Information Technology: System Dynamics Modeling of an IT Major Incident Resolution Process

John Voyer, Ph.D.
Andrea Cahill, MBA
Kathryn Laustsen, MBA
Benjamin Philbrick, MBA

University of Southern Maine

Abstract

We did a system dynamics analysis of a significant IT problem—poor handling of Critical Incidents—at a medium-sized health care organization. We used the usual system dynamics process of describing the process verbally, identifying reference modes, and creating a dynamic hypothesis. From that, we developed a quantitative model, which was a variant on the familiar Project Model. Using the model, we then tested various policy options for staff allocation and their coordination. The conclusion was that, even though the organization could improve work quality by using fewer staff members, poor coordination—regardless of number of staff—wiped out gains in quality, productivity, and speed. We end by recommending ways to improve coordination.

Problem Description

We did a system dynamics analysis of a significant IT problem at a medium-sized health care organization, using the usual system dynamics process of describing the process verbally, identifying reference modes, creating a dynamic hypothesis, developing a quantitative model, and testing various policy options using the model.

Marshall's Promontory Healthcare (MPHC) employs approximately 800 people from Caribou, Maine to Syracuse, New York. The business consists of two primary models: clinics, which provide patient care, and health insurance plans. There are nine clinics in the MPHC family and the health plan business operates two plans.

The Information Technology Department at MPHC employs approximately 40 people. The department receives many calls, ranging from password resets to issues that are more complex affecting a larger portion of the organization. The department calls the latter Major Incidents. Two types of Major Incidents affect the IT Department:

- High
  - Affects, or makes unavailable, a single non-core service/business process
  - The disruption affects 5-19 patients or members
  - An incident prevents 2-99 employees from working
- Critical
  - Disruption is System wide or Companywide or line of business wide
  - Disruption affects 20 or more patients or members
  - An incident prevents more than 100 employees from working
  - There is a risk to public safety or reputation

Example: For over nine months during 2014, there was a memory leak issue within Internet Explorer. Though this does not sound like a huge problem, MPHC's Electronic Health Record is cloud based. Having IE crash quite often for practice users affects workflow and patient service. In another Major Incident example, when the credit card scanning machines do not work in the Virtual Desktop Infrastructure (VDI) environment, checkout personnel are unable to receive payment. By not documenting the details, users, adaptations, root cause, and resolution, MPHC's IT department cannot resolve similar issues quickly if they reoccur.

The problem MPHC has been having is that the quality of the department's work starts out high, but then drops, and the level of coordination among staff assigned to the major incidents is, after a brief spike, quite low.

Major Incident Process Flow

MPHC has a published process flow for dealing with Major Incidents, which we show in Figure 1. For purposes of the present paper, there are three crucial points in this process:

1. After identifying the existence of a Major Incident, the IT staff is supposed to coordinate its efforts and develop a recovery plan.
2. From its recovery plan, the staff is supposed to create and implement a temporary workaround plan.
3. After implementing the workaround, the staff continues working to resolve the Major Incident, ultimately stabilizing it.

