

PhD research subject proposed by LIP6

PhD thesis topic: Dynamic service orchestration based on non-functional constraints and instantaneous QoS measurements in a SOA architecture.

Context of the PhD thesis work:

The world of distributed computing is undergoing profound change due to the emergence of service-oriented architectures. This architectures uncouple a distributed application in two separate levels: a service offerings level specified by their interfaces and, possibly, semantic information in a service registry. The second level is the physic implementation of these services on interconnected platforms (via internet) and accessible using standard protocols.

As such, the development of a new application can be seen as the high-level coordination of a swarm of services. The composition of these services enables the fulfillment of the application's goal. A process written in an extension to the "BPEL" business-process execution language can implement this coordination, called "orchestration". The execution of this orchestration will link abstract service calls to compatible service implementation as provided by the service registry.

The matter is there to build the link between an abstract specification of the process requirements and web-services' properties available on the network in order to conduct the process to its realization.

PhD thesis focus:

From the intersection of distributed computing and pervasive systems at one end and SOA architectures at the other end, comes to light multiple constraints (like network disconnections, independent failures, low and predictable response times) and characteristics (mobility, redundancy, heterogeneity). In such context, a BPEL process must be able to take into account not only service and network failures but, also, a set of runtime choices between various possible services. These choices will be made dynamically according to the instantaneous Quality of Service (QoS) values provided by the various services. More generally, these QoS values include all non-functional properties that the services are willing to publish. They can range from technical non-functional information like disponibility, availability and bandwidth to more high-level properties often related to the services' domains of capability.

In this thesis, we propose to develop a BPEL engine not only capable of dynamically linking between an abstract service call and its implementation based on semantically described functional requirements and properties of the various SOA components; but also able of dynamically managing in a "late binding" fashion the instantaneous QoS provided by various web-services.

The other challenge in this thesis will be to present the developers with high-level programming abstractions easy enough to be fluently manipulated and that can be adapted to each application domain.