

University of Massachusetts Amherst ScholarWorks@UMass Amherst

Computer Science Department Faculty Publication
Series

Computer Science

1999

Using Self-Diagnosis to Adapt Organizational Structures

Bryan Horling

University of Massachusetts - Amherst

Brett Benyo

University of Massachusetts - Amherst

Victor Lesser

University of Massachusetts - Amherst

Follow this and additional works at: https://scholarworks.umass.edu/cs_faculty_pubs

 Part of the [Computer Sciences Commons](#)

Recommended Citation

Horling, Bryan; Benyo, Brett; and Lesser, Victor, "Using Self-Diagnosis to Adapt Organizational Structures" (1999). *Computer Science Department Faculty Publication Series*. 66.

Retrieved from https://scholarworks.umass.edu/cs_faculty_pubs/66

This Article is brought to you for free and open access by the Computer Science at ScholarWorks@UMass Amherst. It has been accepted for inclusion in Computer Science Department Faculty Publication Series by an authorized administrator of ScholarWorks@UMass Amherst. For more information, please contact scholarworks@library.umass.edu.

Using Self-Diagnosis to Adapt Organizational Structures *

Bryan Horling, Brett Benyo, and Victor Lesser
Department of Computer Science
University of Massachusetts, Amherst, MA 01003
{bhorling, bbenyo, lesser}@cs.umass.edu

Abstract

The specific organization used by a multi-agent system is crucial for its effectiveness and efficiency. In dynamic environments, or when the objectives of the system shift, the organization must therefore be able to change as well. In this abstract we propose using a general diagnosis engine to drive this process of adaptation, using the TÆMS modeling language as the primary representation of organizational information. A complete version of this paper is at [1].

As the sizes of multi-agent systems grow in the number of their participants, the *organization* of those agents will be increasingly important. In such an environment, an organization is used to limit the range of control decisions agents must make, which is a necessary component of scalable systems. Are agent agents arranged in clusters, a hierarchy, a graph, or some other type of organization? Are the agents' activities or behaviors driven solely by local concerns, or do external peers or managers have direct influence as well? Is communication between agents active, via messaging of some sort, or passive, using observations or engineered assumptions? These and other characteristics define the *organizational structure* of a multi-agent system - the rules which define the roles agents play and the manners in which they interact with other agents in the system.

Clearly the characteristics described above will have an impact on the efficiency and responsiveness of both large and small multi-agent systems. It should also be intuitively clear that the effectiveness of the organization is dependent

on the agents, environment, and goals involved in the system. The problem then, is how to derive and maintain such a structure given a particular situation. The term Organizational Self-Design (OSD) has been used previously to describe the general technique of employing the members of a multi-agent system to generate or adapt their own organizational structures at runtime. Earlier work in this area tended to focus on adapting specific qualities of the organization, such as task allocation or load balancing. Organizational structure generation has also been proposed as arising from local, global, and hybrid perspectives. In this abstract we outline a more general approach, using diagnosis, to detect deficiencies in the organizational model and assist in the creation of solutions to those deficiencies; the eventual goal being to create a reusable organizational adaptation engine. We will show how a general diagnosis engine, coupled with a powerful representation of that organization, can be used to effect change in a wide range of characteristics.

In the architecture we propose, critical components within the agent, such as those responsible for problem solving, negotiation and scheduling, obtain the vast majority of their information from an organizational design layer. This layer abstracts and filters elements of the operating environment in a manner consistent with the agent's role in the organization. The abstraction is composed of one or more information sources, such as TÆMS structures, capable of encoding the various aspects of the organization. To permit adaptation, the organizational design layer is maintained by a diagnostic subsystem, which attempts to repair faults and inefficiencies by adjusting elements of the organizational structure. This diagnostics process can itself be driven by a number of sources, including observations of the environment, conditions monitored within the agent, and discourse with other agents. The direct effects of these diagnoses will typically (but not necessarily) take place within a relatively small group of agents, so one can think of this technique as a search for the correct organization through local adaptation.

The range of information that comprises an organizational structure can be quite broad. It is our opinion that there is no single, comprehensive set of characteristics that

*The effort represented in this paper has been sponsored by the Defense Advanced Research Projects Agency (DARPA) and Air Force Research Laboratory, Air Force Materiel Command, USAF, under agreement numbers F30602-99-2-0525 and F30602-97-1-0249, and by the Department of the Navy, Office of the Chief of Naval Research, under Grant No. N00014-97-1-0591. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Defense Advanced Research Projects Agency (DARPA), Air Force Research Laboratory, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon.