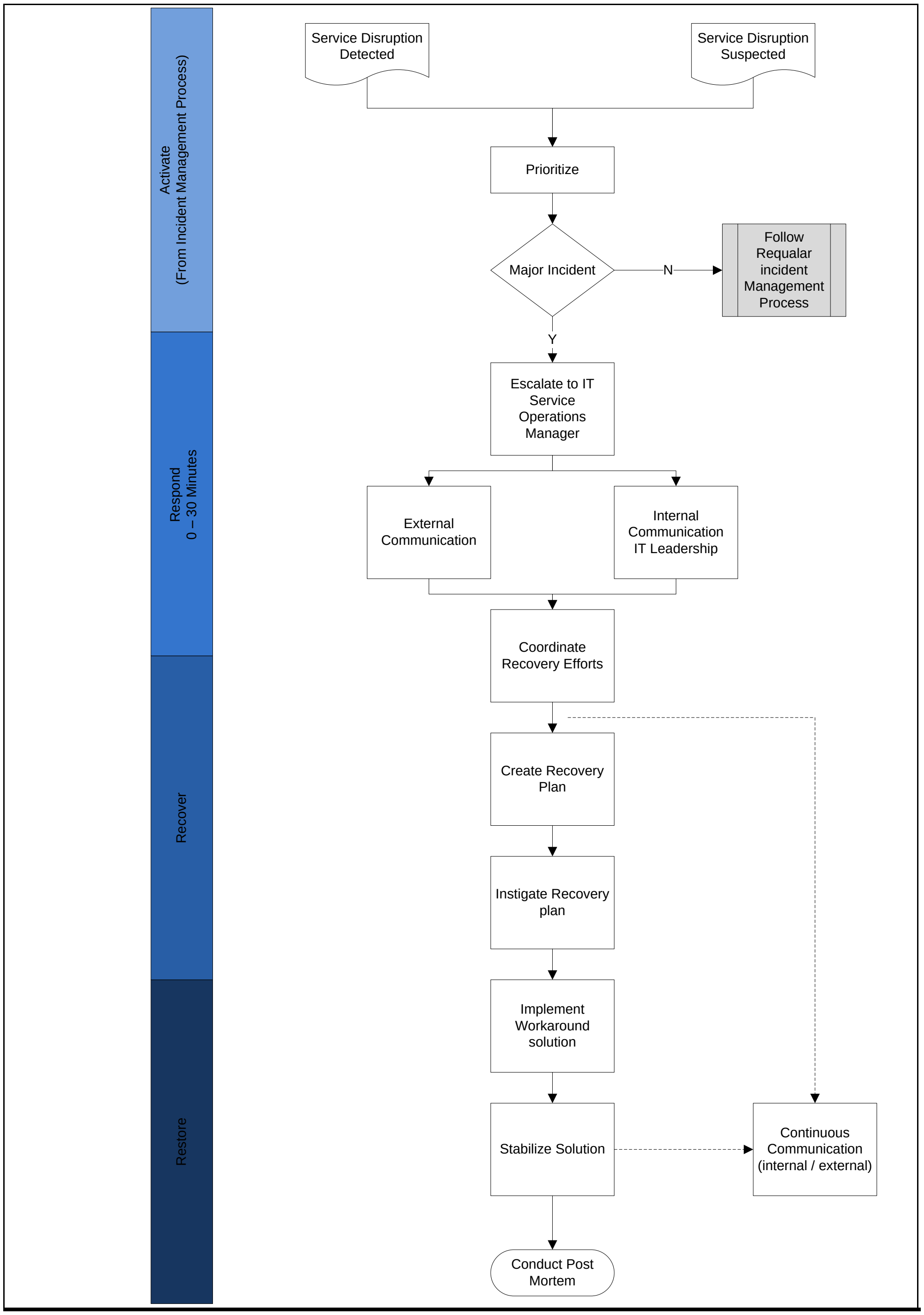

Figure 1. MPHC's Process Flow for Major Incident Resolution

Year	Number of Major Incident Tickets
2012	22
2013	17
2014 (est.)	14

Table 1. Major Incident IT Tickets at Marshall's Promontory Health Care

Year	Average Number of Calls/Ticket	Average Length of Ticket (days)
2012	12.18	12.49
2013	17.59	17.5
2014 (est.)	14.44	6.8

Table 2. Characteristics of Major Incident IT Tickets at MPHC. (Data for 2014 are estimates from 9 months of actual data.)

Ticket Data

MPHC's IT Department uses a piece of software called TrackIT to track all incoming tickets to the IT Service Desk. The department polled those data for Major Incident Tickets, which we show in Table 1. Our model (discussed later) used an average number of Major Incidents from 2012/2013, as they were complete-year data. This number is 19.5 tickets per year, which is equal to .053425 Major Incidents per day (days is the default unit of measure in the model), or roughly one every three weeks.

Data were not available about number of employees working on an incident, whether the department implemented a workaround, and about the efficiency/coordination of the employees working on the incident. However, MPHC's IT department provided the information about the Number of Calls by Major Incident and the Duration of each Incident, i.e. the time between the initial report and the implementation of the resolution. We averaged those data, which we show in Table 2.

Reference Modes

Based on interviews and the personal experience of on member of the modeling team who is also a staff member of the IT Department at MPHC, we can infer the following reference modes for this problem.

