8 STPA: A New Hazard Analysis Technique

Hazard analysis can be described as "investigating an accident before it occurs." The goal is to identify potential causes of accidents, that is, scenarios that can lead to losses, so they can be eliminated or controlled in design or operations *before* damage occurs.

The most widely used existing hazard analysis techniques were developed fifty years ago and have serious limitations in their applicability to today's more complex, software-intensive, sociotechnical systems. This chapter describes a new approach to hazard analysis, based on the STAMP causality model, called STPA (System-Theoretic Process Analysis).

8.1 Goals for a New Hazard Analysis Technique

Three hazard analysis techniques are currently used widely: Fault Tree Analysis, Event Tree Analysis, and HAZOP. Variants that combine aspects of these three techniques, such as Cause-Consequence Analysis (combining top-down fault trees and forward analysis Event Trees) and Bowtie Analysis (combining forward and backward chaining techniques) are also sometimes used. *Safeware* and other basic textbooks contain more information about these techniques for those unfamiliar with them. FMEA (Failure Modes and Effects Analysis) is sometimes used as a hazard analysis technique, but it is a bottom-up reliability analysis technique and has very limited applicability for safety analysis.

The primary reason for developing STPA was to include the new causal factors identified in STAMP that are not handled by the older techniques. More specifically, the hazard analysis technique should include design errors, including software flaws; component interaction accidents; cognitively complex human decision-making errors; and social, organizational, and management factors contributing to accidents. In short, the goal is to identify accident scenarios that encompass the entire accident process, not just the electromechanical components. While attempts have been made to add new features to traditional hazard analysis techniques to handle new

technology, these attempts have had limited success because the underlying assumptions of the old techniques and the causality models on which they are based do not fit the characteristics of these new causal factors. STPA is based on the new causality assumptions identified in chapter 2.

An additional goal in the design of STPA was to provide guidance to the users in getting good results. Fault tree and event tree analysis provide little guidance to the analyst—the tree itself is simply the result of the analysis. Both the model of the system being used by the analyst and the analysis itself are only in the analyst's head. Analyst expertise in using these techniques is crucial, and the quality of the fault or event trees that result varies greatly.

HAZOP, widely used in the process industries, provides much more guidance to the analysts. HAZOP is based on a slightly different accident model than fault and event trees, namely that accidents result from deviations in system parameters, such as too much flow through a pipe or backflow when forward flow is required. HAZOP uses a set of guidewords to examine each part of a plant piping and wiring diagram, such as *more than*, *less than*, and *opposite*. Both guidance in performing the process and a concrete model of the physical structure of the plant are therefore available.

Like HAZOP, STPA works on a model of the system and has "guidewords" to assist in the analysis, but because in STAMP accidents are seen as resulting from inadequate control, the model used is a functional control diagram rather than a physical component diagram. In addition, the set of guidewords is based on lack of control rather than physical parameter deviations. While engineering expertise is still required, guidance is provided for the STPA process to provide some assurance of completeness in the analysis.

The third and final goal for STPA is that it can be used before a design has been created, that is, it provides the information necessary to guide the design process, rather than requiring a design to exist before the analysis can start. Designing safety into a system, starting in the earliest conceptual design phases, is the most cost-effective way to engineer safer systems. The analysis technique must also, of course, be applicable to existing designs or systems when safety-guided design is not possible.

8.2 The STPA Process

STPA (System-Theoretic Process Analysis) can be used at any stage of the system life cycle. It has the same general goals as any hazard analysis technique: accumulating information about how the behavioral safety constraints, which are derived from the system hazards, can be violated. Depending on when it is used, it provides the information and documentation necessary to ensure the safety constraints are

enforced in system design, development, manufacturing, and operations, including the natural changes in these processes that will occur over time.

STPA uses a functional control diagram and the requirements, system hazards, and the safety constraints and safety requirements for the component as defined in chapter 7. When STPA is applied to an existing design, this information is available when the analysis process begins. When STPA is used for safety-guided design, only the system-level requirements and constraints may be available at the beginning of the process. In the latter case, these requirements and constraints are refined and traced to individual system components as the iterative design and analysis process proceeds.

STPA has two main steps:

1. Identify the potential for inadequate control of the system that could lead to a hazardous state. Hazardous states result from inadequate control or enforcement of the safety constraints, which can occur because:
   a. A control action required for safety is *not* provided or not followed.
   b. An unsafe control action *is* provided.
   c. A potentially safe control action is provided too early or too late, that is, at the wrong time or in the wrong sequence.
   d. A control action required for safety is stopped too soon or applied too long.
2. Determine how each potentially hazardous control action identified in step 1 could occur.
   a. For each unsafe control action, examine the parts of the control loop to see if they could cause it. Design controls and mitigation measures if they do not already exist or evaluate existing measures if the analysis is being performed on an existing design. For multiple controllers of the same component or safety constraint, identify conflicts and potential coordination problems.
   b. Consider how the designed controls could degrade over time and build in protection, including
      i. Management of change procedures to ensure safety constraints are enforced in planned changes.
      ii. Performance audits where the assumptions underlying the hazard analysis are the preconditions for the operational audits and controls so that unplanned changes that violate the safety constraints can be detected.
      iii. Accident and incident analysis to trace anomalies to the hazards and to the system design.

While the analysis can be performed in one step, dividing the process into discrete steps reduces the analytical burden on the safety engineers and provides a

structured process for hazard analysis. The information from the first step (identifying the unsafe control actions) is required to perform the second step (identifying the causes of the unsafe control actions).

