# Interactive distributed and networked autonomous systems:delegation or participation

**First Author**
Prof dr Frances M.T. Brazier
Intelligent Interactive Distributed Systems Group
Dept of Computer Science
Vrije University Amsterdam
fmt.brazier@cs.vu.nl

**Second Author**
Prof. dr. Gerrit van der Veer
Dept. Computer Science
Open University
gerrit@acm.org


CHI 2009, April 4 – April 9, 2009, Boston, MA, USA
ACM 978-1-60558-247-4/08/04.

**Abstract**
Networks of autonomous systems in dynamic environments are explored within multi-agent system research and development and within autonomic computing. Self-management is core, self-configuration often necessary. Current literature distinguishes two types of assignment: (1)n delegating a system to perform a task, and (2) mandating a system to perform a task. These two types of assignment, however, do not suffice. This paper explores the need for a new type of assignment in networks of autonomous systems*,* *participation*.

**Keywords**
Autonomous systems, hci, actor-agent organisations

**ACM Classification Keywords**
H5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous.

## Introduction

Virtual adaptive organisations of networked autonomous systems are the core of multi-agent systems research and autonomic computing. In both of these fields autonomous systems are designed to manage their own behaviour in interaction with other autonomous systems. They are designed to adapt, to be aware of their own situation (eg to be aware of the abilities and interests of the systems with which they are connected.), to connect to other systems when applicable. Within such networks, autonomous systems collaborate, compete, or co-exist, depending on the

situation. Interaction, in particular negotiation is core to group formation.

Self-configuration, self-healing, self-protection, and self-optimalisation are characteristics of self-management [1]: characteristics used to describe the behaviour of both individual systems and virtual organisations of autonomous systems within autonomic computing. The assumption behind autonomic computing is that self-management frees the user of the burdon of these tasks.

Within multi-agent system research, autonomous systems, ie agents, are designed to interact to achieve the goals they have been designed to achieve. They are designed to be pro-active and reactive, in interaction with other systems, to adapt to new situations. They are designed to be autonomous and to negotiate.

In both autonomic computing and multi-agent systems research individual autonomous systems are assumed to have a certain level of awareness [2]: autonomous systems are designed to be aware of (1) their own identity and abilities, (2) their own goals, (3) their own state, (4) characteristics and roles of other systems in relation to their own (including language of interaction, identity, integrity, trust relationships), (5) interaction/negotiation strategies. A common ground for interaction [3,4] is potentially available.

## 1. Interaction

Interaction between human systems and automated distributed networked autonomous systems, however, is not often considered during design of such systems. Autonomous systems are most often designed to fulfill functional design requirements wrt self-management. Interaction with the user is not of primary concern.

As a result, human autonomous systems are often unaware of the nature of assignment of tasks to one or more non-human autonomous systems. The extent of autonomy of the underlying systems is often unknown.
A car, for example, can be viewed to be an autonomous system that reacts to human user input, autonomously regulating many other automated autonomous systems (some cars have >256 processors).
The human user explicitly delegates the goal of manoeuvring the car as indicated by interaction with the interface (a combination of steering wheel, gear, gas pedal, brakes, etc) implicitly delegating the task of regulating the many other systems involved to the system. The human user has no direct influence on these systems, and has very few means to tract the behaviour of these systems. The human user often only becomes aware of the existence and/or his/her dependency on these other systems when something goes wrong. Tracing the cause of an error in interaction with these systems, for example, is often impossible. The interface of the car is, in fact, designed to hide these interactions from the human user, acting to a certain extent as a mediator.

Autonomous control without human intervention has saved many lives, eg in aerospace and in military applications. It has, however, also cost lives. The balance between the two extremes is not easily determined. Tasks are explicitly assigned to specific systems. Accountability in such domains is taken very seriously. Extensive logs of interaction between underlying systems are made for tracing purposes at a later date, if and when needed.

## 2. Designing human interaction with autonomous networked systems

Designing interaction between (networks of) human autonomous systems and automated autonomous systems requires explicit knowledge of the relationship, the roles and obligations of all systems involved. Shared responsibility or explicit individual responsibility, is an option.

A strict destinction can be made between two different types of task assignment: [5,6, 7] (1) delegation; and (2) mandating as defined below:
Delegation refers to a situation in which an autonomous system (A) assigns the performance of a task to another autonomou system (B) with the expectation and condition that B makes decisions based on rules agreed with or set by A, and according to A's values for outcomes of the decision, A trusts B to perform the task in the way A has prescribed. An example: a manager A delegates the task of planning his/her appointments to his/her secretary B, knowing that B understands the manager''s priorities and knows the importance and relevance of scheduling meetings in order and in time, taking all practical constraints (such as travel) into account..
Mandating refers to situations in which an autonomous system (A) assigns a task to another autonomous system (B) without specifiying how the task is to be performed (eg which decision rules or values are to be used). A trusts B to perform the task to the best of B's ability given B's expertise and values.. E.g., Patient A mandates the task of diagnosing his/her condition to physician B, trusting B to perform the appropriate tests/ask the relevant questions, to evaluate and interpret the meaning and importance of the results, determine the symptoms, and determine the best treatment for the condition diagnosed at a given point in time...