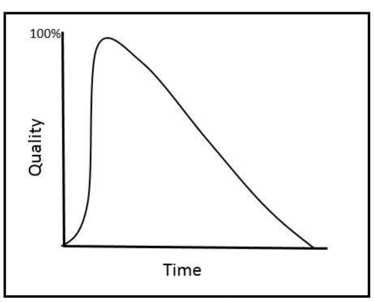

Figure 2. Quality of IT Work

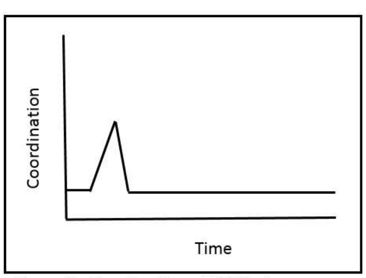

Figure 3. Coordination of IT Work

Quality of IT Work (Figure 2)

Quality of response to Major Incidents starts high, but as time passes, the quality goes down, because the number of employees working on the issue decreases as other projects and priorities come along, and as there is lack of resolution to the original Major Incident.

Coordination of IT Work (Figure 3)

This stays and remains low, as the IT employees recreate the issue and then go off in different directions to work in silos to find a resolution. Because of this atomized approach, they make many changes at once, which increase the difficulty of pinpointing the actual resolution (or resolving subsequent problems, if other things are no longer working). There is a spike in the mode to represent staff checking in with one another, which is brief and often unlikely.

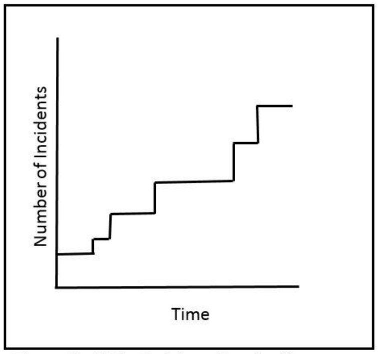

Figure 4. Major Incidents Resolved

Major Incidents Resolved (Figure 4)

This represents the resolution of Major Incidents over time, and shows how the length of resolution time varies for each Major Incident.