The assumption in this chapter is that the system design exists when STPA is performed. The next chapter describes safety-guided design using STPA and principles for safe design of control systems.

STPA is defined in this chapter using two examples. The first is a simple, generic interlock. The hazard involved is exposure of a human to a potentially dangerous energy source, such as high power. The power controller, which is responsible for turning the energy on or off, implements an interlock to prevent the hazard. In the physical controlled system, a door or barrier over the power source prevents exposure while it is active. To simplify the example, we will assume that humans cannot physically be inside the area when the barrier is in place—that is, the barrier is simply a cover over the energy source. The door or cover will be manually operated so the only function of the automated controller is to turn the power off when the door is opened and to turn it back on when the door is closed.

Given this design, the process starts from:

Hazard: Exposure to a high-energy source.

Constraint: The energy source must be off when the door is not closed.[1]

Figure 8.1 shows the control structure for this simple system. In this figure, the components of the system are shown along with the control instructions each component can provide and some potential feedback and other information or control sources for each component. Control operations by the automated controller include turning the power off and turning it on. The human operator can open and close the door. Feedback to the automated controller includes an indication of whether the door is open or not. Other feedback may be required or useful as determined during the STPA (hazard analysis) process.

The control structure for a second more complex example to be used later in the chapter, a fictional but realistic ballistic missile intercept system (FMIS), is shown in figure 8.2. Pereira, Lee, and Howard [154] created this example to describe their use of STPA to assess the risk of inadvertent launch in the U.S. Ballistic Missile Defense System (BMDS) before its first deployment and field test.

The BMDS is a layered defense to defeat all ranges of threats in all phases of flight (boost, midcourse, and terminal). The example used in this chapter is, for

1. The phrase "when the door is open" would be incorrect because a case is missing (a common problem): in the power controller's model of the controlled process, which enforces the constraint, the door may be open, closed, or the door position may be unknown to the controller. The phrase "is open or the door position is unknown" could be used instead. See section 9.3.2 for a discussion of why the difference is important.

HAZARD: Human exposed to high energy source

SYSTEM SAFETY CONSTRAINT: The energy source must be off whenever the door is not completely closed.

FUNCTIONAL REQUIREMENTS of the Power Controller:

(1) Detect when the door is opened and turn off the power

(2) When the door is closed, turn on the power

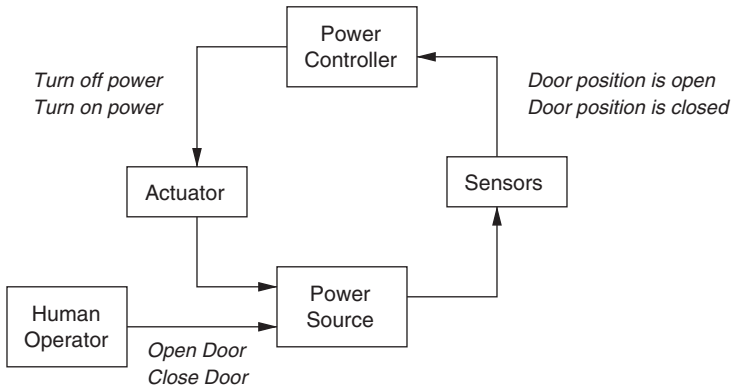

Figure 8.1
The control structure for a simple interlock system.

security reasons, changed from the real system, but it is realistic, and the problems identified by STPA in this chapter are similar to some that were found using STPA on the real system.

The U.S. BDMS system has a variety of components, including sea-based sensors in the Aegis shipborne platform; upgraded early warning systems; new and upgraded radars, ground-based midcourse defense, fire control, and communications; a Command and Control Battle Management and Communications component; and ground-based interceptors. Future upgrades will add features. Some parts of the system have been omitted in the example, such as the Aegis (ship-based) platform.

Figure 8.2 shows the control structure for the FMIS components included in the example. The command authority controls the operators by providing such things as doctrine, engagement criteria, and training. As feedback, the command authority gets the exercise results, readiness information, wargame results, and other information. The operators are responsible for controlling the launch of interceptors by sending instructions to the fire control subsystem and receiving status information as feedback.

