

Ubiquitous and Pervasive Application Design

Mohamed Bakhouya*

Université de Technologie de Belfort-Montbéliard
Laboratoire Systems et Transports
Rue Thierry Mieg 90010 Belfort Cedex, France
voice: +33 3-8458-3431
fax: +33 3-8458-3342
email: Mohamed.bakhouya@utbm.fr

Jaafar Gaber

Université de Technologie de Belfort-Montbéliard
Laboratoire Systems et Transports
Rue Thierry Mieg 90010 Belfort Cedex, France
voice: +33 3-8458-3252
fax: +33 3-8458-3342
email: gaber@utbm.fr

(* Corresponding author)

Ubiquitous and Pervasive Application Design

M. Bakhouya, Université de Technologie de Belfort-Montbéliard, France

J. Gaber, , Université de Technologie de Belfort-Montbéliard, France

INTRODUCTION

The recent evolution of network connectivity from wired connection to wireless to mobile access together with their crossing has engendered their widespread use with new network-computing challenges. More precisely, network infrastructures are not only continuously growing but their usage is also changing and they are now considered to be the foundation of other new technologies. Related research area concerns ubiquitous and pervasive computing systems and their applications. The design and development of ubiquitous and pervasive applications require new operational models that will permit an efficient use of resources and services and a reduction of the need for the administration effort typical in client-server networks (Gaber, 2000, Gaber, 2006). More precisely, in ubiquitous and pervasive computing, to be able to develop and implement applications, new ways and techniques for resource and service discovery and composition need to be developed. Service discovery is the process of locating which services are available to take part in a service composition. Service composition process so far concentrate in combining different available existing services as results of service discovery process. Most of research works to date in service discovery and composition are based on the traditional Client-Server paradigm (CSP). This paradigm is impracticable in ubiquitous and pervasive environments and does no meet their related needs and requirements. Gaber has proposed two alternative paradigms to the traditional client/server

paradigm (CSP) to design and implement Ubiquitous and Pervasive Computing applications in (Gaber, 2000, Gaber, 2006): the adaptive Servers/Client Paradigm (SCP) and the Spontaneous Service Emergence Paradigm (SEP).

Bio-inspired approaches are adequate to carry out these new paradigms for designing and implementing ubiquitous and pervasive applications (Gaber, 2000). Indeed, the adaptive Servers/Client Paradigm (SCP), considered as the opposite to CSP, could be implemented via a self-adaptive and reactive middleware inspired by a biological system like the natural immune system. The Service Emergence Paradigm (SEP) could be also implemented by a natural system that involves self-organizing and emergence behaviors (Gaber, 2000).

Recently, agent-based approaches, with self-adapting and self-organizing capabilities, have been proposed in (Bakhouya, 2005; Bakhouya & Gaber, 2006 a; Gaber & Bakhouya, 2006 b) to implement SCP and SEP respectively. More precisely, these approaches, inspired by the human immune system, provide scalable and adaptive service discovery and composition systems for ubiquitous and pervasive environments.

UBIQUITOUS COMPUTING

In Ubiquitous Computing (UC), the objective is to provide users the ability to access services and resources all the time and irrespective to their location (Weiser, 1993). Service discovery and access systems can be classified into three categories as depicted in figure 1: structured systems, unstructured systems and self-organized systems. Structured systems can be classified also in indexation-based architectures and hashing-based architectures. In indexation-based architectures, there are two categories: centralized and decentralized systems. In centralized indexation-based systems, typical resource discovery architectures (Bettstetter & Renner, 2000), such as Jini (Jini, 2001), consists of three entities: service providers that create

and publish services, a broker that maintain a repository of published services to support their discovery, and services requesters that search the service broker's repository. Centralized approaches scale poorly and have a single point of failure. To overcome the scalability problem, decentralized approaches, such m-SLP (Zhao, Schulzrinne & Guttman, 2000) or Secure Service Discovery Service (Xu, Nahrstedt & Wichadakul, 2001), have traditionally a hierarchical architecture consisting of multiple repositories that synchronize periodically. In Hashing-based architectures (Wang & Li, 2003), proposed primary to file-sharing, Distributed Hash Tables (DHTs) are used to assign files to specific nodes. This technique allows the implementation of direct search algorithm to efficiently locate files. However, a Hashing-based architectures require overlay networks between nodes that are generally hard to maintain.