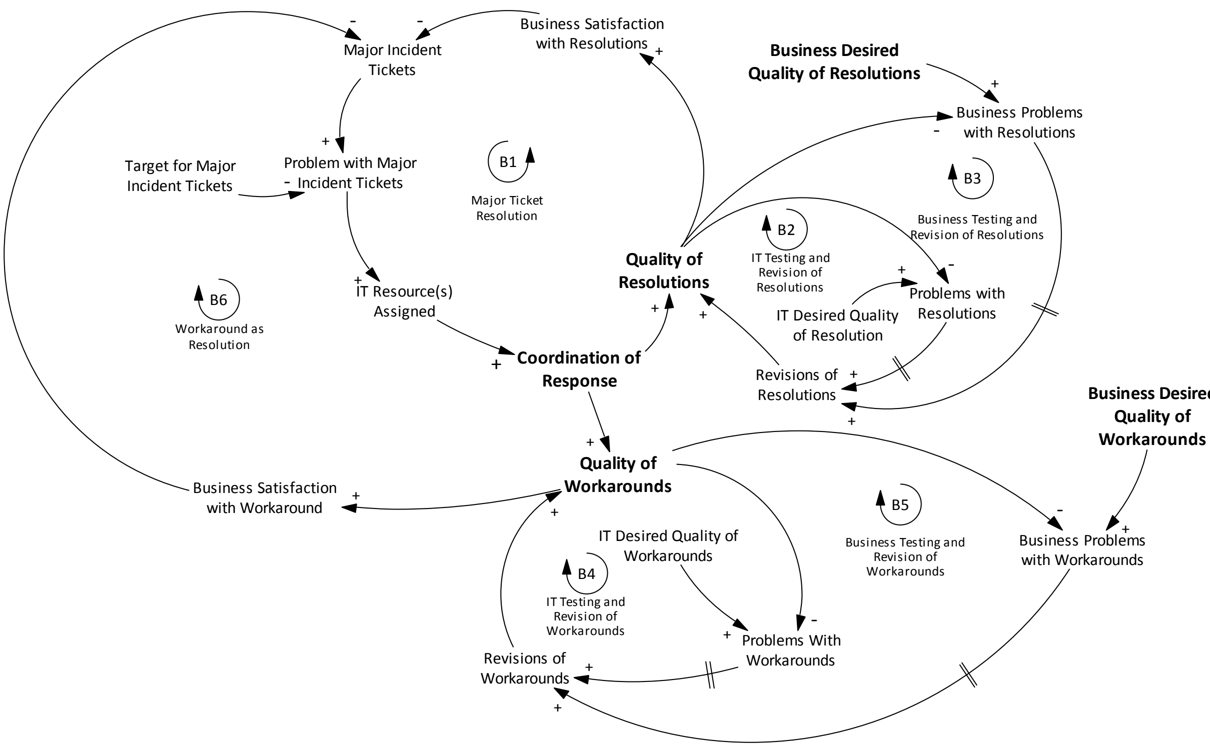

Figure 5. Causal Loop Diagram and Dynamic Hypothesis

Dynamic Hypothesis

We present our dynamic hypothesis in Figure 5. We placed Quality of Workarounds and Quality of Resolutions in boldface centrally in the causal loop diagram, since both affect the time it takes the IT department to present a workaround or resolution to the business (which is the name the IT department gives to its internal clients at MPHC). In addition, the Coordination of the Response from the IT department (ostensibly a key early element of its process flow) has a direct correlation to Quality of the Workaround and Quality of the Resolution, so that is also in boldface in the center of the diagram. Lastly, without timely business involvement in both a workaround and a resolution, it is difficult for IT to know if it has proposed an adequate workaround, or when it has resolved the Major Incident.

Loops

There are six loops in the Causal Loop Diagram, all balancing.

Loop **B1 - Major Ticket Resolution**, describes the process that the IT department follows, by policy, to resolve a ticket. There is one exogenous policy input to this loop, Target for Major Incident Tickets, which is the target for completion of any Major Incident ticket.

Loop **B2 – IT Testing and Revision of Resolutions**, describes the process within the IT department relating to the quality of and the need to revise possible resolutions, ultimately to meet quality standards the department sets.

Loop **B3 – Business Testing and Revision of Resolutions**, describes how the IT department relies on the business to test each possible resolution and to sign off in agreement that the resolution is fully functional. This loop has an exogenous input of Business Desired Quality of Resolutions, as often the business's perception of the quality of the resolution differs from that of the IT Department.

Loop **B4 – IT Testing and Revision of Workarounds**, describes the workaround process that *sometimes* occurs if the IT Department cannot resolve the root cause to the Major Incident fast enough. In this case, if there is a way for the business to continue to serve patients and members, even if the workaround is lengthy, then the department recommends a workaround as a stopgap from turning patients and members away at the door. This loop has an exogenous input of IT Desired Quality of Workarounds, which creates the familiar goal-gap situation between desired and actual quality of the workaround.

Loop **B5 – Business Testing and Revision of Workarounds**, describes the same workaround process just described, only this time from the perspective of the business. In this case, once the IT Department has proposed a workaround, the business must test and sign off on whether the workaround is sufficient for it to continue to do business. This loop has an exogenous input of Business Desired Quality of Workarounds, as the Quality of the Workaround, the Business Desired Quality of the Workaround and the IT Quality of the Workaround may all differ, due to expectations and/or perceptions.

The final balancing loop, **B6 – Workaround as Resolution**, describes a situation where the workaround built for the Business becomes the resolution. In this loop, the business accepts that the workaround will suffice as the resolution, possibly because IT has been unable to determine a resolution, or, possibly, because the workaround is an improvement to the business flow.

Note the critically central importance in our dynamic hypothesis of Coordination of the Response from the IT department. This variable feeds all six loops, and we hypothesize that the issues the IT Department at MPHC experiences result from *poor* Coordination of the Response, mostly (we submit) caused by too many staffers thrown at the Major Incident in a "siloed," atomized way. This will feature prominently in our model, to which we now turn.