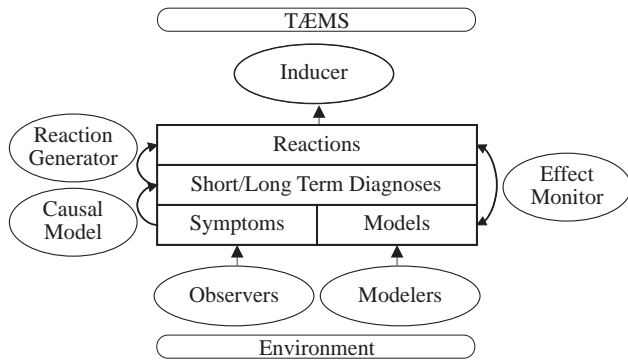


Figure 1. High-level architecture of the diagnostics subsystem.

might make up *the* definition of an organizational structure. Instead, the set is dependent more on what alternatives are possible within a particular multi-agent system and which of those alternatives can have an impact on the system's behavior and effectiveness. To address this, we use TÆMS models as our organizational representation, which are flexible enough to model a wide range of characteristics.

Figure 1 shows the architecture of the diagnostics subsystem we currently employ. It uses a blackboard-based design, separating the process into three distinct layers: symptoms, diagnoses, and reactions. This type of system offers several advantages. It promotes a clear chain of reasoning, since the diagnoses supporting a given reaction can easily be identified, as can the symptoms that support a particular diagnosis. The layers also clearly define the separation of responsibilities, which enables arbitrary components or even remote agents to asynchronously use and add to elements on the blackboard.

The lowest level of the blackboard contains symptoms, elements that contain observations about such things as the environment, agent activities and commitments. This level is maintained by both observers, which passively collect data, and modelers, which use learning techniques to make predictive models. The diagnosis level contains descriptions of possible faults or inefficiencies derived from the data on the symptom level. These diagnosis are produced with the use of a causal model, which will be described below. At the highest level, a set of potential reactions to the diagnoses are generated, which may be selected from by the inducer to effect change in the organizational design layer. A feedback loop is maintained by the effect monitor, which determines if changes made were successful and effective.

Diagnosis is a well-researched field, with many different methods and techniques already available to the system designer. Our goal was to use a technique that offered great flexibility in the information it could use and the diagnoses it could generate, without sacrificing subject scope or domain independence. It is not clear from the outset, howev-

er, that any single diagnostic technique (e.g., model-based, symptom-directed, collaborative) is suitable for the entire range of faults exhibited by multi-agent systems. It was therefore desirable to use a system or framework capable of incorporating different diagnostic techniques.

Expanding on our previous work in the area, we chose to organize our diagnostic process using a causal model. The causal model is a directed, acyclic graph that organizes a set of diagnosis nodes. Each node in the causal model corresponds to a particular diagnosis, with varying levels of precision and complexity. As a node produces a diagnosis, the causal model can determine what other, more detailed, diagnoses may further categorize the problem. Within the diagnosis system, the causal model then acts as a sort of road map, allowing diagnosis to progress from easily detectable symptoms to more precise diagnostic hypotheses as needed. Individual nodes can also use internally whatever diagnostic techniques are most appropriate for their fault, so in this architecture we can make use of a variety of different methods, given the types of failures they best address.

Adapting organizations at runtime becomes important when the environment, goals, or participants of a system are liable to change. Several specific techniques have been offered by previous work in this area; we propose a more general solution to the problem by organizing such activity under the umbrella of diagnosis. A general diagnostic engine such as that shown in this paper is capable of detecting and diagnosing a variety of faults and inefficiencies, which can be used to drive organizational change. The organization itself is represented using models, such as TÆMS structures, which abstract the relevant portions of agents' capabilities and interactions in a way that facilitates both its use by agent control components and its adaptation by diagnosis. In this architecture, the methods driving change, and the characteristics affected by adaptation, can then be simplified to general techniques updating a domain independent representation, which can be reused from one system to the next.

A system using the architecture outlined in this paper has been implemented and tested using scenarios from the producer/ consumer/ transporter domain. In this system, a set of four heterogeneous factories and two transporters were given two distinct production phases, differentiated by what needed to be produced, and which factories could manufacture the raw materials. As shown in [1], the diagnostics system successfully converged on a suitable organization for the first phase, and adapt it to meet the needs of the second.

References

- [1] B. Horling, B. Benyo, and V. Lesser. Using Self-Diagnosis to Adapt Organizational Structures. Computer Science Technical Report TR-99-64, University of Massachusetts at Amherst, November 1999 [<http://mas.cs.umass.edu/bhorling/papers/99-64/>].