In unstructured systems, the most typical localization mechanisms are flooding and random walk. There are two main flooding techniques: the push and the pull technique. In the first technique, the server advertises periodically their services across the network. The clients receive the service advertisement and cache the information. This information must have a time period associated with it, and must be flushed out from the cache when this time period expires. Hence, the user has a complete knowledge on the available services and no request resolution process is required. In the pull technique, the client has no knowledge of services present in the network. In this case, a service request is broadcast to all neighbors within a certain radius with TTL tag (TTL for Time To Live) (Wang & Li, 2003). A random walk is a stochastic process that evolves in the following manner (Gaber & Bakhouya, 2006 a). A client send its query message (i.e., a walker) to a randomly chosen neighbor. At each step, the query message is forwarded to a neighbor of its current location and the process continues this way by taking random steps that are independent of all the previous ones until to meet the required service. Consequently, the

random walk technique avoid message duplication inherent to the flooding mechanism (Wang & Li, 2003). More precisely, by using one walker, it cuts down the message overhead significantly. Nevertheless, the delay for a successful request resolution could be high. To decrease this delay, a requester could send k parallel query messages, and each query message takes its own random walk. However, it is difficult to determine a priori a suitable value for k . In other words, if this number k is big enough, the message traffic could increase considerably. An alternative approach to avoid this problem uses both random walks together with an intelligent cloning agent-based technique for service discovery (Gaber & Bakhouya, 2006 a).

It should be noted that the fundamental aspect of these systems is the process of service discovery based on the traditional Client to Server paradigm. More precisely, it is the user who should initiates a request, should know a priori that the required service exists and should be able to provide the location of a server holding that service. This is why the use of repositories is essential in these discovery systems. However, ubiquitous environments have the potential ability to integrate a continuously increasing number of services and resources that can be nomadic mobiles and partially connected. A user can be mobile or partially connected and its ability to use and access services will no longer be limited to those that she/he has currently at hand or those statically located on a set of hosts known a priori. Therefore, the ability to maintain, allocate and access a variety of continuously increasing number of heterogeneous resources and services distributed over a mixed network (i.e., wired, wireless, and mobile network) is difficult to achieve with the traditional Client-Server approaches (Gaber, 2000, Gaber, 2006). More precisely, these architectures cannot meet the requirements of scalability and adaptability simultaneously. The way in which they have typically been constructed is often very inflexible due to the risk of bottlenecks, the difficulty of repositories updating or the network

loading problem. This is particularly true for the cases where some services could be disconnected from the network and new ones may join it at any time.

An appropriate model has been proposed originally in (Gaber, 2000) as an alternative to the traditional Client/Server paradigm (CSP). This model can be viewed as opposed to the Client/Server model and is denoted *adaptive Servers/Client Paradigm* (SCP). In this model, it is the service that comes to the user. In other words, in this paradigm, a decentralized and self-organizing middleware should be able to provide services to users according to their availability and the network status. As pointed out in (Gaber, 2000), such a middleware can be inspired from biological systems like the natural immune system. The immune system has a set of organizing principles such as scalability, adaptability and availability that are useful for developing a distributed networking model in a highly dynamic and instable setting. In (Gaber, 2000, Gaber, 2006), the immune-based approach operates as follows: unlike the classical Client/Server approach, each user request is considered as an attack launched against the global network. The immune networking middleware reacts like an immune system against pathogens that have entered the body. It detects the infection (i.e., user request) and delivers a response to eliminate it (i.e., satisfy the user request).