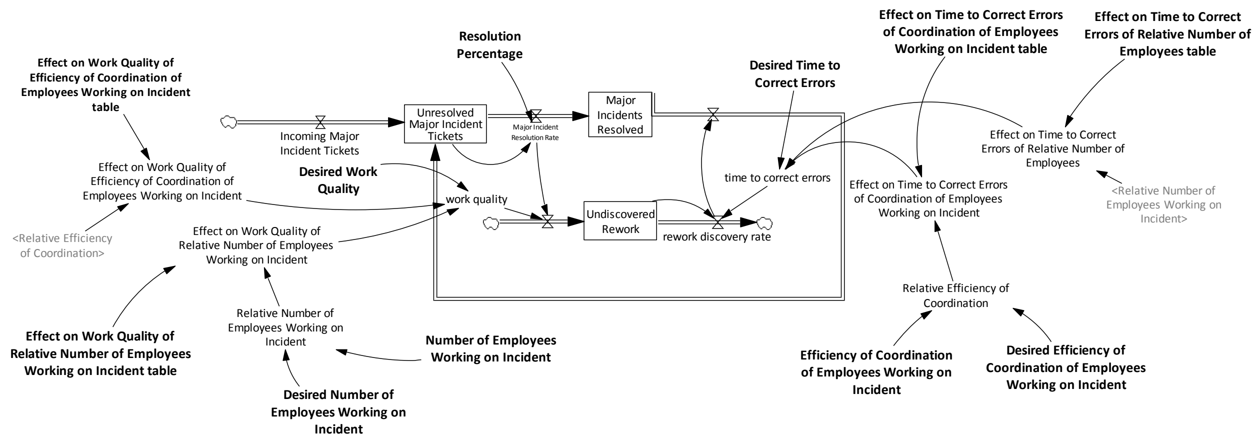

Figure 6. Full Stock-and-Flow Model

The Model Explained

We show our full stock-and-flow model in Figure 6. We believe that this situation is a variation of the familiar "Project Model." In that model, there is a stock of "Work to be Done," whose contents move into a stock of "Completed Work." However, a percentage of items ostensibly moved to the stock of "Completed Work" are actually "Undiscovered Rework." These items appear finished, but actually were done so poorly that, once their flaws are discovered, will have to return to the stock of "Work to be Done." The project ends when these iterations end and all the work truly is finished.

A major difference in the situation described at MPHC, however, is that the "Work to be Done" never actually ends. It is not a finite stock of work, as in the Project Model. It is a stock *continuously replenished* as new Major Incidents crop up. However, there are some similarities. As in the Project Model, not all the Major Incidents are resolved on their first pass through the process; the IT staff will need to redo some of them as the flaws in their workaround or resolution emerge. In another similarity to the Project Model, what drive the problem under study here are two things:

1. The quality of the work done in the first place.
2. The speed at which the IT staff discovers its errors.

We believe that at MPHC, poor coordination of staff drives both of these key variables.

For Work Quality, we chose to model this problem using two table functions:

1. The input for the first table function is the ratio of employees assigned to work on a problem and the *desired* or *appropriate* number assigned to that work (on the theory that the siloed, atomized response increases as management assigns a larger number of staff members to the Major Incident). This is a downward sloping function, where quality declines as management deploys more employees to the Critical Incident.
2. The other table function has the input of ratio of *actual* efficiency of coordination and *desired* efficiency of coordination. Coordination is an explicit element of the organization's resolution process, yet we believe that the organization does not practice good coordination. This is an upward sloping function, as quality improves as coordination improves.

For Time to Correct Errors, we also chose to model this problem using two table functions:

1. The first table function has the input of the ratio of *actual* efficiency of coordination and *desired* efficiency of coordination. As mentioned earlier, coordination is an explicit element of the organization's resolution process, yet we believe that the organization does not practice good coordination. This is a downward sloping function, where the time required to correct errors declines as management is more efficient in coordinating the employees assigned to resolving a Critical Incident.
2. The input for the second table function is the ratio of employees assigned to work on a problem and the *desired* or *appropriate* number assigned to that work (on the theory that the siloed, atomized response increases as management assigns a larger number of staff members to the Major Incident). This is an upward sloping function, where the time required to correct errors increases as management deploys more employees to the Critical Incident.

Simulation Run	Employees	Coordination
Base	6	100%
Additional employee	7	100%
Reduced efficiency	6	80%
Suboptimal on both	7	80%

Table 3. Parameter Values in Four Simulation Runs

Analysis of Simulation Runs

Table 3 shows the parameters of the simulation runs we did to test various scenarios.

Figure 7 shows the effects of the parameter settings on Work Quality. The Base scenario (curve 1) allows Work Quality to be perfect, which is expected.

Adding an additional employee (curve 2) reduces quality, again as expected. Reducing coordination (curve 3) is a bit more deleterious to Work Quality, and it is hardly surprising that adding an employee and reducing coordination reduces work quality to its lowest level (curve 4).

Figure 8 shows the effects of these scenarios on Undiscovered Rework. The pattern is the same as it is for Work Quality. Undiscovered Rework increases a little more for each of the scenarios beyond the optimal Base—poor for the Additional Employee (curve 2), worse for Reduced Coordination (curve 3), and worst for both policies together (curve 4).

