr

Abstract—**SOA is now the reference architecture for medical imaging processing on the grid. Imaging services must be composed in workflows to implement the processing chains, but the need to handle end-to-end qualities of service hampered both the provision of services and their composition. This paper analyses the variability of functional and non functional aspects of this domain and proposes a first architecture in which services are organized within a product line architecture and metamodels help in structuring necessary information.**

I. Introduction

Service Oriented Architectures (SOA) emerged in distributed computing and IT management as a conceptual architecture to reduce the complexity of software systems. This paradigm promotes an organization in which basically i) reusable self-contained services are provided through standardized interfaces and easily exchange the structured information — handling execution platform heterogeneity —, and ii) services are composed and coordinated in workflows, possibly in a dynamic way — so that business processes can be easily created and maintained on the long term —.

SOA is also becoming a reference architecture for grid computing in different scientific domains [1]. Our work stands in the medical imaging area, in which grids help in building patient-specific models and in reducing computing time for meeting time constraints of clinical practice. Grid computing makes also possible to deal with many problems related to large medical data sets manipulation, usually heavily fragmented, on very wide distributed infrastructures. Besides image analysis tool pipelines are undergoing homogenization, strongly motivated by the need for mutualizing software development and easily comparing results. The SOA paradigm is thus especially adapted to this domain, with imaging services inherently decoupled and abstracted from technical platforms, and workflows to design composed algorithms through process chains.

But to fully exploit this paradigm, the community faces two major challenges that we identify as being related to an important *variability*. First, code maintainers have to provide basic imaging services from heterogeneous code, with detailed information so that they can be easily composed to construct new algorithms and to master their deployment on the grid. Second, maintainers have to manage the numerous non functional properties that have to be exploited during deployment or run times, in order to ensure a quality of service (QoS) adapted to the user. These QoS properties expose different forms of variability as they may be related to a service itself (reliability, availability, cost, expected execution time...), to its provision on the grid (parallelism grain, data handling protocol, adaptability to resources...) or to some user needs (emergency of a computation, expected output quality...).

Variability is now an essential concept in software engineering. Its realization can be seen as a means to describe the whole generality of a software artefact through the specification of commonalities and differences. The best support for this concept is currently the Software Product Line (SPL) paradigm [2], which applies to software the general industrial notion of engineering family of similar entities. In our context, we aim at tackling the variability issues in imaging services and workflows on the grid by including them within a product line architecture. This *service product line* would describe major variations in functional and non functional medical imaging service specifications, as well as in process pipelines described through workflows. In this paper, we focus on shorter term goals:

- We present the results of our domain analysis on imaging grid services, detailing segmentation services as an illustration. We therefore propose appropriate metamodels to represent functional variability and QoS mechanisms (Section II).

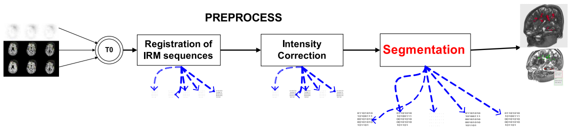

Figure 1. A medical imaging worflow

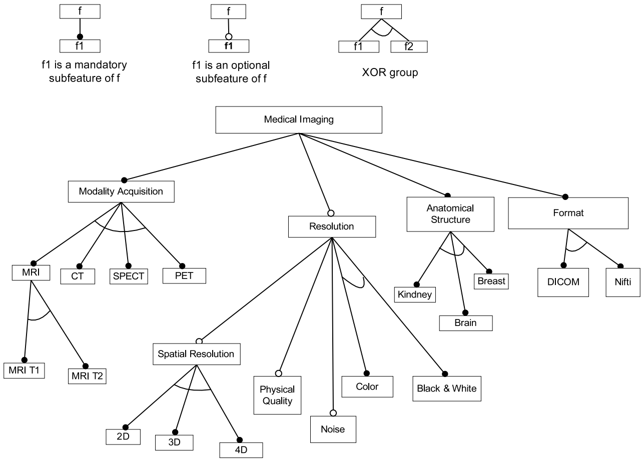

Figure 2. Metamodel handling functional variability in medical imaging (FODA representation)

- We propose a first architecture that enables service providers and workflow experts to capture the commonalities and the differences of legacy services, to build the right service according to these commonalities and differences efficiently and to use the line to select appropriate services according to functional and non functional criteria (Section III).

II. Variability in Medical Imaging Services

In a medical imaging workflow, medical experts compose different kinds of processing on images, each algorithm being provided by a service.

