

Enhancing safety in the event of catastrophic failure: the case for continual use systems

S. A. McComb¹ & A. V. Deshmukh²

¹*Isenberg School of Management,
University of Massachusetts Amherst, USA*

²*Department of Mechanical and Industrial Engineering,
University of Massachusetts Amherst, USA*

Abstract

Ensuring safety in the event of catastrophic failure has received a significant attention in recent years. Organizations are putting systems in place to counteract man-made and natural disruptions to critical infrastructures (e.g., health care networks, transportations systems, electric grids, telecommunication networks) and business value chains to keep the population safe and minimize the impact on the economy. Many of these systems, however, are designed for emergency use only. This paper presents evidence that suggests this approach may not provide the safest and most effective results. Specifically, our research demonstrates that people do not respond well to an environmental disruption when the response procedures are not well established and part of the routine practices. Thus, we suggest incorporating continual use, robust systems that can be used during normal operations and function effectively during disruptions as an alternative means of preparing for extreme event failures. We present an example of an information infrastructure that allows the realization of such continual use processes. We conclude with a discussion of future research needed in this area.

Keywords: environmental uncertainty, continual use systems, robust systems.

1 Introduction

For many, safety in day-to-day life is the norm that is taken for granted. A continuously safe environment, however, is unrealistic. Catastrophic failures will occur. These failures may be natural (e.g., hurricanes) or man-made (e.g.,



terrorist attacks). Moreover, these failures range in size from a relatively small (e.g., single house fire) to unfathomably large (e.g., Asian tsunami). In order to recover a safe state after a catastrophic failure, response systems are necessary.

These response systems organize the collaboration required for interdependent activity among independent individuals and groups of individuals. For example, in the event of a multi-car accident on an expressway, police officers, firefighters, emergency medical technicians, emergency room personnel, media providers, affected family members, towing companies, among others must all interact to handle the crisis. Under normal day-to-day activity, these collaborators may not routinely interact. Thus, a response system must be able to coordinate the activities occurring among the various parties that may have established relationships. The level of coordination required is dictated by the amount of environmental uncertainty perceived by the individuals involved.

Perceptions of the environment are influenced by the content and amount of information available to them about the environment, which, to some degree, are controllable Huber et al. [1]. One plausible means of controlling information load is through the use of routines. Specifically, Gittell [2] found that routines including protocols used by all players, tasks to be completed, sequence requirements, and, most importantly, links among required actions, improve performance by increasing the relational coordination among interdependent individuals and groups of individuals. These effects increase when uncertainty increases.

A current trend in emergency planning is to create systems that are utilized only in crisis settings. These special use systems, however, may not be adequate when unusually disruptive events occur. Zellmer-Bruhn [3] found that these types of events might cause individuals to revert back to a comfortable routine because the affected individuals must figure out how to survive. Under these circumstances, a well-rehearsed routine would be of the most use. We illustrate the need for both a routine and the ability to apply the routine with experimental data. We use this evidence to make a case for the design of continual use, robust systems that allow relevant parties to interact under normal operating conditions and also during crisis.

2 Experiment

We now present experimental evidence that highlights the importance of routines during emergencies. An experiment was conducted to ascertain the effects of environmental uncertainty on team collaboration. Participants were 81 male and 117 female undergraduate Isenberg School of Management students at the University of Massachusetts Amherst. Their ages ranged from 17 to 31 ($M=19.6$, $SD=1.33$). They participated in the experiment in exchange for extra course credit and a one in five chance to win \$50 (based on team performance). The students were randomly assigned to 66 teams of three individuals each. One half of the teams collaborated via a computer-mediated format and the other half interacted face-to-face.



The experimental task was based on the research of Earley [4], Bachrach et al. [5] and Steele-Johnson et al. [6]. In the experiment, teams were asked to complete an assignment problem. Specifically, they were required to heuristically create a 7 day, 12 hour work-force schedule with varying staffing levels by day. Ten individuals, who earn from \$5 to \$10 per hour, had to be assigned to two hour time slots. Each of the team members also had unique requirements that induced collaboration on the task, as follows: (1) no individual could work more than 10 hours per day and 50 hours per week, (2) no individual could work less than four hours per day and 30 hours per week, (3) all staffing levels had to be met exactly. If any of these requirements were not met, the team was assessed a financial penalty. The teams were given a performance goal to achieve the lowest cost schedule.