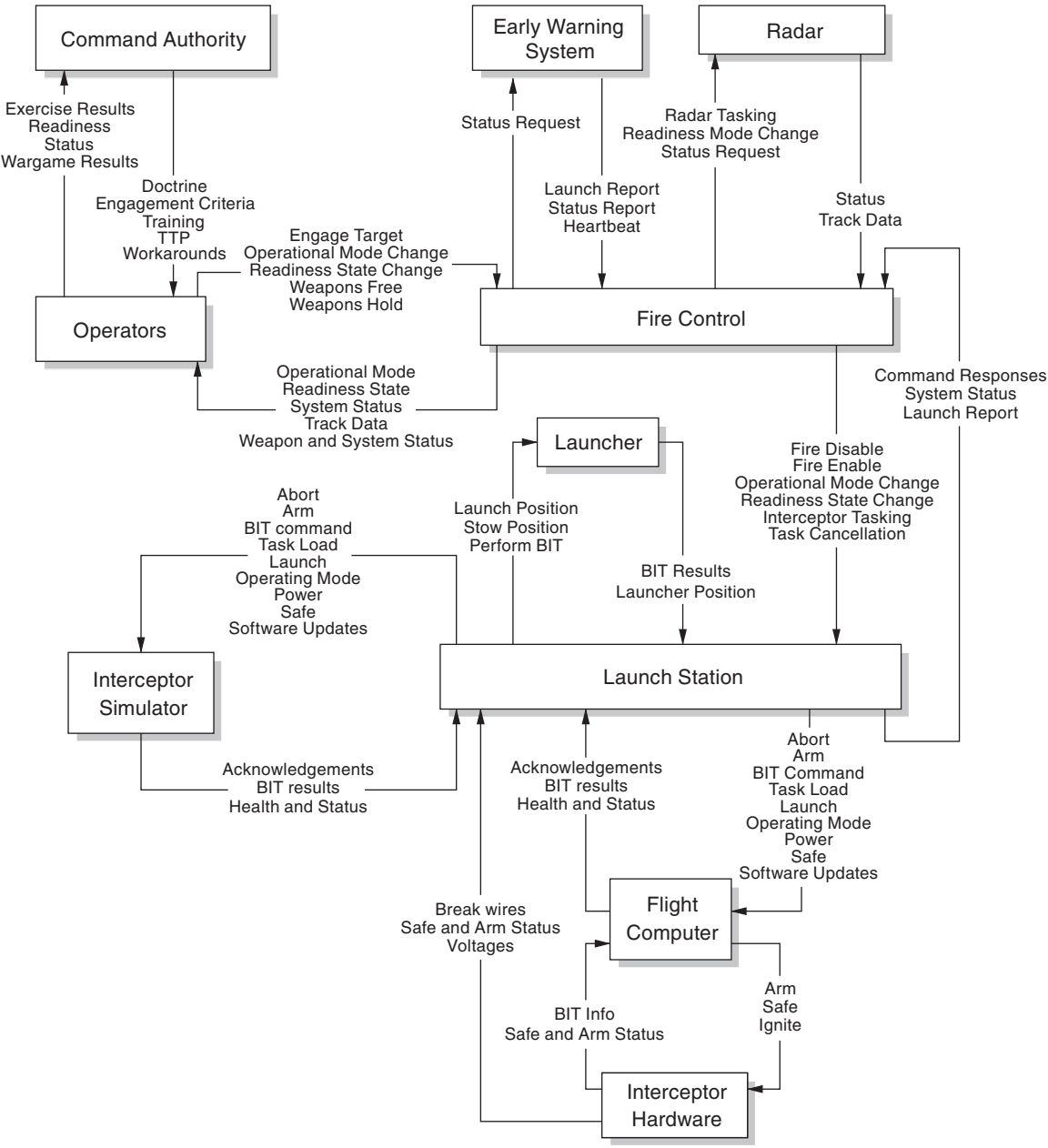

Figure 8.2
The control structure for a fictional ballistic missile defense system (FMIS) (adapted from Pereira, Lee, and Howard [154]).

Fire control receives instructions from the operators and information from the radars about any current threats. Using these inputs, fire control provides instructions to the launch station, which actually controls the launch of any interceptors. Fire control can enable firing, disable firing, and so forth, and, of course, it receives feedback from the launch station about the status of any previously provided control actions and the state of the system itself. The launch station controls the actual launcher and the flight computer, which in turn controls the interceptor hardware.

There is one other component of the system. To ensure operational readiness, the FMIS contains an interceptor simulator that periodically is used to mimic the flight computer in order to detect a failure in the system.

8.3 Identifying Potentially Hazardous Control Actions (Step 1)

Starting from the fundamentals defined in chapter 7, the first step in STPA is to assess the safety controls provided in the system design to determine the potential for inadequate control, leading to a hazard. The assessment of the hazard controls uses the fact that control actions can be hazardous in four ways (as noted earlier):

1. A control action required for safety is not provided or is not followed.
2. An unsafe control action is provided that leads to a hazard.
3. A potentially safe control action is provided too late, too early, or out of sequence.
4. A safe control action is stopped too soon or applied too long (for a continuous or nondiscrete control action).

For convenience, a table can be used to record the results of this part of the analysis. Other ways to record the information are also possible. In a classic System Safety program, the information would be included in the hazard log. Figure 8.3 shows the results of step 1 for the simple interlock example. The table contains four hazardous types of behavior:

1. A POWER OFF command is not given when the door is opened,
2. The door is opened and the controller waits too long to turn the power off;
3. A POWER ON command is given while the door is open, and
4. A POWER ON command is provided too early (when the door has not yet fully closed).

Incorrect but non-hazardous behavior is not included in the table. For example, not providing a POWER ON command when the power is off and the door is opened

Control Action	Not Providing Causes Hazard	Providing Causes Hazard	Wrong Timing or Order Causes Hazard	Stopped Too Soon or Applied Too Long
Power off	*Power not turned off when door opened*	Not Hazardous	*Door opened, controller waits too long to turn off power*	Not Applicable
Power on	Not Hazardous	*Power turned on while door opened*	*Power turned on too early; door not fully closed*	Not Applicable

Figure 8.3
Identifying hazardous system behavior.

or closed is not hazardous, although it may represent a quality-assurance problem. Another example of a mission assurance problem but not a hazard occurs when the power is turned off while the door is closed. Thomas has created a procedure to assist the analyst in considering the effect of all possible combinations of environmental and process variables for each control action in order to avoid missing any cases that should be included in the table [199a].

The final column of the table, *Stopped Too Soon or Applied Too Long*, is not applicable to the discrete interlock commands. An example where it does apply is in an aircraft collision avoidance system where the pilot may be told to climb or descend to avoid another aircraft. If the climb or descend control action is stopped too soon, the collision may not be avoided.