A. *Functional Variability*

Figure 1 shows a workflow corresponding to a brain Magnetic Resonance Imaging (MRI) segmentation. The first stage consists in the registration of images so that they are spatially aligned in the same coordinate system. Then, the brain region is isolated and an intensity correction is applied. This preprocessing stage is followed by image segmentation, which delineates some structures in the brain. For each task of the workflow, numerous services are available on the grid. For instance several segmentation algorithms exist. They are as many candidates to realize the last task in the workflow. However, in our particular context, medical images are MRI, in a specific format — the studied body part is the brain, and following the preprocessing stages, noise can be considered as weak. This context is indeed a combination of varied features: the medical image could have been a CT-scan or an ultrasound and it could represent completely different anatomical regions. In the same time, the functional capabilities of services also change according to the context. For instance some algorithms are better adapted to process MRI than CT-scans; indeed, they are developed to reach this goal. It is then necessary to introduce specific information on images and manipulated body structures, and this form of variability can be directly expressed as *alternative choices*. Following our approach based on metamodels, one metamodel must be capable to provide to medical imaging experts the means to explicit these service features through the concepts of software product line. Model Driven Engineering (MDE) techniques are intended to be used to capture the description of service variability and product-line capabilities.

As a result, we propose a first metamodel to handle functional variability in medical imaging. Figure 2 shows a partial representation of this metamodel as a feature diagram [3]. Several variability issues are then handled : the knowledge of information associated to the image is either mandatory (*e.g.* the image format), optional (*e.g.* image noise) or with variants (*e.g* body structure). The metamodel makes possible to characterize both the target data set and the capabilities of an algorithm to process it.

B. *QoS Variability*

The behavior of image processing algorithms also changes according to *quality* criteria. Typically, one would like to specify a service performance, its capacity

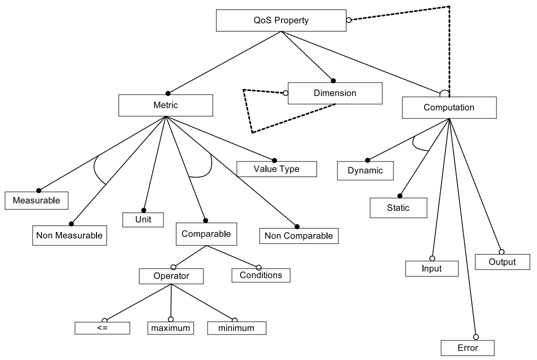

Figure 3. Metamodel handling variability of QoS mechanisms

to handle all kinds of data (robustness), to produce a *good* result (accuracy) or to ensure data confidentiality. Some high-level QoS dimensions have been identified as relevant for grid workflows (time, cost, fidelity, reliability, security) [4]. They are also relevant in medical imaging [5]. Our approach consists in modelling QoS attributes of medical imaging so that grid schedulers, workflow engines or developers can select services on the grid according to QoS constraints. The composition process is then similar to a QoS-driven composition [6]. To illustrate our motivations and approach, we take the case of image segmentation. It consists in locating and extracting perceptual units of the image, so that the image is more meaningful and easier to analyze. In the medical area, the goal of segmentation is to select anatomical or artificial objects, which correspond to the real anatomy of the patient, and which need to be measured or visualised by the clinical user. The process of segmentation is a crucial (and often preliminary) step for medical imaging analysis and diagnosis. We first note that segmentation has no general solution and the need to evaluate segmentation algorithms has often been pointed out [7], both to compare hundreds of algorithm variants developed by researchers and to provide to final users some means to choose the appropriate algorithm for their problems.

An analysis of evaluation mechanisms for segmentation algorithms also shows an important variability. For instance, the computation of quality measures cannot be realized before the processing. This is the case of *goodness* methods, which compute specific properties of segmented objects in the image (such as intra-region uniformity, inter-region-contrast, entropy, shape, edge quality, *etc*). These methods are subjective — it is always possible to build an algorithm capable to outperform all others according to another chosen evaluation criteria — , and consequently they may be biased but they have the advantage of being automatic [8].