Figure 7. Results of Scenarios on Work Quality

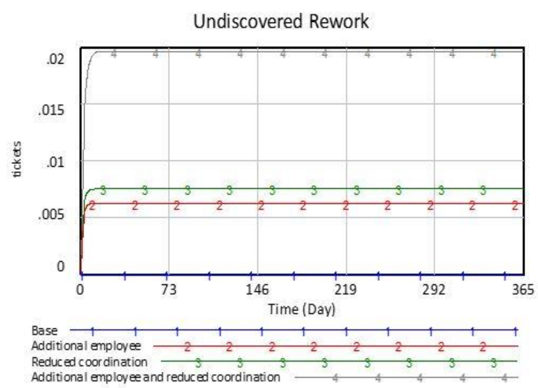

Figure 8. Results of Scenarios on Undiscovered Rework

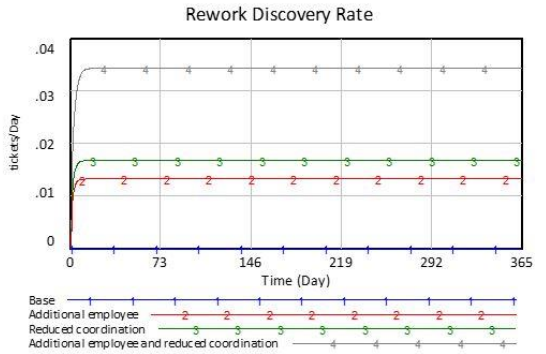

Figure 9. Results of Scenarios on Rework Discovery Rate

Figure 10. Results of Scenarios on Major Incident Resolution Rate

Figure 11. Results of Scenarios on Major Incident Tickets

Figure 9 shows the effects of the scenarios on Rework Discovery Rate. Each of the scenarios beyond the Base scenario increases the Rework Discovery Rate, but mostly because the increased level of the Undiscovered Rework stock by definition raises the rate of its discovery.

Figure 10 shows the effects of the scenarios on the Major Incident Resolution Rate. The pattern is the same, with the Additional Employee (curve 2), the Reduced Coordination (curve 3) and the two policies combined (curve 4) showing progressively higher rates of Major Incident Resolution. This is probably the result of the increased number of unresolved tickets in those three scenarios.

We show this in Figure 11, which shows the effects of the scenarios on the number of unresolved tickets. Again, the Additional Employee and the Suboptimal on Both scenarios increase the number of Unresolved Major Incident Tickets, which leads to their higher resolutions rates.

Figure 12 shows that reducing the number of staff assigned to a Critical Incident improves work quality, there is a limit—work quality will never be better than 100%. However, even with fewer employees, a reduction of coordination will hurt Work Quality (Figure 13). This strongly implies that it is the quality of coordination, not the number of employees, that determines the ultimate performance of this system.

Figure 12. Subtracting Staff Improves Work Quality, But There Is a Limit

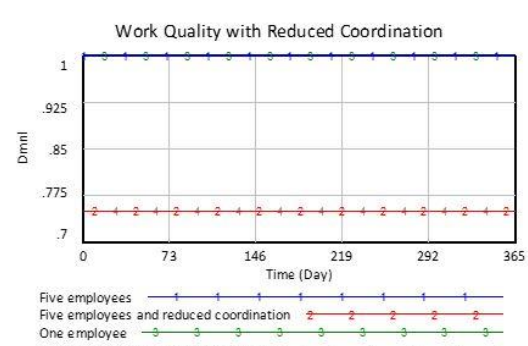

Figure 13. Reducing Coordination Reduces Work Quality More than Subtracting Staff Improves It.

Policy Analysis and Suggestions

As we showed in Figure 1, MPHC's current process flow for how to handle a Major Incident has a process section for "Coordinate Recovery Efforts." However, there is detail lacking as to how the coordination occurs. Based on the output from our model, the department should assign only the appropriate resources to work the issue. If those resources need assistance or have questions for others within the IT Department (or the business), they should reach out to them during the coordination, but management should consider those people *supplementary*, and not part of the Resolution/Workaround Team. Furthermore, instead of these primary resources calling into the phone bridge that is opened (a conference line), those people should set up in a conference room in the IT building so that they can communicate and troubleshoot together, instead of in silos. This will reduce the time to correct the error at hand.

This working group should document the steps taken to troubleshoot, implement a workaround and implement the final resolution. This not only would speed the rollback process (reversing steps taken so far), if necessary, but would help in the future for similar or repeated Major Incidents, leading to faster resolution times.