Recently, an immune approach using mobile agents with cloning capabilities has been proposed in (Bakhouya, 2005; Bakhouya & Gaber, 2006 a; Gaber & Bakhouya, 2006 b) to implement SCP. A mobile agent is a software program which may move from location to location to meet other agents or to access resources provided at each location. Using mobile agent that can clone itself in order to increase system robustness and performance is an attractive idea. The clone operation creates multiple instances of an agent that run on different machines. For example, an initial mobile agent starts on the requesting machine and after a local step,

creates replications (i.e. clones) that initiate parallel walks to further machines. This would allow agents to cover a much wide area of machines space in a reasonable amount of time (Gaber & Bakhouya, 2006 a). However, it should be noted that increasing agent population, with cloning operation, will increase resource demands in the network, which would indirectly affect network performance. Since mobile agents operate in a dynamic and distributed environment, it is difficult even impossible to estimate a priori an appropriate number of agents in the network (Bakhouya, 2005). Also, changing the population dynamically in response to its environment is a complex issue in the absence of central controller. A distributed approach for the regulation of mobile agent population and inspired by the immune system is proposed in (Bakhouya, 2005; Bakhouya & Gaber, 2006 b).

The immune system consists primarily of lymphocytes that circulate through the body in the blood and lymph system. There are two categories of lymphocytes, the B-cells and T-cells. The B-cells are developed in the bone marrow and the T-cells are developed in the thymus. The principle function of T-cells is to potentate immune response by the secretion of specialized factors that activate other cells to fight off infection. The major function of B-cell is the production of antibodies in response to foreign antigen. According to Jerne, B-cells are interconnected by affinity relationship against foreign antigens and form idiotypic networks (Jerne, 1974).

The mapping between the immune system entities and the middleware agents is done in the following manner. T-cells represent servers while B-cells represent services and resources. Antigens correspond to client requests while antibodies corresponds to delivered responses. Servers are organized into communities by the creation of affinity relationships in order to represent services in the network. The establishment of relationship affinities between servers

allows to solve, by collaboration, user requests. A reinforcement learning mechanism is used to adjust and reinforce dynamically relationship affinity values according to delivered responses (Bakhouya, 2005). This reinforcement mechanism permits to cope with dynamic changes in the network, the services availability and the user requests. Similar to the natural immune system, new communities may be created or other may be modified according to dynamically changing environment. In other words, servers may acquire new or drop current servers through establishing or deleting affinity relationship.