In contrast, the *discrepancy* methods rely on a reference image, which is supposed to be the *ground truth*, so that it can be compared according to various criteria to the results of segmentation algorithms. Evaluation methods then expose several variability degrees: some need *explicit* knowledge, can be computed statically (with an *a priori* knowledge) or dynamically, are subject to biases, *etc*. We also consider evaluation methods as being characterized by some QoS properties, also variable: necessary execution time, computation cost or evaluation reliability, *etc*. Besides, during evaluation the *application domain* must also be specified. According to [9], it is determined by three entities: the aim of the segmentation, the studied body region and the image protocol (including the acquisition method). The behavior of a segmentation algorithm is then dependent of this context, as well as the different qualities provided. For instance, a particular segmentation method may have high performance in determining the volume of a tumor in the brain on an MRI image, but may have a low performance in segmenting a cancerous mass from a mammography scan of a breast. It is clear that a QoS metric makes only sense in a very precise context. Figure 3 summarizes our proposed QoS metamodel and its variability description. A QoS property is then described through a metric, a dimension, and optionally a computation method. All analysed variability points can be expressed through this metamodel, and especially the interdependencies between dimensions and QoS properties themselves (informally represented by the dotted lines on the figure, expressed through constraints[1] in our case).

Back to segmentation, several QoS dimensions must be taken into account. The main considered quality is accuracy, which refers to the degree to which the segmentation results agree with the true segmentation. The reproductibility (or precision) is the capability for a process to be repeatable (*i.e.* to reproduce the same result for the same input) and it corresponds to the measure of a random fluctuation. The predictability of an image analysis is also important for experts, which have mental models of how an algorithm works and can then correct a malfunction. The reliability of an image processing chain is also evaluated in terms of robustness, *i.e.* the

[1] using Object Constraint Language (OCL)

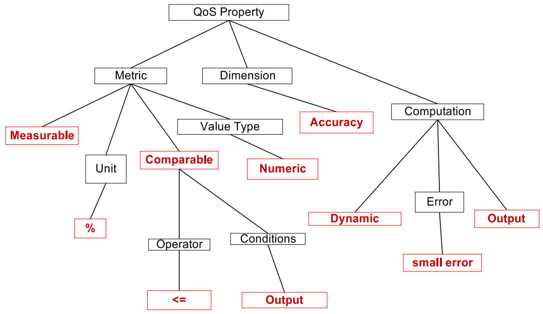

Figure 4. One QoS property model

performance realized in the presence of disruptive factors or its capacity to handle a wide range of images as input. Finally more classic QoS dimensions, such as time or cost, are expressed through theoretical complexity of the algorithms (in space and time). It is clearly important to be able to refine some high level QoS concerns in such a way that medical or software experts have the means to specify their requirements. But a major issue is then to handle mutually dependent constraints between QoS dimensions, which is a classic open problem in computer science: a restriction on memory usage may easily imply a performance drop. With segmentation, such interdependencies are also found, as it is difficult to evaluate the specificity and the sensitivity of an algorithm through a unique metric without trade-off [10]. As a result, we can instantiate the QoS metamodel to characterize each QoS property of a service. In Figure 4, a property model with the following characteristics is instantiated: the QoS dimension is accuracy, it is dynamically measurable, the evaluation is made on the output with a small margin of error and it is possible to compare it with its output. The variability of QoS mechanisms have impacts on the operations to select services, control or adapt the workflow. Back to the workflow of Figure 1, selecting a service according to QoS constraints implies that the service can evaluate *a priori* the QoS dimensions. Here the presented QoS characteristics are not compatible with this selection operation as it only supports dynamic computation and requires the knowledge of the output. But it can be used to perform appropriate monitoring at runtime.

III. Architecture of the Service Product Line

Considering the mentioned requirements, our aim is to tackle the determined variability issues by including services within a product line architecture.

A. Principles

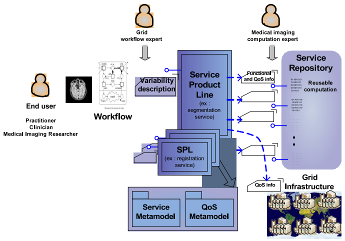

Figure 5. The Software Product Line Framework

Figure 5 sets the principles of the proposed architecture. It relies on *i)* a Service Product Line Framework (SPLF) describing the business domain, *ii)* a service repository where one may find legacy services of the business domain and *iii)* metamodels that capture knowledge of the SPLF. The SPLF describes possible types of services and workflows for the domain of medical imaging. It considers services variability, including:

- a set of common properties (a structured list of assumptions that are satisfied for all members of the domain). For example, any service for image segmentation requires a medical image as input;
- a set of possible differences. For example, the format of medical image may vary depending on the service that is chosen (DICOM, Nifti, *etc*).

The choice to insert or not one type of service (for instance in a preprocessing step) is another variation point, but at the workflow level.