The sessions lasted 49 to 147 minutes ($M=83.5$, $SD=20.6$). To initiate environmental uncertainty, the teams were interrupted after 20 minutes of operation. This timing was selected because it allowed a team enough time to devise an approach to the process and begin assigning workers to time slots. At the interruption, one worker was promoted from a \$5 per hour worker to a \$10 per hour worker. This interruption required that the teams reevaluate their schedule and revise it as necessary.

The experimental results are shown in Table 1. Each team's experimental session was reviewed to determine if they developed a routine for solving the task and if they were able to recover after the interruption. We compared the performance of (1) those with and without routines, (2) those who recovered with those who did not recover, and (3) those that had a routine *and* were able to recover with those that did not exhibit both behaviours. As can be seen in the table, teams with routines ($p=0.12$) and teams able to recover ($p=0.07$) perform more effectively than those in the opposite condition. Teams meeting both conditions, however, demonstrated the most significant difference between conditions ($p=0.05$). Thus, our results indicate that both an established routine and the ability to invoke it after an environmental disturbance are critical for a team's ability to perform at the highest levels.

Table 1: Experimental results.

	Routine		Recovery		Routine and Recovery	
	Yes	No	Yes	No	Yes	No
N	53	13	43	23	41	25
Mean	3126	3229	3103	3226	3098	3225
SD	200	267	148	293	145	285
p-value	0.12		0.07		0.05	

Note: Low scores indicate better performance.



3 Continual use systems

From the previous discussion it is evident that continual use systems are necessary to aid individuals and groups who intermittently face environmental uncertainty. Moreover, these systems must allow individuals and groups to collaborate across organizational boundaries during emergencies. Several key issues make the continual use systems different than the normal use systems or systems used only during emergencies:

- These systems need to be flexible and be able to handle uncertainties faced during emergency situations.
- These systems need to be easy to use and efficient during normal operating conditions.
- These systems need to allow inclusion of new participants and allow existing team members to leave the system during emergencies.
- These systems need to allow discovery of new resources and potential participants during emergencies or under changing operational conditions.

3.1 System requirements

A multi-organizational continual use system needs to facilitate flexible process flows across all user organizations. Process flows can be defined as procedures or routines adopted to achieve certain goals or complete given tasks. One can envision process flows as a series of steps that need to be performed in order to meet the overall goals. It is easy to see that in the case of emergency situations the same goals may need different tasks or participants in order to be accomplished. Hence, the underlying infrastructure of any continual use system needs to be able to allow execution of flexible processes and inclusion of new participants. Given the proliferation of computers and information systems in government and business functions, most continual use systems will be deployed within existing information systems used by these organizations. Hence, any multi-organization continual use system must enable flexible process flows within current computational environments.

Four essential elements of the computational framework that could support continual use systems are their ability to: (1) integrate different tools and platforms easily; (2) abstract computational environmental data from the end-user; (3) provide uniform representation schemes for tools, tasks, linkages, and parameters; and (4) allow reusability of information. In addition to a computational framework, the system must also include a distributed decision framework for managing complex, flexible process flows. Such a system will (1) allow state dependent, dynamic decision making to occur during execution (domain based and domain independent); (2) include iterative, distributed solution search techniques; and (3) provide diagnostic capabilities for process monitoring.

In addition to the overall system requirements, the computational infrastructure must also meet the following end-user requirements. The end-



users should be able to: (1) easily integrate software tools on local platforms; (2) create flexible process flows on-the-fly; (3) monitor and provide guidance to the search process. Additionally, the computational architecture should allow: (1) analysis codes to run on different platforms; (2) plug-in-play support for distributed decision protocols; (3) dynamic updating of domain knowledge; (4) instant, secure and error free data transfer; (5) integration with other frameworks; and (6) automatic retrieval and reuse of archived data.