The identified hazardous behaviors can now be translated into safety constraints (requirements) on the system component behavior. For this example, four constraints must be enforced by the power controller (interlock):

1. The power must always be off when the door is open;
2. A POWER OFF command must be provided within *x* milliseconds after the door is opened;
3. A POWER ON command must never be issued when the door is open;
4. The POWER ON command must never be given until the door is fully closed.

For more complex examples, the mode in which the system is operating may determine the safety of the action or event. In that case, the operating mode may need to be included in the table, perhaps as an additional column. For example, some spacecraft mission control actions may only be hazardous during the launch or reentry phase of the mission.

In chapter 2, it was stated that many accidents, particularly component interaction accidents, stem from incomplete requirements specifications. Examples were

Command	Not Providing Causes Hazard	Providing Causes Hazard	Wrong Timing/Order Causes Hazard	Stopped Too Soon or Applied Too Long
Fire Enable	Not Hazardous	*Will accept interceptor tasking and can progress to a launch sequence*	*EARLY: Can inadvertently progress to an inadvertent launch*	Not Applicable
			OUT OF SEQUENCE: Disable comes before the enable	
...				

Figure 8.4
One row of the table identifying FMIS hazardous control actions.

provided such as missing constraints on the order of valve position changes in a batch chemical reactor and the conditions under which the descent engines should be shut down on the Mars Polar Lander spacecraft. The information provided in this first step of STPA can be used to identify the necessary constraints on component behavior to prevent the identified system hazards, that is, the safety requirements. In the second step of STPA, the information required by the component to properly implement the constraint is identified as well as additional safety constraints and information necessary to eliminate or control the hazards in the design or to design the system properly in the first place.

The FMIS system provides a less trivial example of step 1. Remember, the hazard is inadvertent launch. Consider the FIRE ENABLE command, which can be sent by the fire control module to the launch station to allow launch commands subsequently received by the launch station to be executed. As described in Pereira, Lee, and Howard [154], the FIRE ENABLE control command directs the launch station to enable the live fire of interceptors. Prior to receiving this command, the launch station will return an error message when it receives commands to fire an interceptor and will discard the fire commands.[2]

Figure 8.4 shows the results of performing STPA Step 1 on the FIRE ENABLE command. If this command is missing (column 2), a launch will not take place. While this omission might potentially be a mission assurance concern, it does not contribute to the hazard being analyzed (inadvertent launch).

2. Section 9.4.4 explains the safety-related reasons for breaking up potentially hazardous actions into multiple steps.

If the FIRE ENABLE command is provided to a launch station incorrectly, the launch station will transition to a state where it accepts interceptor tasking and can progress through a launch sequence. In combination with other incorrect or mistimed commands, this control action could contribute to an inadvertent launch.

A late FIRE ENABLE command will only delay the launch station's ability to process a launch sequence, which will not contribute to an inadvertent launch. A FIRE ENABLE command sent too early could open a window of opportunity for inadvertently progressing toward an inadvertent launch, similar to the incorrect FIRE ENABLE considered above. In the third case, a FIRE ENABLE command might be out of sequence with a FIRE DISABLE command. If this incorrect sequencing is possible in the system as designed and constructed, the system could be left capable of processing interceptor tasking and launching an interceptor when not intended.

Finally, the FIRE ENABLE command is a discrete command sent to the launch station to signal that it should allow processing of interceptor tasking. Because FIRE ENABLE is not a continuous command, the "stopped too soon" category does not apply.

8.4 Determining How Unsafe Control Actions Could Occur (Step 2)

Performing the first step of STPA provides the component safety requirements, which may be sufficient for some systems. A second step can be performed, however, to identify the scenarios leading to the hazardous control actions that violate the component safety constraints. Once the potential causes have been identified, the design can be checked to ensure that the identified scenarios have been eliminated or controlled in some way. If not, then the design needs to be changed. If the design does not already exist, then the designers at this point can try to eliminate or control the behaviors as the design is created, that is, use safety-guided design as described in the next chapter.

Why is the second step needed? While providing the engineers with the safety constraints to be enforced is necessary, it is not sufficient. Consider the chemical batch reactor described in section 2.1. The hazard is overheating of the reactor contents. At the system level, the engineers may decide (as in this design) to use water and a reflux condenser to control the temperature. After this decision is made, controls need to be enforced on the valves controlling the flow of catalyst and water. Applying step 1 of STPA determines that opening the valves out of sequence is dangerous, and the software requirements would accordingly be augmented with constraints on the order of the valve opening and closing instructions, namely that the water valve must be opened before the catalyst valve and the catalyst valve must be closed before the water valve is closed or, more generally, that the water valve

must always be open when the catalyst valve is opened. If the software already exists, the hazard analysis would ensure that this ordering of commands has been enforced in the software. Clearly, building the software to enforce this ordering is a great deal easier than proving the ordering is true after the software already exists.

But enforcing these safety constraints is not enough to ensure safe software behavior. Suppose the software has commanded the water valve to open but something goes wrong and the valve does not actually open or it opens but water flow is restricted in some way (the *no flow* guideword in HAZOP). Feedback is needed for the software to determine if water is flowing through the pipes and the software needs to check this feedback before opening the catalyst valve. The second step of STPA is used to identify the ways that the software safety constraint, even if provided to the software engineers, might still not be enforced by the software logic and system design. In essence, step 2 identifies the scenarios or paths to a hazard found in a classic hazard analysis. This step is the usual "magic" one that creates the contents of a fault tree, for example. The difference is that guidance is provided to help create the scenarios and more than just failures are considered.