Introducing variability within the description of the used services allows the developers (*e.g.* medical imaging computation expert) to describe the structure and the behaviour of the services, as well as to propose variants, define optional parts, *etc*. Then, the grid workflow experts are able to transparently deploy services and to efficiently execute complete applications composed of several of them. During the process of building and executing the workflow, they choose the most appropriated services for each task of the workflow, using the service product line. Thus, end users (*e.g. practitioners*) are able to specify the data on which the application will be run and to execute it on a grid, specifying their requirements and QoS needs.

B. Services in the repository

The SPLF uses services in a repository. They are making a collection of algorithms for image processing. We provide to medical imaging experts means to organize the information associated to services and make explicit their functional (see Figure 2) and non functional (see Figure 3) characteristics. The concept of service is also modelized through a metamodel, as proposed for example in [11]; it aims at integrating these properties. Through metamodels, services are sharing a common framework in order to describe their interfaces, especially preconditions (*i.e.* nature of manipulated images) and postconditions (*i.e.* QoS provided). QoS properties of services must take into account grid infrastructure characteristics. In particular, the grid latency may lead to performance drop on the execution of a service [12]. QoS grid concerns are referring to many QoS dimensions and are then using variable mechanisms. Typically, one want to express a dynamic measure on the response time of grid nodes, as well as a statistical evaluation about fault tolerance of resources. We thus propose a metamodel representing this information (see Figure 3). Finally, legacy services of the repository are identified as candidates, and are potentially reusable in different contexts.

C. Building the SPL

There are hundreds of segmentation algorithms available on the grid that are able to be used for the segmentation task of the workflow on Figure 1. Intuitively these services may be handled through an actual service (a service interface) of type *Segmentation*, which can be included in one SPL description. Services are then included in the SPL based on additional information associated. Consequently, a line of services can be seen as a service that is able to provide access to multiple services, members of the line. Moreover, a service can be described by multiple interfaces, but has one capability (*i.e.* segmentation). In order to describe the whole generality of an entity, we specify common elements and differences. The result is the construction of a generic interface, which describes indirectly multiple interfaces [13].

As the metamodel would describe all possible software product lines of the business domain (medical imaging), it will be able to represent all functional and non-functional commonalities and variations of the services belonging to the repository. Each service of this repository, which belongs to the software product line, will be conform to a given model and by extension (transitivity) to the metamodel. Thus, the software architect can infer a software product line considering some services of the repository.

D. Using the SPL

Reasoning and verification on the SPLF is supported by model engineering techniques. As the variability of grid services for medical imaging is captured in metamodels, this makes possible to reason on services and to achieve the necessary operations determined in Section II, such as selection, adaptation and monitoring. The SPLF relies on these metamodels to represent one software product line, which corresponds to one of its instances. Thanks to the SPLF, the repository and the metamodels, it is possible to derive services from a given software product line. The main focus during product derivation is on satisfying complex dependencies, *i.e.* dependencies that affect the binding of a large number of variation points, such as quality attributes. A key aspect in resolving these dependencies is having an overview on these complex dependencies and how they mutually relate. As we have already noticed in Section II, an algorithm in medical imaging tries to perform a trade-off between sensibility and sensitivity. Another example of a complex dependency is a restriction on memory usage of a software system. Relation to other dependencies is how this restriction interacts with a requirement on the performance. These examples show the need for the first-class representation of dependencies, including complex dependencies, in variability models and the need for appropriate means to model the relations between these dependencies in the SPL. Thus, the SPL provides to software architect a generic interface describing the set of functional and non functional characteristics and means to express constraints in order to choose the most adapted service for each workflow task.

IV. Conclusion

The goal of this paper was to propose an approach for improving the reuse of services dedicated to the domain of medical imaging. We addressed variability of grid services for medical imaging by using an approach based on software product lines. Functional and non functional variability of imaging services have been analysed using segmentation as a running example. We focused on QoS attributes and provided two supporting metamodels, one for expressing functional variability in image processing, the other to describe QoS mechanisms. We also described the architecture of a service product line relying on these metamodels.

This architecture is undergoing implementation within the Kermeta workbench and its support for Feature Diagram[2]. Beside implementation, we want to validate our approach on a large service and data set. The proposed architecture is intended to be used in a project called NeuroLOG [14]. This project is targeting the neurology domain and adopts a user-centric perspective to meet the neuroscientists expectations. It also aims at fostering the adoption of HealthGrids in a pre-clinical community.

Other open issues deal with the modeling of lines of services as well as with the specification of the QoS properties. As we mentioned in this paper, we are interested in two kinds of variability : *i)* variability of the service itself (for example the reliability of one segmentation algorithm) and *ii)* the variability of the QoS processing mechanism (for example the computation of the reliability measure may be static or dynamic). These two kinds of variability interact one with the other and one crucial issue is to compose them in order to obtain a full-fledged architecture.

