2013 IEEE 7th International Conference on Self-Adaptive and Self-Organizing Systems

Self-Organising Services

Giovanna Di Marzo Serugendo, Jose Luis Fernandez-Marquez
University of Geneva, Institute of Services Science, Geneva, SWITZERLAND
{Giovanna.DiMarzo, JoseLuis.Fernandez}@unige.ch

***Abstract*—This paper discusses the notion of self-organising mechanisms, such as spreading or gossip, provided as services on top of which more complex applications can be built. Their functionality is provided as the result of the interactions among several entities, possibly distributed across several nodes. Self-organising services are either provided as "core" services by the underlying platform, or as higher level services. A typical example of a "core" service is the diffusion of a gradient, which is a spatial structure spanning multiple nodes. This paper describes a series of self-organising services, their implementation and a crowd steering case study.**

***Index Terms*—Self-organisation, spatial structures, services.**

I. Introduction

The thousands (even millions) of mobile phones, laptops, tablets locally connected to each other form huge computing and storage infrastructures, currently under-exploited but available on motorways, city-centres, inside buildings, etc. Those infrastructures pave the way for a new category of services based on data propagation among devices, e.g. information dissemination in a crowd to better steer the crowd. The development of those services faces multiple challenges. One of them being their engineering. The engineering of distributed applications on top of mobile infrastructure is still done in an ad hoc manner and is far from software engineering practice, where applications are developed in a modular way by relying on (and reusing) libraries and lower-level services.

This paper proposes *self-organising services*, i.e. self-organising mechanisms *provided as services*. We discuss two series of self-organising services: (1) "core" services, provided by a computing environment, that act as building blocks useful to develop distributed applications or higher-level services; (2) higher-level services spatially distributed and provided by the interactions arising among many nodes. We conclude with implementation considerations and a case study.

II. Related Work

This section briefly describes two approaches we leverage for engineering self-organising systems.

A. Self-Organising Mechanisms as Design patterns

Self-organising mechanisms expressed as design patterns help identifying the *problems* that each mechanism can solve, the specific *solution* that it brings, the *dynamics* among the entities involved in the pattern and details of their *implementation*. Several authors have proposed self-organising mechanisms following the software design pattern scheme [1], [2], [3]. Leveraging and extending these work by focusing on the relation between the mechanisms, [4] define a catalogue of self-organising design patterns showing how the different patterns relate with each other and how some patterns are composed by others. This paves the way for their definition as services. Figure 1 shows its organisation into basic patterns (e.g spreading), used into composed ones (e.g. gradients), finally exploited by higher-level ones (e.g. chemotaxis).

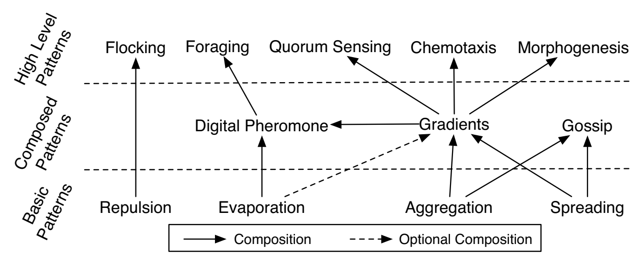

Fig. 1: The Self-organising Design Patterns catalogue [4]

B. Execution Models

We highlight three different execution models for communications protocols: (1) Fraglets [5] unify code and data into a single unit, the so called "fraglet" - a string composed of symbols values. The first symbol value is a tag that represents the instruction to execute over the fraglet. Fraglets are stored in a fraglet store in the same way as tuples are stored in a shared tuple-space; (2) Rule-based Systems [6], where self-organisation is reached by a given set of rules acting on passive data packets. They are similar to fraglets, however, a piece of data does not contain code, and a rule can operate over several pieces of data in the same execution; and (3) BIO-CORE [7] follows both fraglets and rule-based systems. It provides basic bio-inspired mechanisms under the form of "core bio-inspired services" implemented by rules. Moreover, pieces of data upon which services apply contain properties, similar to tags, indicating to the engine how that piece of data must be processed. BIO-CORE provides core self-organising services acting as building blocks on top of which higher-level self-organising services and applications are built.

III. Self-organising services

We propose to program bio-inspired systems using services providing self-organising mechanisms abstracting away the implementation of low-level operations for the programmer and favouring the re-use of code. The computational environment provides reliable "CORE" services, i.e. low-level self-organising mechanisms (e.g., gradient, digital pheromone) and

978-0-7695-5129-6/13 $26.00 © 2013 IEEE
DOI 10.1109/SASO.2013.19

proposes them to the applications under the form of ready-to-use "operators" or "services". The implementation of higher-level self-organising services and applications happens in a modular way, reusing mechanisms, decoupling them from the applications' functionality, and delegating responsibilities for them to the computational environment, as shown in Figure 2.