To create causal scenarios, the control structure diagram must include the process models for each component. If the system exists, then the content of these models should be easily determined by looking at the system functional design and its documentation. If the system does not yet exist, the analysis can start with a best guess and then be refined and changed as the analysis process proceeds.

For the high power interlock example, the process model is simple and shown in figure 8.5. The general causal factors, shown in figure 4.8 and repeated here in figure 8.6 for convenience, are used to identify the scenarios.

8.4.1 Identifying Causal Scenarios

Starting with each hazardous control action identified in step 1, the analysis in step 2 involves identifying how it could happen. To gather information about how the hazard could occur, the parts of the control loop for each of the hazardous control actions identified in step 1 are examined to determine if they could cause or contribute to it. Once the potential causes are identified, the engineers can design controls and mitigation measures if they do not already exist or evaluate existing measures if the analysis is being performed on an existing design.

Each potentially hazardous control action must be considered. As an example, consider the unsafe control action of not turning off the power when the door is opened. Figure 8.7 shows the results of the causal analysis in a graphical form. Other ways of documenting the results are, of course, possible.

The hazard in figure 8.7 is that the door is open but the power is not turned off. Looking first at the controller itself, the hazard could result if the requirement is not passed to the developers of the controller, the requirement is not implemented

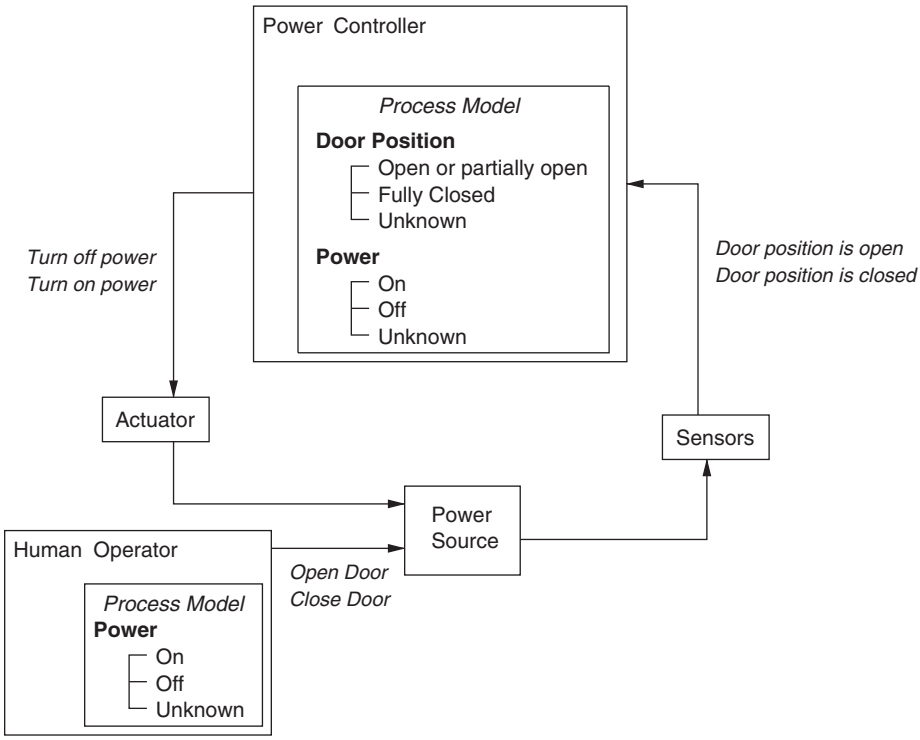

Figure 8.5
The process model for the high-energy controller.

correctly, or the process model incorrectly shows the door closed and/or the power off when that is not true. Working around the loop, the causal factors for each of the loop components are similarly identified using the general causal factors shown in figure 8.6. These causes include that the POWER OFF command is sent but not received by the actuator, the actuator received the command but does not implement it (actuator failure), the actuator delays in implementing the command, the POWER ON and POWER OFF commands are received or executed in the wrong order, the door open event is not detected by the door sensor or there is an unacceptable delay in detecting it, the sensor fails or provides spurious feedback, and the feedback about the state of the door or the power is not received by the controller or is not incorporated correctly into the process model.

More detailed causal analysis can be performed if a specific design is being considered. For example, the features of the communication channels used will determine the potential way that commands or feedback could be lost or delayed.

Once the causal analysis is completed, each of the causes that cannot be shown to be physically impossible must be checked to determine whether they are

