ADELFE

Name

ADELFE – Atelier de DEveloppement de Logiciels à Fonctionnalité Emergente (i.e. Toolkit to develop software with emerging functionalities)

Reference Materials

- RNTL-funded French national project. December 2000-August 2003.
- Partners: Artal Technologies (http://www.artal.fr), Institut de Recherche en Informatique de Toulouse (http://www.irit.fr) at Paul Sabatier University, Laboratoire Informatique, Image, Interaction (http://www.univ-lr.fr/labo/l3i) for the University of La Rochelle and TNI-Valiosys (http://www.tni-valiosys.com/).
- Web site for ADELFE (download, tools): http://www.irit.fr/ADELFE
- Related web site: http://www.irit.fr/SMAC
- Papers: http://www.irit.fr /SMAC/PROJETS/Project_ADELFE.html#References

Contact

Marie-Pierre GLEIZES (gleizes@irit.fr)

Design Tool

ADELFE is based on OpenTool, a Rational Rose-like software developed and licensed by TNI-Valiosys. OpenTool has been adapted for ADELFE and a trial-version is provided with the methodology.

Modelling Language

UML is adopted. AUML is also used.

Scope

Problems today are becoming more and more complex, incompletely specified, and immersed in a dynamical environment; therefore an a priori known algorithm cannot be given as solutions. Self-organization in a means to rearrange the structure of a system and then changing the global function it performs. If every agent is able to rearrange its interaction depending on local criteria, the system is able to adapt without having the knowledge of the global function it has to achieve. In Adaptive Multi-Agent Systems (AMAS), self-organization is led by cooperation and an agent finds the right place within the organization because it wants to cooperatively interact with others. According to the AMAS theory (cf. http//www.irit.fr/SMAC), this cooperation is sufficient to obtain a functionally adequate system.
ADELFE is an agent-oriented methodology intended to guide an engineer during the development of AMAS. Agents considered in these systems are cooperative ones.

Short Description

ADELFE wants to cover all the phases of a classical software design: from the requirements to the deployment. A well-know process, the RUP, has been tailored to take into account specificities coming from the design of Adaptive Multi-Agent Systems. Only the requirements, analysis and design phases require modifications in order to be adapted to AMAS, others appearing in the RUP remain the same.

Therefore, ADELFE adds nothing to preliminary requirements.
The adaptation process of the system depends on the interactions between the system and its environment; this latter must then be studied by the designer, during the final requirements. Entities which interact with the system are identified and qualified as being active (autonomous or dynamic behaviour, agents will be found among them later) or passive (resource) ones. The system's environment is studied and qualified with given terms as a beginning of reflection. During the identification of use-cases, cooperation failures must be highlighted using a specific notation.
In the analysis stage, once the preliminary class diagram elaborated, the designer has to verify if the proposed technology (AMAS) is useful to build the target-system. A graphical tool is provided to more easily study this point. This adequacy is studied at the system level but also at the component one because a component can be viewed as an AMAS itself if it needs to evolve.
Agents are not considered as being known in advance and must be identified depending on the features entities may have (autonomy, partial view of its environment, cooperation failures…). Interactions between entities are described and relations between agents are expressed using protocol diagrams (and AUML supported by OpenTool).
During the design stage, if agents communicate by interaction (they can also communicate using the environment), interaction languages (classes, implementation of FIPA ACL…) are studied after having identified the detailed architecture.
Cooperative agents are equipped with several modules representing a partition of their "physical", "cognitive" or "social" abilities. These modules must be identified for every agent in the system, therefore building the behaviour of the agent. The most important module is the "Non Cooperation Situations (NCS)" one; it gives agents the ability to detect and remove NCSs and to always stay in a cooperative state.
Once the behaviour of agents defined, the simulation functionality of OpenTool enables the designer to test them.

Tools Linked with ADELFE

Three main tools are integrated into the ADELFE toolkit:

- An interactive tool which describes the process and helps the designer to apply it,
- OpenTool, a graphical modelling tool that supports the UML notation and which has been modified to integrate new stereotypes linked with ADELFE and AUML interaction protocols,
- A tool which analyses answers given by the designer to tell him if using the AMAS technology is useful to implement the target system.