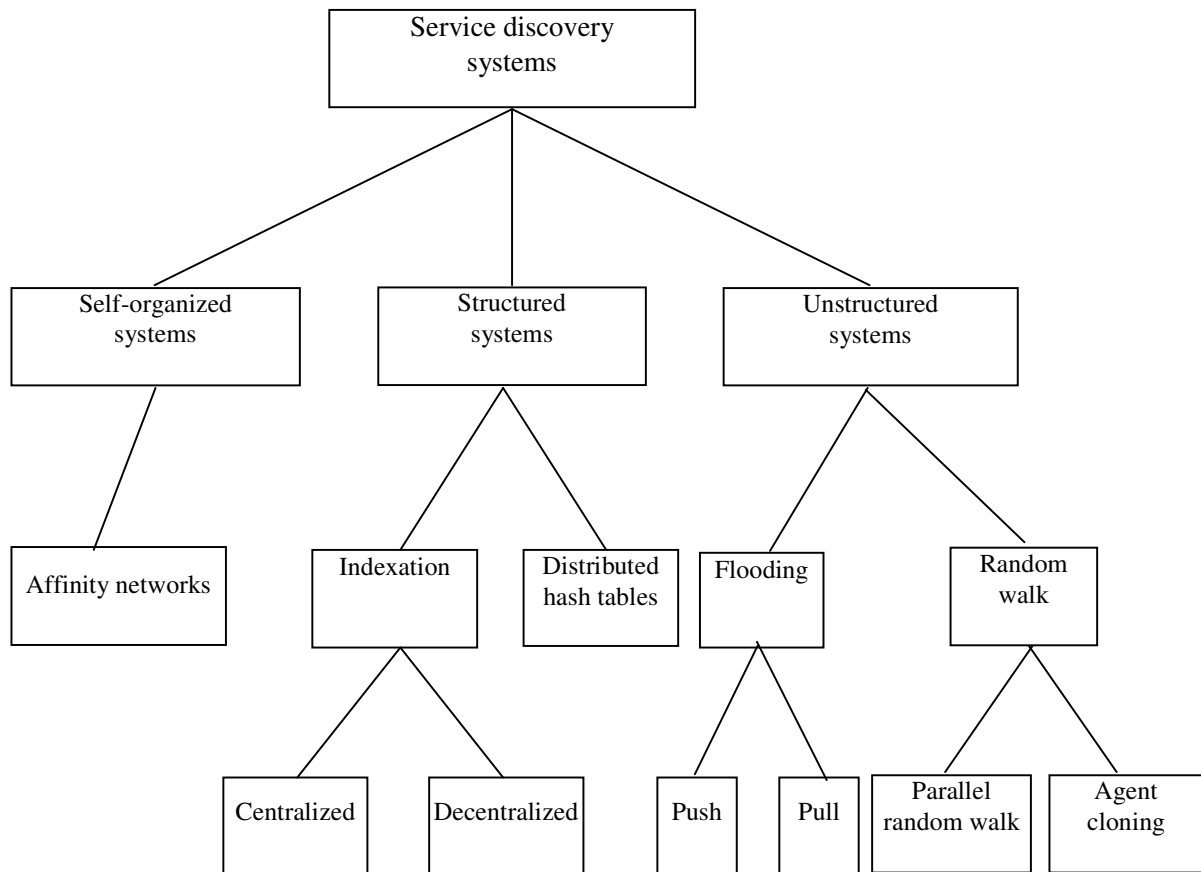


Figure 1. Classification of service discovery systems according to their architectures and their operating modes

PERVASIVE COMPUTING

Pervasive Computing (PC) often considered the same as ubiquitous computing in the literature, is a related concept that can be distinguished from ubiquitous computing in terms of

environment conditions. We can consider that the aim in UC is to provide any mobile device an access to available services in an existing network all the time and everywhere while the main objective in PC is to provide spontaneous services created on the fly by mobiles that interact by ad hoc connections (Gaber, 2000; Gaber, 2006). Service composition systems can be classified into three categories such depicted in figure 2 : proactive composition systems, reactive composition systems and spontaneous emergent service systems. The first category refers to offline composition of available services to form new ones. Services that may be used for proactive service composition can be considered stable, widely and always available (Chakraborty & Joshi, 2001). The second category refers to the process of creating a composite service on demand. In other words, a composite service is created only when a user requests the execution of that service. Most known reactive and proactive service composition systems such as eFlow system (Casati, Innicki, Jin, Krishnamoorthy & Shan, 2000) are based on a centralized broker which manages the service composition process (Chakraborty & Joshi, 2001). The drawback is that if a huge number of users attempt to access variety and increasing number of services distributed over the network, the broker becomes quickly a bottleneck. It should be noted also that these systems are based on the client/server paradigm; it is the user who should initiates a request and moreover services and future demands are known in advance.

Gaber has proposed a second alternative paradigm to the Client/Server one for service composition (Gaber, 2000; Gaber, 2006) that suits pervasive environments. This paradigm involves the concept of spontaneous emergence and is called the *Spontaneous Service Emergence Paradigm* (SEP). This paradigm can be carried out also by an inspired natural immune middleware that allows the emergence of ad hoc services on the fly according to dynamically changing context environments such as computing context and user context (Gaber,

2000). More precisely, in this model, ad hoc or composite services are represented by an organization or group of autonomous agents. Agents correspond to the immune system B-cells. Agents establish relationships based on affinities to form groups or communities of agents in order to provide composite services. A community of agents corresponds to the idiotypic network in the human immune system (Gaber, 2000).