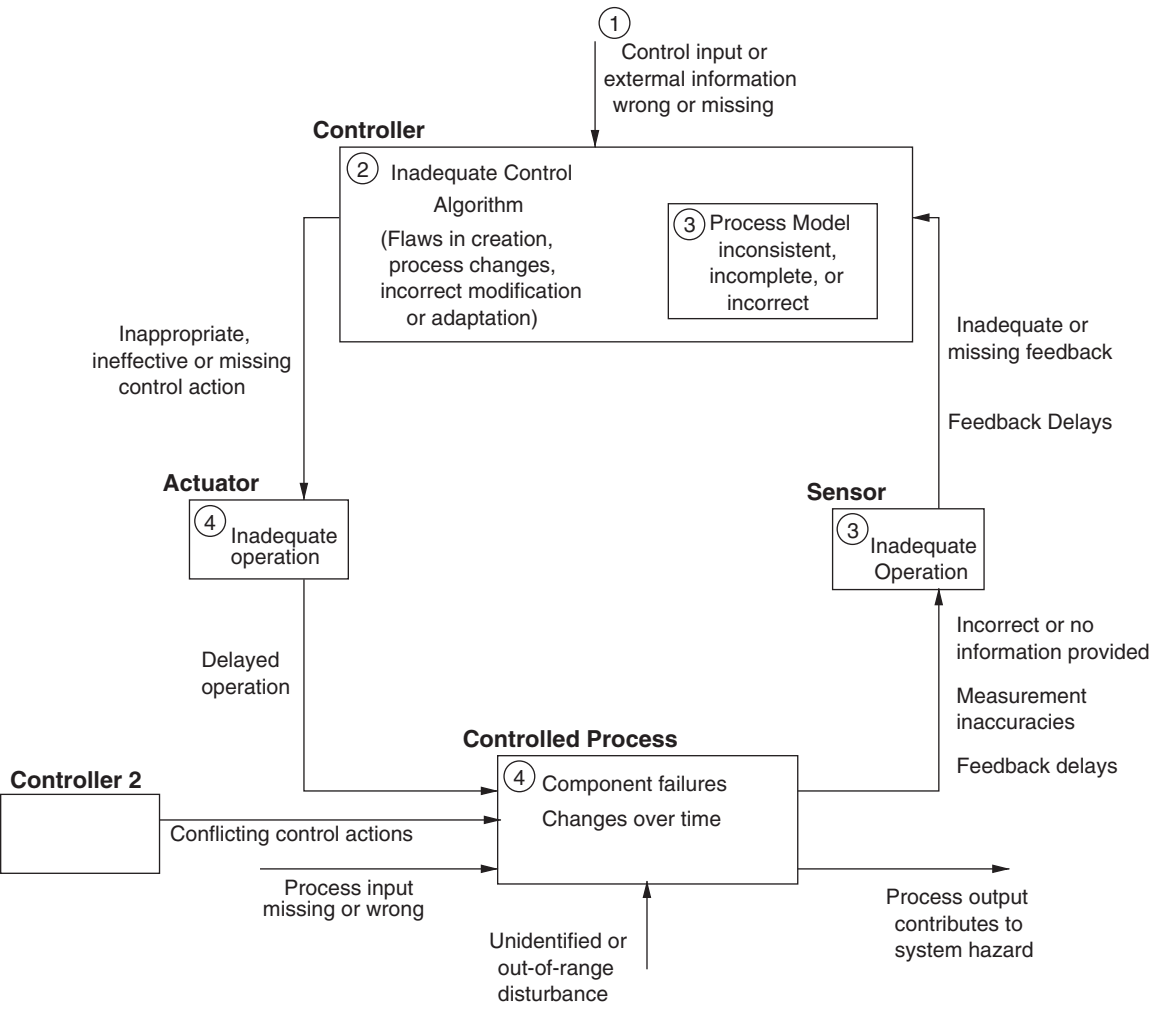

Figure 8.6
The causal factors to be considered to create scenarios in step 3.

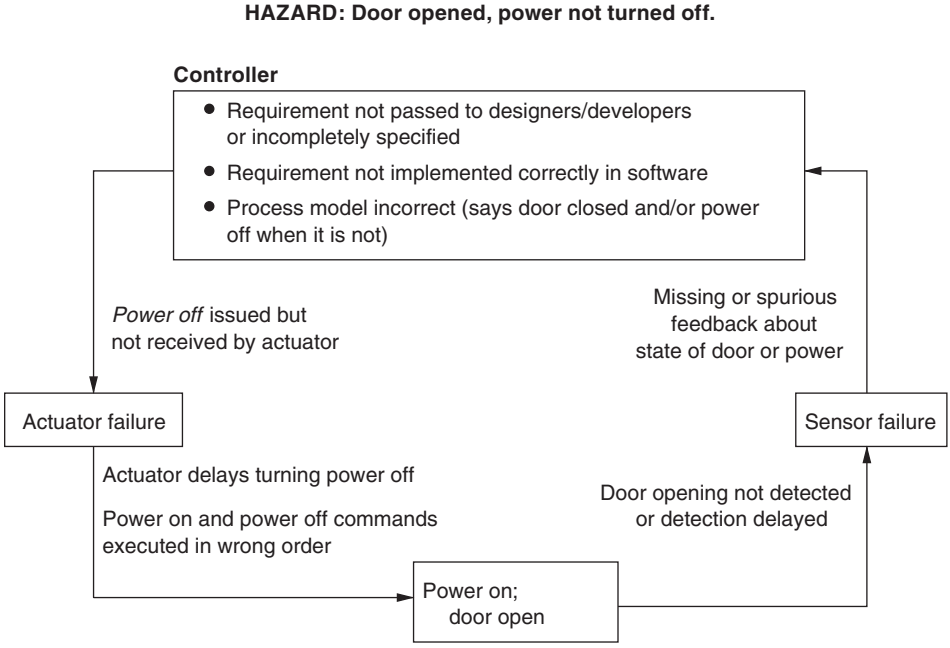

Figure 8.7
Example of step 2b STPA analysis for the high power interlock.

adequately handled in the design (if the design exists) or design features added to control them if the design is being developed with support from the analysis.