In some cases mediator agents act as the interface between human systems, between humans and (networks of) automated autonomous systems and between (networks of) automated autonomous systems. Given a specific task by mandate, or by delegation, mediator agents can:
(1) Inform the user of a specific situation (providing data without providing the meaning (e.g., medical symptoms); providing information with meaning based on a presumed common ground (e.g., a diagnose);
(2) Interpret this information (providing meaning to data);
(3) Advise the user (suggesting activities based on interpretation of current states and of history of communication);
(4) Act as a broker (negotiating between communicating autonomous systems on actions, including the provision of data and information).

In other cases the user interacts with one or more systems on the basis of the task at hand. Both systems have, to a certain extent, shared knowledge and a shared model. Such shared task models [8] designed for interaction between a human user and an intelligent system, assume that the system's tasks, the roles, the tasks and the responsibilities are understood by both parties..

In more open and less predictable networks this is not possible: behaviour is emergent. The two options for assignment of tasks current literature distinguishes: (1) to delegate a task to an autonomous system, trusting the agent to perform the task according to the actor's specifications or (2) to mandate an autonomous system to independently perform a given task with little or no input. will not suffice.

Designing interaction between groups of human and automated autonomous systems, in which humans are full fledged participants, requires a new type of task assignment: ***participation***. The two options described above for assignment of tasks: mandating and delegating no longer suffice. Participants *interact* with each other, negotiate agreements with each other, defining the terms and conditions for such agreements, and penalties involved if negotiations are not honoured. The tasks to be performed may change during the course of time in interaction with other autonomous systems, given a changing environment. Autonomous systems can learn and adapt, pro-actively changing a course of action with no further consultation with a human autonomous system.

In some systems, however, human systems are explicitly modelled. Within autonomic computing, human systems are modeled as manual managers [9]: autonomous managers that perform a task comparable to that of the manager of an autonomous system in interaction with other autonomous managers, as a manager. Autonomous systems can be

inter-changed: an automated autonomous system by a human or vice versa. The goal of autonomic computing is clearly “about shifting the burden of managing systems from people to technologies” [10] in an evolutionary development.

Tasks are so well-defined that the roles and obligations of each of these systems are clear. There is no real assignment of tasks by the systems to each other: there is collaboration or possibly competition. Users need to be able to “understand” interaction processes, to understand the roles, the division of responsibility and the implications of participation. Different types of information are provided and/or negotiated.,

In more open situations, however, tasks are not so well defined and delegated in advance. Task performance relies on the interaction between autonomous systems, the results of which often can not be predicted. Human users are participants, as are the automated autonomous systems.

A new theory of HCI is needed.

### 3. Discussion

In this paper we focus primarily on the task assignment, in which agents are designed to be participants in dynamic virtual organizations, interacting directly with actors and other agents in dynamic virtual organisations, as participants. Mandating and delegating tasks no longer suffices: new theory is needed to deal with emergent behaviour, to judge the risks involved.

### References

[1] Jeffery O. Kephart and David M. Chess, “The Vision of Autonomic Computing, IEEE Computer, 2003, 36(1):41-50.

[2] Capra, F. The web of life: a new scientific understanding of living systems, Anchor Books, 1997.

[3] Clark, H.H., Arranging to Do Things With Others, CHI conference opening plenary 1996.

[4] Clark, H.H. (1993) Arenas of Language Use. The University. Chicago Press, ISBN: 978-0-226-10781-3, 1993.

[5]Castelfranchi C. and Falcone R.; Towards a theory of delegation for agent-based systems
Robotics and autonomous systems, 1998, vol. 24 ( 3-4), pp. 141-157

[6] Jonathan Gelati, Guido Governatori, Antonino Rotolo, Giovanni Sartor (2002)
Declarative power, representation, and mandate.
In: Trevor Bench-Capon, Aspassia Daskalopulu, and Radboudb Winkels (eds) Legal Knowledge and Information Systems. Pp 41-52. Jurix 2002: The
Fifteenth Annual Conference. Amsterdam: IOS Press, 2002, pp. 41-52.

[7]Giandomenico Majone, "Two Logics of Delegation: Agency and Fiduciary Relations in EU Governance" European Union Politics.2001; 2: 103-122,

[8] Brazier, F. M. T., Jonker, C. M., Treur, J. and Wijngaards, N. J. E., “On the use of shared task models in knowledge acquisition, strategic user interaction and clarification agents”, in: IJHCS, 2000, 52(1), 77-110.

[9] Bert Miller, "The autonomic computing edge: The role of the human in autonomic computing" Developer Works IBM, November 2005.

[10] An architectural blueprint for autonomic computing, IBM Corporation 2003.