More generally, agents together with their affinity relationships as a whole form a *propitient multi-agent system* (Bakhouya & Gaber, 2006 b). A propitient system is a system with the ability to self-organize in order to adapt towards the most appropriate agent organization structures according to unpredictable changes in the environment. This emergent behavior is delivered as result of agents-to-agents and agent-to-environment interactions that adapt until the system hits a most suitable affinity network. Therefore, a propitient multi-agent system implements the Service Emergence Paradigm (SEP).

A self-organizing approach assumes that individual agents are autonomous agents while multi-agent organizations are emerged structures that are not represented explicitly, but they exist through the affinity relationships between agents. In other words, agents cooperate equally rather than being assigned subordinate and supervisory relationships. It is worth noting that this multi-organizations based on dynamic affinities supported by relationships provides a highly decentralized system while remaining adaptive in dynamic and open environments. More precisely, this decentralized organizational structure offers a high degree of resilience against agent leaving the organization. For example, when an agent leaves an organization, all the peer affinity relationships with other agents are removed without additional messages since it does not rely on any overlay control structure. An affinity-driven clustering learning mechanism could be used to adjust the affinity relationships between nodes to cope with the user context and provoke

or produce an emergent service (Gaber, 2000). More precisely, an ad hoc emergent service is created spontaneously on the fly for a user or between a group of users in unpredictable manner (i.e., without a priori intention).

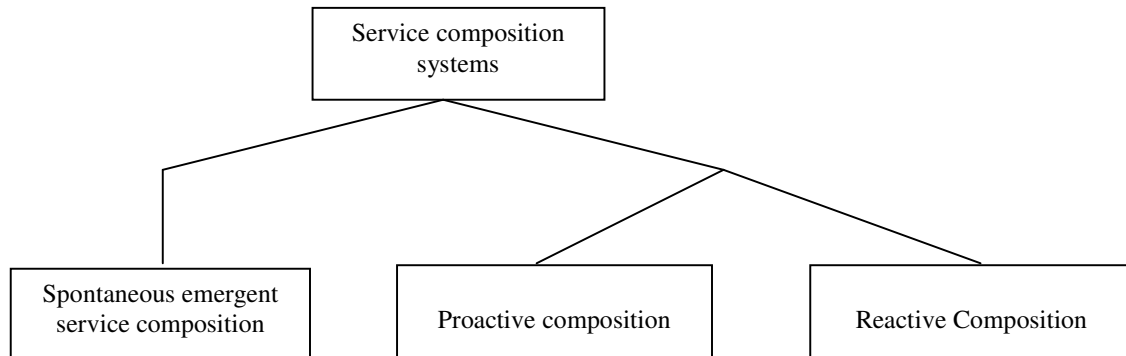


Figure 2. Classification of service composition systems according to their architectures and their operating modes

CONCLUSION

The design and development of ubiquitous and pervasive applications require alternative operational models to the traditional Client/server paradigm (CSP). Adaptive Servers/Client Paradigm (SCP) and Spontaneous Service Emergence Paradigm (SEP) are more adequate to ubiquitous and pervasive computing respectively. Service discovery and composition systems based on these three paradigms and proposed in the literature are presented with emphasis on a self-organizing and self-adapting approaches inspired by the immune system to implement SCP and SEP. Self adaptation and Self-organization are crucial issues in systems that operate in an open and dynamic environment.

REFERENCES

Weiser M. (1993), "Hot topics: Ubiquitous computing". *IEEE Computer*.