The first step in designing for safety is to try to eliminate the hazard completely. In this example, the hazard can be eliminated by redesigning the system to have the circuit run through the door in such a way that the circuit is broken as soon as the door opens. Let's assume, however, that for some reason this design alternative is rejected, perhaps as impractical. Design precedence then suggests that the next best alternatives in order are to reduce the likelihood of the hazard occurring, to prevent the hazard from leading to a loss, and finally to minimize damage. More about safe design can be found in chapters 16 and 17 of *Safeware* and chapter 9 of this book.

Because design almost always involves tradeoffs with respect to achieving multiple objectives, the designers may have good reasons not to select the most effective way to control the hazard but one of the other alternatives instead. It is important that the rationale behind the choice is documented for future analysis, certification, reuse, maintenance, upgrades, and other activities.

For this simple example, one way to mitigate many of the causes is to add a light that identifies whether the power supply is on or off. How do human operators know that the power has been turned off before inserting their hands into the high-energy

power source? In the original design, they will most likely assume it is off because they have opened the door, which may be an incorrect assumption. Additional feedback and assurance can be attained from the light. In fact, protection systems in automated factories commonly are designed to provide humans in the vicinity with aural or visual information that they have been detected by the protection system. Of course, once a change has been made, such as adding a light, that change must then be analyzed for new hazards or causal scenarios. For example, a light bulb can burn out. The design might ensure that the safe state (the power is off) is represented by the light being on rather than the light being off, or two colors might be used. Every solution for a safety problem usually has its own drawbacks and limitations and therefore they will need to be compared and decisions made about the best design given the particular situation involved.

In addition to the factors shown in figure 8.6, the analysis must consider the impact of having two controllers of the same component whenever this occurs in the system safety control structure. In the friendly fire example in chapter 5, for example, confusion existed between the two AWACS operators responsible for tracking aircraft inside and outside of the no-fly-zone about who was responsible for aircraft in the boundary area between the two. The FMIS example below contains such a scenario. An analysis must be made to determine that no path to a hazard exists because of coordination problems.

The FMIS system provides a more complex example of STPA step 2. Consider the FIRE ENABLE command provided by fire control to the launch station. In step 1, it was determined that if this command is provided incorrectly, the launch station will transition to a state where it accepts interceptor tasking and can progress through a launch sequence. In combination with other incorrect or mistimed control actions, this incorrect command could contribute to an inadvertent launch.

The following are two examples of causal factors identified using STPA step 2 as potentially leading to the hazardous state (violation of the safety constraint). Neither of these examples involves component failures, but both instead result from unsafe component interactions and other more complex causes that are for the most part not identifiable by current hazard analysis methods.

In the first example, the FIRE ENABLE command can be sent inadvertently due to a missing case in the requirements—a common occurrence in accidents where software is involved.

The FIRE ENABLE command is sent when the fire control receives a WEAPONS FREE command from the operators and the fire control system has at least one active track. An active track indicates that the radars have detected something that might be an incoming missile. Three criteria are specified for declaring a track inactive: (1) a given period passes with no radar input, (2) the total predicted impact time elapses for the track, and (3) an intercept is confirmed. Operators are allowed to

deselect any of these options. One case was not considered by the designers: if an operator deselects all of the options, no tracks will be marked as inactive. Under these conditions, the inadvertent entry of a WEAPONS FREE command would send the FIRE ENABLE command to the launch station immediately, even if there were no threats currently being tracked by the system.

Once this potential cause is identified, the solution is obvious—fix the software requirements and the software design to include the missing case. While the operator might instead be warned not to deselect all the options, this kind of human error is possible and the software should be able to handle the error safely. Depending on humans not to make mistakes is an almost certain way to guarantee that accidents will happen.

The second example involves confusion between the regular and the test software. The FMIS undergoes periodic system operability testing using an interceptor simulator that mimics the interceptor flight computer. The original hazard analysis had identified the possibility that commands intended for test activities could be sent to the operational system. As a result, the system status information provided by the launch station includes whether the launch station is connected only to missile simulators or to any live interceptors. If the fire control computer detects a change in this state, it will warn the operator and offer to reset into a matching state. There is, however, a small window of time before the launch station notifies the fire control component of the change. During this time interval, the fire control software could send a FIRE ENABLE command intended for test to the live launch station. This latter example is a coordination problem arising because there are multiple controllers of the launch station and two operating modes (e.g., testing and live fire). A potential mode confusion problem exists where the launch station can think it is in one mode but really be in the other one. Several different design changes could be used to prevent this hazardous state.

In the use of STPA on the real missile defense system, the risks involved in integrating separately developed components into a larger system were assessed, and several previously unknown scenarios for inadvertent launch were identified. Those conducting the assessment concluded that the STPA analysis and supporting data provided management with a sound basis on which to make risk acceptance decisions [154]. The assessment results were used to plan mitigations for open safety risks deemed necessary to change before deployment and field-testing of the system. As system changes are proposed, they are assessed by updating the control structure diagrams and assessment analysis results.

8.4.2 Considering the Degradation of Controls over Time

A final step in STPA is to consider how the designed controls could degrade over time and to build in protection against it. The mechanisms for the degradation could

be identified and mitigated in the design: for example, if corrosion is identified as a potential cause, a stronger or less corrosive material might be used. Protection might also include planned performance audits where the assumptions underlying the hazard analysis are the preconditions for the operational audits and controls. For example, an assumption for the interlock system with a light added to warn the operators is that the light is operational and operators will use it to determine whether it is safe to open the door. *Performance audits* might check to validate that the operators know the purpose of the light and the importance of not opening the door while the warning light is on. Over time, operators might create workarounds to bypass this feature if it slows them up too much in their work or if they do not understand the purpose, the light might be partially blocked from view because of workplace changes, and so on. The assumptions and required audits should be identified during the system design process and then passed to the operations team.