3.2 Computational architecture for continual use systems

Figure 1 shows an instantiation of a computational architecture that meets the abovementioned requirements of continual use systems. The overall architecture is based on the concept of autonomous agents, where each participant or process is encapsulated as an agent in the system. The process flow is defined by the series of interactions between different agents in the environment. The environment shown in Figure 1 can be divided into two types of activities: (1) the activities related to the creation and monitoring of agents and (2) the agents or the different *nodes* in the continual use system. The first set of activities allow decision makers involved in the continual use system to create their proxies in the computational domain, using the *Tool Agent Creator* and the *Collaboratory* via the *Browser-based Front-End*. The second part of the environment is related to the different agents representing the decision makers or process steps in the continual use system.

Each agent or a node is managed by a *Tool Service*. Detailed representation of a tool service is shown in Figure 2. The tool service is responsible for running the codes, and managing incoming and outgoing data. The tool service accepts data packets from other servers and routes them to the proper code, which is accessed via its analysis package. The tool service is also responsible for managing large data transfers for which the data packets only contain a reference. The tool service is responsible for the life cycle of a computation and as such only works with parameters, data and type. Any higher level abstractions are handled by the tool agent.

The tool agent is a computational object instantiated by the tool service to manage the higher level relationships of the analysis. The tool agent has the following responsibilities:

- Manage the *rules* associated with each step of the process.
- Determine the location of the next step in the overall process plan.
- Coordinate with other tool agents
- Generate routing information for the tool service.

Each tool agent uses decision rules to guide its actions. The decision rules are the dynamic decision making component of the continual use system. The rules allow the environment to make decisions dynamically based on the state of the system during runtime. Tool agents manage the rules and coordinate the decision making process. The rules capabilities are declared in a *rule map*, which the tool agents use to route the data from the tool and to coordinate the results of the decisions. There are different rule classes that handle the various dynamic



characteristics of the problem. They are the user rules, system rules and monitor rules. The user rules are used to choose between alternative links in the process path by examining the state of the system. These rules use domain dependent knowledge to select the appropriate action based on the data thus adapting to the current analysis state. Once the process decision is made regarding what link to use, the appropriate computational location can then be selected by applying the system rules. System rules are similar to user rules. They do not distinguish between links but between servers. The system rules are deployed across an entire process map and can be changed based on the topology of the servers (such as geographic location and cost). The system load balancing is implemented as a system rule. Monitoring rules are used to globally observe data and collect aggregate statistics about the overall solution. Typically the system and monitoring rules are applied to an entire scenario and can provide complementary information for the user and rules.

Event scenarios are generated by the process generation module, which takes the user requirements and forms a process map. The process map is created by searching for a valid path through pre-existing links or matching parameters. Any ambiguities in link selection can be resolved by applying user rules or by querying the user directly. The lookup module provides the search interface between the computational environment and the available resources.

This computational environment allows multiple users to interact in an efficient manner, during normal operating conditions and in the event of emergencies. The autonomy and modularity offered by the agent representation allows the system to be robust to failures in the event of catastrophes or allow new entities to interact with the system.

3.3 Other considerations and future research opportunities

Continual use systems have costs associate with them (e.g., development costs, implementation costs). Moreover, they may degrade performance by making it very cumbersome to complete work at a normal pace. Thus, system design is of the utmost importance. Systems that make day-to-day tasks cumbersome will create inefficiencies in operations. Under these circumstances, users may find ways to shortcut the system, thereby undermining the purpose of a continual use system.

Designers must ascertain what system features need to be incorporated for continual use and what, if any, features can be introduced in times of emergency without system degradation. For example, when the environment becomes highly uncertain, coordinators may be necessary to ensure integration across all interdependent players Daft [7]. For example, as reported in the 9/11 Commission Report [8], on September 11, 2001, New York City 911 Operators and the Fire Department Dispatchers were the primary point of contact for the individuals in the World Trade Center. They were not informed by the Fire Department that full evacuation was necessary and that roof top rescues were not feasible. This lack of integration resulted in 911 operators telling civilians not to self-evacuate and not telling them that descending was the only possible way for evacuation. Under these conditions, lateral processes are necessary to coordinate