- Jerne N. (1974), "Towards a network theory of the immune system". *Ann. Immunol*, 125 (Inst. Pasteur) 125-373.
- Gaber J. (2000), "New paradigms for ubiquitous and pervasive computing", *Research Report RR-09*, Université de Technologies de Belfort-Montbéliard (UTBM), France.
- Gaber J. (2006), "New paradigms for ubiquitous and pervasive applications", Proc. of *First Workshop on Software Engineering Challenges for Ubiquitous Computing*, Lancaster, UK.
- Xu D., Nahrstedt K., & Wichadakul D. (2001), "Qos-aware discovery of wide-area distributed services". In *First IEEE/ACM International Symposium on Cluster Computing and the Grid (CCGrid)*, Brisbane, Australia.
- Robert M. (2000), "Discovery and Its Discontents : Discovery Protocols for Ubiquitous Computing", *Research Report UIUCDCS-R-99-2132*, Department of Computer Science University of Illinois Urbana-Champaign, Urbana, USA.
- Bakhouya M. (2005), "Self-adaptive approach based on mobile agent and inspired by human immune system for service discovery in large scale networks", *PhD Thesis N°34*, Université de Technologies de Belfort-Montbéliard (UTBM).
- Bakhouya M. & Gaber J. (2006 a), Adaptive Approaches for Ubiquitous computing. *Mobile networks and wireless sensor networks*, Eds. Hermes Science, ISBN 2-7462-1292-7, pages 129-163.
- Bakhouya M. & Gaber J. (2006 b), "Adaptive approach for the regulation of a mobile agent population in a distributed network", In *5th International Symposium on Parallel and Distributed Computing (ISPDC'06)*. IEEE Press. Timisoara, Romania.

- Bakhouya M. & Gaber J. (2006 c), “Self-organizing Approach for Emergent Multi-agent Structures”, In *Workshop on Complexity through Development and Self-Organizing Representations (CODESOAR'06)* at GECCO'06 Genetic and Evolutionary Computation Conference. ACM Press, Seattle, Washington, USA.
- Gaber J. & Bakhouya M. (2006 a), “An affinity-driven clustering approach for service discovery and composition for pervasive computing”, In *Proceedings of IEEE International Conference on Pervasive Services (ICPS'06)*, Lyon, France.
- Gaber J. & Bakhouya M. (2006 b), “Mobile agent-based approach for resource discovery in peer-to-peer networks”, In *Fifth International Workshop on Agents and Peer-to-Peer Computing (AP2PC) at AAMAS*, Future University-Hakodate, Japan.
- Bettstetter C. & Renner C. (2000), “A Comparison of Service Discovery Protocols and Implementation of the Service Location Protocol”, Proceedings of Sixth EUNICE Open European Summer School.
- Jini (2001) “Jini Technology Core Platform Specification”, Sun Microsystems, <http://www.sun.com/jini/specs>.
- Zhao W., Schulzrinne H. & Guttman E. (2000), “mSLP-Mesh-enhanced Service Location Protocol”, *International Conference on Computer Communications and Networks (ICCCN 2000)*, Internet Draft draft-zhao-slp-da-interaction-07.txt.
- Wang C. & Li B. (2003), “Peer-to-peer overlay networks : A survey”. *Technical Report*, Department of Computer Science, HKUST. Available: <http://comp.uark.edu/cgwang/>.
- Casati F., Ilnicki S., Jin L., Krishnamoorthy V. & Shan M. (2000), “Adaptive and dynamic service composition in efow”. *Technical Report, HPL-200039*, Software Technology Laboratory, Palo Alto, CA.

Chakraborty D. & Joshi A. (2001), “Dynamic service composition: State-of-the-art and research directions”, *Technical report, TR-CS-01-19*, Department of Computer Science and Electrical Engineering, University of Maryland, Baltimore County, MD, USA.
<http://citeseer.ist.psu.edu/chakraborty01dynamic.html>

Hofmeyr, S. A., & Forrest, S. (2000), “Architecture for an artificial immune system”.
Evolutionary Computation, volume 8, issue 4, pages: 443–473.

Watanabe Y., Ishiguro A., & Uchikawa Y. (1999), Decentralized behavior arbitration mechanism for autonomous mobile robot using immune system. In *Books Artificial Immune Systems and Their Applications*, Springer-Verlag, ISBN 3-540-64390-7.