

ALIVE: A Framework for Flexible and Adaptive Service Coordination*

J. SC. Lam¹, W. W. Vasconcelos¹, F. Guerin¹, D. Corsar¹, A. Chorley¹, T. J. Norman¹,
J. Vázquez-Salceda², S. Panagiotidi², R. Confalonieri², I. Gomez², S. Hidalgo²,
S. A. Napagao², J. C. Nieves², M. Palau Roig³, L. Ceccaroni³, H. Aldewereld⁴,
V. Dignum⁴, F. Dignum⁴, L. Penserini⁴, J. Padget⁵, M. De Vos⁵, D. Andreou⁵,
O. Cliffe⁵, A. Staikopoulos⁶, R. Popescu⁶, S. Clarke⁶, P. Sergeant⁷, C. Reed⁷,
T. Quillinan⁸, K. Nieuwenhuis⁸

University of Aberdeen¹, Universitat Politècnica de Catalunya²,
Tech Media Telecom Factory SL³, Universiteit Utrecht⁴, University of Bath⁵,
Trinity College Dublin⁶, Calico Jack Ltd⁷, Thales Nederland B.V.⁸

Abstract. There is a large body of research on software services, but the issues of communication and dynamic reconfiguration have received little attention, as have adaptation to environment and dynamic combination of service building blocks into new applications. Here, we present the approach of the FP7 ALIVE project to the use of formal models of *coordination* and *organisation* mechanisms to deliver a flexible, high-level means to describe the structure of interactions between *services* in the environment. Our aim is to create a framework for services engineering for “live” open systems of active services. We propose to build on the current activities in service-oriented engineering by defining three levels: (i) An organisational level models the organisational structure of executing and interlinked services and the context around them. (ii) A coordination level provides flexible ways to model interaction between the services. (iii) These two levels connect with existing (semantic) Web services, which contain semantic descriptions to make components aware of their social context and of the rules of engagement with other services.

1 Introduction

New generations of networked applications based on the notion of software services which can be dynamically deployed, adjusted and composed will make it possible to create radically new types of software systems. These systems shall be able to communicate and reconfigure flexibly at runtime, adapt to their environment and dynamically combine sets of building block services into new applications. In order to achieve this objective, the ALIVE project aims at providing a semantic-based and context-aware framework by bringing together the leading edge methods from two highly promising areas – Coordination Technology and Organisational Theory. The project will adopt the latest Semantic Web technologies to connect to existing service-oriented systems. The ALIVE framework combines model driven development with coordination and organisational mechanisms, providing support for live (that is, highly dynamic) and open systems of services. We will demonstrate the implemented framework which validates and

* This work has been performed in the framework of the FP7 project ALIVE IST- 215890, which is funded by the European Community (<http://www.ist-alive.eu>)

tests the ALIVE approach in three use cases from three industrial partners respectively: dynamic crisis management, communication in entertainment domains and dynamic orchestration of distributed services on interactive community displays.

2 ALIVE Framework

The ALIVE framework aims to support the design, deployment and maintenance of distributed systems by (1) allowing the coordination, reorganisation and adaptation of Web services, (2) following operational constraints defined in the organisation level; and (3) adapting to the dynamic nature of Web Services at runtime.

2.1 Organisation Level

The organisational model views an organisation as a social system, and describes what the aims and the concerns of the organisation are with respect to the social system. The OperettA tool¹, an Eclipse plugin, is an organisational modelling tool. Its function is to create and manage the organisational model of a given distributed system. The designer is able to design the whole organisational level of a given distributed system through abstract concepts such as objectives, roles, obligations, violations, sanctions and high-level interaction diagrams that only identify critical states (called landmarks) and landmark patterns. The organisational model is specified in terms of four structures: (1) The *social structure* specifies objectives of the society, its roles, role dependencies and what kind of model governs coordination (see Fig. 1 (B)). (2) The *interaction structure* describes interaction moments, as scene scripts, representing a society task that requires the coordinated action of several roles, and gives a partial ordering of scene scripts, which specify the intended interactions between roles (see Fig. 1 (A)). (3) The *normative structure* expresses organisational norms and regulations related to roles. (4) The *communicative structure* specifies the ontologies for description of domain concepts and communication illocutions.