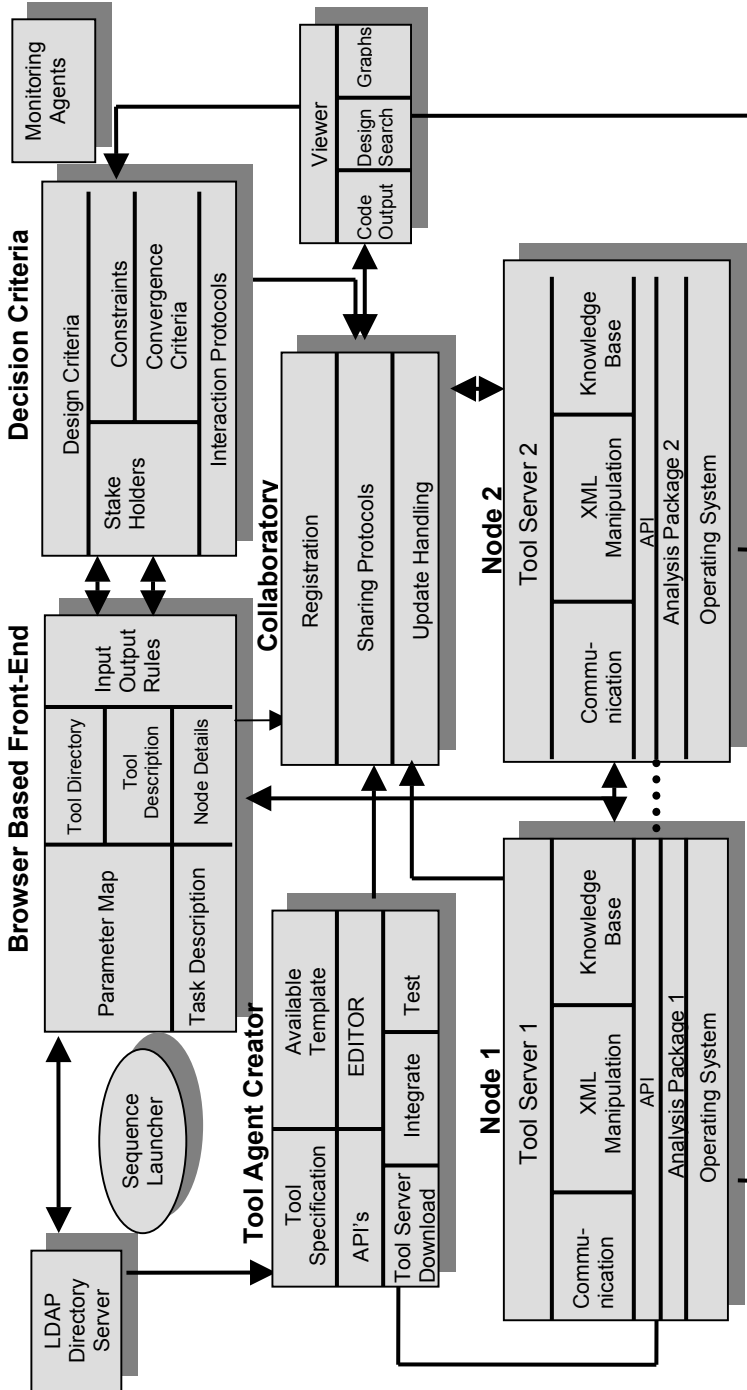


Figure 1: Computational architecture for continual use systems.