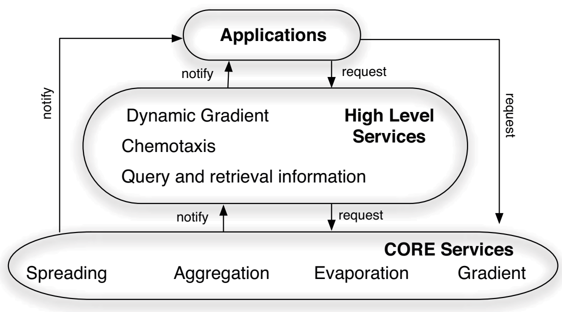

Fig. 2: Self-Organising Services

A. Core Self-Organising Services

CORE services provided at the infrastructure's level are:

Spreading is a service that propagates information among the nodes. It allows agents to increase the global knowledge of the system.

Aggregation service allows the system to reduce the amount of information spread in the system or taken from the environment. Aggregation is also used to get meaningful information for large sets of data.

Evaporation service, also called decay service, helps dealing with dynamic environment. It periodically decreases the information's relevance to get rid of outdated information.

Gradient service combines the Spreading and Aggregation services in order to provide information about the sender's distance (i.e. hops number) and direction.

B. Higher-Level Self-Organising Services

Higher-level services, developed by extending or combining self-organising services provided by the infrastructure, are:

Dynamic Gradient is an extension of the gradient service. It provides a spatial structure (similar to the one provided by the gradient) that is periodically updated to deal with network topology changes.

Chemotaxis allows information to be routed to the gradient source by following a gradient's information.

Query and retrieval information is a composed service that spreads a query using the Gradient or Dynamic Gradient service and uses the Chemotaxis service to bring back information from remote nodes. An exhaustive analysis of the performance of this service can be found in [8].

C. Implementation

The proposed services are being implemented using a middleware provided in SAPERE EU project [9]. In that particular implementation, "CORE" services have been implemented by rules following the BIO-CORE execution model. Higher-level services are implemented as external libraries and provided with the SAPERE middleware.

IV. Crowd Steering in a Music Festival

This case study was designed for the Awareness Summer School (AWASS 2012). A large number of mobile devices are locally connected with each other, without any centralised entity (i.e. switch or router).

Let's consider two users: user A wants to meet user B and user B allows to share its position with user A. A injects in the network a query for the GPS position of B. The query spreads among the nodes using the Gradient service. When B receives the query, it sends back the information using the Chemotaxis service. The Chemotaxis service routes the information following the gradient created by the query. When the information about B position reaches A, the application, navigation tool, steers A towards the desired position. Navigation tools are commonly available for smart phones, by giving a desired position, the current position and the direction, a compass in the screen steers the user to the desired position. Periodic queries for B positions help dealing with the movement of the users. Thus, even though B changes its position, A will be able to find her.

Future work will concentrate on studying the reliability of these services and their compositions.

Acknowledgment

This work has been supported by the EU-FP7-FET Proactive project SAPERE Self-aware Pervasive Service Ecosystems, under contract no.256873.

References

[1] O. Babaoglu et al., "Design patterns from biology for distributed computing," *ACM Trans. on Autonomous and Adaptive Sys*, pp. 26–66, 2006.

[2] L. Gardelli, M. Viroli, and A. Omicini, "Design patterns for self-organizing multiagent systems," in *2nd Int. Workshop on Engineering Emergence in Decentralised Autonomic System (EEDAS) 2007*, T. D. Wolf, F. Saffre, and R. Anthony, Eds. CMS Press, June 2007, pp. 62–71.

[3] T. De Wolf and T. Holvoet, "Design patterns for decentralised coordination in self-organising emergent systems," in *proc. of the 4th int. conf. on Engineering self-organising systems*. Springer-Verlag, 2007, pp. 28–49.

[4] J. L. Fernandez-Marquez, G. D. M. Serugendo, S. Montagna, M. Viroli, and J. L. Arcos, "Description and composition of bio-inspired design patterns: a complete overview," *Natural Computing*, pp. 1–25, 2012.

[5] C. F. Tschudin, "Fraglets - a metabolistic execution model for communication protocols," in *proc. of 2nd Annual Symposium on Autonomous Intelligent Networks and Systems (AINS), Menlo Park*, 2003.

[6] F. Dressler, I. Dietrich, R. German, and B. Krüger, "A rule-based system for programming self-organized sensor and actor networks," *Comput. Netw.*, vol. 53, pp. 1737–1750, 2009.

[7] J. L. Fernandez-Marquez, G. D. M. Serugendo, and S. Montagna, "Bio-core: Bio-inspired self-organising mechanisms core," in *Bio-Inspired Models of Networks, Information, and Computing Systems*, ser. LNCS, Social Informatics and Telecommunications Engineering. Springer Berlin Heidelberg, 2012, vol. 103, pp. 59–72.

[8] J. L. Fernandez-Marquez, A.-E. Tchao, G. D. M. Serugendo, G. S. J. Ye, and S. Dobson, "Analysis of new gradient based aggregation algorithms for data-propagation in distributed networks," in *1st Int. Workshop on Adaptive Service Ecosystems: Nature and Socially Inspired Solutions (ASENSIS) at 5th IEEE International Conference on Self-Adaptive and Self-Organizing Systems (SASO2012)*, 2012.

[9] F. Zambonelli et al., "Self-aware pervasive service ecosystems," *Procedia Computer Science*, vol. 7, pp. 197–199, Dec. 2011, proc. of the 2nd European Future Technologies Conference and Exhibition 2011 (FET 11).