As it has been explained in Section II, the properties that are addressed may be considered at various levels of details. For instance one may reason at the level of the time used for handling one service or at the level of the time needed to send its result (the first one includes the second one). Another important aspect is then to be able to reason even if the scale or the type of several properties is not the same and if the information contained need to be aggregated. QoS dimensions may indeed have multiple views, depending on the domain considered. An expert of the grid would not have the same preoccupations as an expert of business domain considering, for instance, the reliability dimension. The impact of both cases must be studied together.

Acknowledgment

The authors thank Diane Lingrand and Alban Gaignard for their valuable comments regarding this work.

[2] http://www.kermeta.org/mdk/ProductDerivation/

References

[1] I. Foster, C. Kesselman, J. Nick, and S. Tuecke, "The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration," Open Grid Service Infrastructure WG, GGF, Tech. Rep., June 2002.

[2] J. Bosch, G. Florijn, D. Greefhorst, J. Kuusela, J. H. Obbink, and K. Pohl, "Variability issues in software product lines," in *PFE*, ser. Lecture Notes in Computer Science, F. van der Linden, Ed., vol. 2290. Springer, 2001, pp. 13–21.

[3] K. Czarnecki, S. Helsen, and U. Eisenecker, "Formalizing Cardinality-based Feature Models and their Specialization," in *Software Process Improvement and Practice*, 2005, pp. 7–29.

[4] J. Yu and R. Buyya, "A Taxonomy of Workflow Management Systems for Grid Computing," *Journal of Grid Computing (JGC)*, vol. 3, no. 3-4, pp. 171 – 200, Sept. 2005.

[5] P. Jannin, J. Fitzpatrick, D. Hawkes, X. Pennec, R. Shahidi, and M. Vannier, "Validation of Medical Image Processing in Image-guided Therapy," *IEEE Transactions on Medical Imaging (TMI)*, vol. 21, no. 12, pp. 1445–1449, Dec. 2002.

[6] L. Zeng, B. Benatallah, M. Dumas, J. Kalagnanam, and Q. Z. Sheng, "Quality driven web services composition," in *WWW '03: Proceedings of the 12th international conference on World Wide Web*. New York, NY, USA: ACM, 2003, pp. 411–421.

[7] Y. J. Zhang, "A review of recent evaluation methods for image segmentation," in *Signal Processing and its Applications, Sixth International, Symposium on. 2001*, vol. 1, Kuala Lumpur, 2001, pp. 148–151.

[8] J. S. Cardoso and L. Corte-Real, "Toward a generic evaluation of image segmentation," *Image Processing, IEEE Transactions on*, vol. 14, no. 11, pp. 1773–1782, 2005.

[9] J. K. Udupa, V. R. Leblanc, Y. Zhuge, C. Imielinska, H. Schmidt, L. M. Currie, B. E. Hirsch, and J. Woodburn, "A framework for evaluating image segmentation algorithms," *Computerized Medical Imaging and Graphics*, vol. 30, no. 2, pp. 75–87, March 2006.

[10] A. Popovic, M. de la Fuente, M. Engelhardt, and K. Radermacher, "Statistical validation metric for accuracy assessment in medical image segmentation," *International Journal of Computer Assisted Radiology and Surgery*, vol. 2, no. 3-4, pp. 169–181, December 2007.

[11] A. D'Ambrogio, "A model-driven wsdl extension for describing the qos ofweb services," in *ICWS '06: Proceedings of the IEEE International Conference on Web Services*. Washington, DC, USA: IEEE Computer Society, 2006, pp. 789–796.

[12] D. Lingrand, J. Montagnat, and T. Glatard, "Estimating the execution context for refining submission strategies on production grids," in *Assessing Models of Networks and Distributed Computing Platforms (ASSESS) (CCgrid'08)*. Lyon: IEEE, May 2008.

[13] C. Feier, D. Roman, A. Polleres, J. Domingue, M. Stollberg, and D. Fensel, "Towards intelligent web services: Web service modeling ontology (wsmo)," in *Proceedings of the International Conference on Intelligent Computing (ICIC)*, Hefei, China, 8 2005.

[14] J. Montagnat, A. Gaignard, D. Lingrand, J. Rojas Balderrama, P. Collet, and P. Lahire, "NeuroLOG: a community-driven middleware design," in *HealthGrid*. Chicago: IOS Press, June 2008, pp. 49–58.