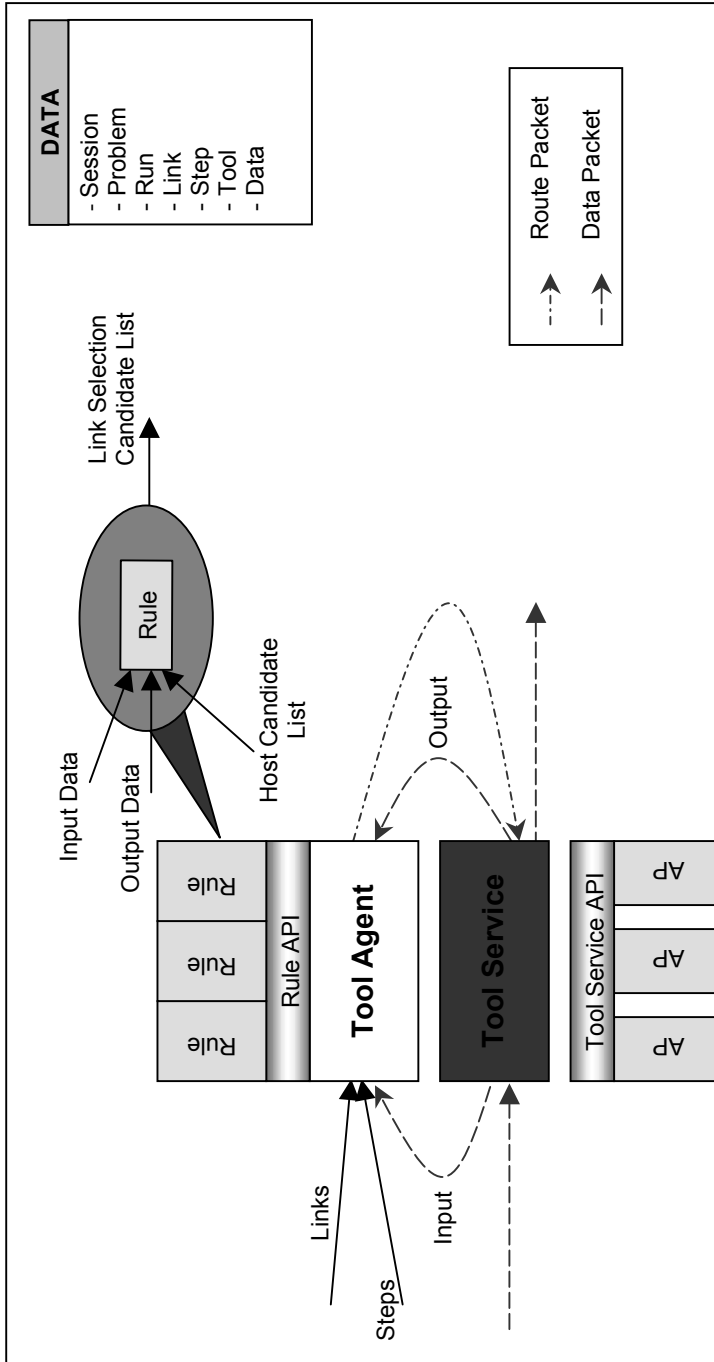


Figure 2: Tool server design.

information exchange and decision-making across decentralized entities Galbraith [9]. Future research needs to determine how to design efficient continual use systems that do not add cost and degrade performance under normal circumstances, but are effective in times of crisis.

Acknowledgements

This research is supported in part by the National Science Foundation, Grants no. 0092805, 0122173; NASA, Grant no. NCC2-1353 and the Collaboration and Knowledge Management program of the Office of Naval Research, Grant no. N000140210535. The authors would like to thank Ralitzia Patrashkova, Tim Middelkoop and Seamus Gahan for their help in preparing this manuscript.

References

- [1] Huber, G.P., O'Connell, M.J. & Cummings, L.L., Perceived environmental uncertainty: Effects of information and structure, *Academy of Management Journal*, **18(4)**, pp. 725-740, 1975.
- [2] Gittell, J.H., Coordinating mechanisms in care provider groups: Relational coordination as a mediator and input uncertainty as a moderator of performance effects, *Management Science*, **48(11)**, pp.1408-1426, 2002.
- [3] Zellmer-Bruhn, M.E., Interruptive events and team knowledge acquisition, *Management Science*, **49(4)**, pp. 514-528, 2003.
- [4] Earley, C., Self or group? Cultural effects of training on self-efficacy and performance. *Administrative Science Quarterly*, **39(1)**, pp. 89-108, 1994.
- [5] Bachrach, D., Bendoly, E. & Podsakoff, P., Attributions of the "causes" of group performance as an alternative explanation of the relationship between organizational citizenship behaviour and organizational performance. *Journal of Applied Psychology*, **86(6)**, pp. 1285-1295, 2001.
- [6] Steele-Johnson, D., Beauregard, R., Hoover, P. & Schmidt, A., Goal orientation and task demand effects on motivation, affect, and performance. *Journal of Applied Psychology*, **85(5)**, pp. 274-284, 2000.
- [7] Daft, R.L., *Organization Theory and Design*, South-Western College Publishing: Cincinnati, Ohio, 2001.
- [8] *9/11 Commission Report*, W.W. Norton and Company: New York, 2004.
- [9] Galbraith, J.R., *Designing Organizations*, Jossey-Bass Publishers: San Francisco, California, 1995.

