Context-aware Distributed Deployment of Component-based Applications

Dhouha Ayed, Chantal Taconet, Nawel Sabri, and Guy Bernard

GET / INT, CNRS Samovar 5157, 9 rue Charles Fourier, 91011 Évry, France

1 Introduction

Software deployment refers to all the activities performed after the development of a software in order to make it available to users. These activities mainly consist of installation, configuration, activation of the software but also of reconfiguration, update and de-installation of the software [1].

With recent advances in wireless networking technologies and mobile computing devices, making applications context-aware becomes essential. The reason is that, with these new technologies, device resources (e.g. memory, battery, CPU) may be scarce, and application execution context(e.g. user location, device screen size) is variable [2].

Context information may be taken into account at different times of the application life cycle. We argue that the deployment is an effective period to take context state into account. Context information allows the deployer to install an application suited to each execution context. We introduce in this presentation the concept of context-aware deployment [3].

Context-aware deployment has to be processed just-in-time (i.e. when a user accesses to a service). Just-in-time deployment enables users to automatically install and configure applications suited to their needs, the resources of their computers and their surrounding environments. Because software is installed at access time only and may be de-installed afterwards, just-in-time deployment enables scarce resources saving. Furthermore, just-in-time deployment implies the automation of the deployment tasks and thus relieves the user from the deployment repetitive tasks.

Our work focuses on multi-component applications which are described by an assembly. An application assembly defines component instances and their connections [4]. With multi-component applications, deployment may be distributed [5]. In this way, some application components may be hosted by the user's computer, while other components are hosted by computer servers.

The solution we present in the poster is dedicated to context-aware distributed deployment of multi-component applications.

2 Context-aware Deployment

We summarize briefly in this section : (1) the type of decisions allowed by a context-aware deployment, (2) the data models proposed for context-aware deployment and (3) the deployment platform we have designed.

During context-aware deployment, the context has the following impacts: (i) the choice of the application structure (components' assembly) which may vary according to the context - the application structure defines the type of each component instance; (ii) since several implementations may be available for each given component type, context-aware deployment has to choose the suited implementation for each component instance; (iii) the placement of each component within a set of computers is another choice of just-in-time deployment; and finally, (iv) the configuration of each component has to be set according to the context.

We define two data models for the description of information necessary for context-aware deployment. Firstly, a data model to define the context which is relevant for the deployment. Secondly, a data model to define the rules necessary to make the deployment choices. For those two data models, the descriptions may be defined at two levels: (i) generic (i.e. applicable to any application) and (ii) specific (i.e. applicable to a given set of applications).

In order to make a deployment service context-aware, we have added a set of components to an existing middleware deployment service layer. Some of these components are hosted by the user terminal and others by the deployment provider domain. These components include context collectors, repositories and a deployer adapter. The role of the deployer adapter is to generate a deployment plan in which all the deployment decisions have been taken according to the context.

3 Conclusion

Our context-aware deployment scheme may be seen as a deployment pre-processor which generates a fixed deployment plan. The deployment is then realized by the underlying middleware deployment service. A first implementation of the context-aware deployer has been developed. It has been coupled with the OpenCCM middleware platform to make context-aware deployment of CCM applications. The implementation is under evaluation on some demonstrator applications.

References

1. Hall, R.S., Heimbeigner, D., Wolf, A.L.: A cooperative approach to support software deployment using the software dock. In: International Conference on Software Engineering. (1999) 174–183
2. Dey, A.K: Understanding and Using Context. Personal and Ubiquitous Computing Journal **5** (2001) 4–7
3. Ayed, D., Taconet, C., Bernard, G.: A Data Model for Context-Aware Deployment of Component-based Applications onto Distributed Systems. Component-oriented approaches to context-aware systems Workshop ECOOP'04 (June 2004)
4. OMG: UML Superstructure 2.0. Adopted Specification ptc/03-08-02 (August 2003)
5. OMG: CORBA Component Model Version 3.0. Adopted Specification formal/2002-06-65 (June 2002)