2.2 Coordination Level

The *coordination level* provides a means to specify the patterns of interaction between services, and to transform the organisational representation coming from the organisational level into service-oriented workflows. The Coordination Design Tool are created (as Eclipse plugins) for administrators to design the whole coordination level of a distributed system by means of actors, tasks, workflows and workflow coordination mechanisms. The tools also support the generation of the agents that will perform the actual coordination tasks and the inspection of predefined and generated workflows. The Graphical Action editor (see Fig. 1 (D)) produces machine processable action descriptions, which can be used by other components, such as the workflow synthesis and agent tab. A workflow is composed of a sequence of steps, where each step is associated with an action and input bindings for the inputs of that action, along with a link to the next step in the workflow. Generated workflows are initially used by the workflow editor (see Fig. 1 (C)), which supports visualisation of a workflow, editing of a workflow, and uploading/downloading of a workflow to/from the workflow repository. Workflows are subsequently used by the agents for enactment. The multi-agent generator (see Fig. 1 (E)) takes the OperettA model and the actions defined for the organisation and creates an initial multi-agent system where every role in the organisation is assigned to an agent

¹ <http://www.cs.uu.nl/research/projects/opera/>

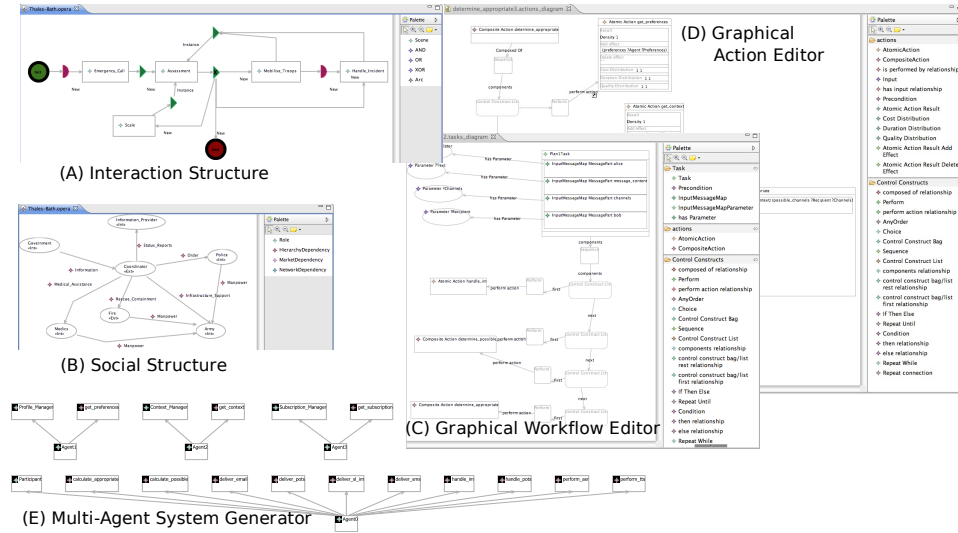


Fig. 1: Screen shots from Eclipse: (A) & (B) Interaction and Social Structure Editor in Operetta; (C) Graphical Workflow Editor; (D) Graphical Action Editor; (E) Multi-agent System

and the actions are distributed to the agents according to the role. The Planning agent chooses a plan and sends it to all the coordination level agents; each agent enacts the actions it is responsible for. The agents run in the AgentScape² platform.

2.3 Service Level

The *service level* supports the semantic description of services and the selection of the most appropriate service for a given task. It also effectively supports higher level and dynamic service composition. The service design tool is used to generate or inspect service descriptions, edit service templates and register them in the service directory. It also connects with the service matchmaking tool (a human interface to the matchmaker component), allowing administrators to search for services matching a given task description or implementing a given service template and registering it in the service directory. The service setup tool is used to check and modify the setup of the running environment, including the URIs of different resources, facilitating components, pre-defined services and service containers. Sometimes the administrator may want to manually change an automatically selected service for another that is considered more suitable (for reasons not modelled within the ALIVE framework); in this case the service matchmaking tool is used to search for services that match a given task description (by using the matchmaker component).

The Monitor Tool covers the three levels. It allows administrators to inspect the status of a system's execution, and to keep track of the events generated at execution time and inspect how the system handles them. The tool aggregates and analyses event logs related to the execution of services, the fulfilment of coordination plans and the achievements of role and/or organisational objectives; and hence feedback is provided to the organisational model and workflow generation.

² <http://www.agentscape.org/>