# Abstract

Due to the increase in collaborative work and the decentralization of processes in many domains, there is an expanding demand for large-scale, flexible and adaptive software systems to support the interactions of people and institutions distributed in heterogeneous environments. Commonly, these software applications should follow specific regulations meaning the actors using them are bound by rights, duties and restrictions. Common to other works, we use the term *normative open systems* to refer to systems of this kind. The development of systems of this kind can produce important benefits because these systems allow communicating heterogeneous institutions, actors and devices in order to achieve their individual and global objectives. However, there are also some important potential issues that can complicate the analysis, design and implementation of these systems. Most of these common issues are related to interoperability, privacy, and the combination of the individual objectives and restrictions of the system's entities. Software engineering methods and tools are necessary in order to deal with these issues and to guide developers during the development process.

We believe that the Multiagent systems (MAS) technology is a good candidate for the development of normative open systems. MAS technologies are used more and more, not only in academic environments, but also in real industrial applications. Multiagent systems technology has emerged over the last decades as a software engineering paradigm for building complex, adaptive systems in distributed, heterogeneous environments.

This thesis is focused on the analysis and design of normative open systems using MAS technology. Some agent-oriented software engineering methodologies deal with the development of systems of this kind. However, after analyzing to what extent agent methodologies support the analysis and design of these systems, we can conclude that there are some open issues in the topic. Some of these open issues are the integration of the normative context of the system during the whole development process; the lack of guidelines to identify and formalize this normative context; and the lack of validation and verification techniques that ensure the coherence of the final design and the requirements of the system and the coherence between the individual objectives and restrictions of each entity and the global system.

The main contribution of this thesis is a new MAS methodology called ROMAS (Regulated Open Multiagent Systems). ROMAS is focused on the analysis and design processes for developing organizational multiagent systems where agents interact by means of services, and where social and contractual relationships are formalized using norms and contracts. ROMAS methodology defines an agent-oriented development process and provides specific guidelines for identifying and formalizing: (1) the normative context of the system, (2) the entities' communications and interchanges, and (3) both the global behavior of the system and the individual features of each entity.

In ROMAS, agents, roles and organizations are defined through a formal social structure based on a service-oriented open MAS architecture. Here, organizations represent a set of individuals and institutions that need to coordinate resources and services across institutional boundaries. In this context, agents represent individual parties who take on roles in the system; within a given organization (e.g. a company), they can both offer and consume services as part of the roles they play. Beyond this, virtual organizations can also be built to coordinate resources and services across institutional boundaries. *Norms* defined as permissions, obligations and prohibitions restrict the behavior of the entities of the system. *Contracts* are used to formalize the relationships between entities. In our approach, we differentiate between two types of contracts: *contractual agreements* and *social contracts*.

This thesis also presents a modeling tool that support the development of normative open systems designed using the ROMAS methodology. This modeling tool integrates a model checking plug-in that allows the verification of the coherence of the normative context of a system, i.e., the coherence between the restrictions and commitments of each entity and the global specification of the system.

Finally, in order to evaluate the quality and usability of our proposal. We have analyzed the ROMAS methodology regarding its support for the analysis and design of normative open systems. We have also performed an empirical evaluation of the applicability of the ROMAS methodology and tools by means of the analysis and design of several case studies from different domains (e-health, manufacturing, commerce and research). The design of such different case studies has been useful to evaluate different dimensions and uses of the ROMAS methodology.