Along with performance audits, *management of change procedures* need to be developed and the STPA analysis revisited whenever a planned change is made in the system design. Many accidents occur after changes have been made in the system. If appropriate documentation is maintained along with the rationale for the control strategy selected, this reanalysis should not be overly burdensome. How to accomplish this goal is discussed in chapter 10.

Finally, after accidents and incidents, the design and the hazard analysis should be revisited to determine why the controls were not effective. The hazard of foam damaging the thermal surfaces of the Space Shuttle had been identified during design, for example, but over the years before the *Columbia* loss the process for updating the hazard analysis after anomalies occurred in flight was eliminated. The Space Shuttle standard for hazard analyses (NSTS 22254, Methodology for Conduct of Space Shuttle Program Hazard Analyses) specified that hazards be revisited only when there was a new design or the design was changed: There was no process for updating the hazard analyses when anomalies occurred or even for determining whether an anomaly was related to a known hazard [117].

Chapter 12 provides more information about the use of the STPA results during operations.

8.5 Human Controllers

Humans in the system can be treated in the same way as automated components in step 1 of STPA, as was seen in the interlock system above where a person controlled the position of the door. The causal analysis and detailed scenario generation for human controllers, however, is much more complex than that of electromechanical devices and even software, where at least the algorithm is known and can be evaluated. Even if operators are given a procedure to follow, for reasons discussed in

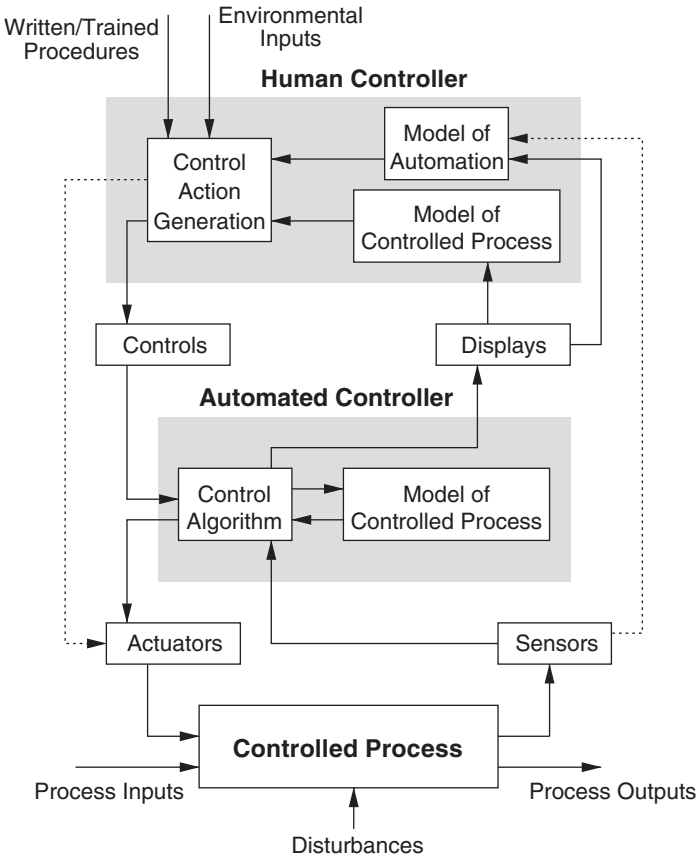

Figure 8.8
A human controller controlling an automated controller controlling a physical process.

chapter 2, it is very likely that the operator may feel the need to change the procedure over time.

The first major difference between human and automated controllers is that humans need an additional process model. All controllers need a model of the process they are controlling directly, but human controllers also need a model of any process, such as an oil refinery or an aircraft, they are indirectly controlling through an automated controller. If the human is being asked to supervise the automated controller or to monitor it for wrong or dangerous behavior then he or she needs to have information about the state of both the automated controller and the controlled process. Figure 8.8 illustrates this requirement. The need for an additional process model explains why supervising an automated system requires extra training and skill. A wrong assumption is sometimes made that if the

human is supervising a computer, training requirements are reduced but this belief is untrue. Human skill levels and required knowledge almost always go up in this situation.

Figure 8.8 includes dotted lines to indicate that the human controller may need direct access to the process actuators if the human is to act as a backup to the automated controller. In addition, if the human is to monitor the automation, he or she will need direct input from the sensors to detect when the automation is confused and is providing incorrect information as feedback about the state of the controlled process.

The system design, training, and operational procedures must support accurate creation and updating of the extra process model required by the human supervisor. More generally, when a human is supervising an automated controller, there are extra analysis and design requirements. For example, the control algorithm used by the automation must be learnable and understandable. Inconsistent behavior or unnecessary complexity in the automation function can lead to increased human error. Additional design requirements are discussed in the next chapter.

With respect to STPA, the extra process model and complexity in the system design requires additional causal analysis when performing step 2 to determine the ways that both process models can become inaccurate.

The second important difference between human and automated controllers is that, as noted by Thomas [199], while automated systems have basically static control algorithms (although they may be updated periodically), humans employ dynamic control algorithms that they change as a result of feedback and changes in goals. Human error is best modeled and understood using feedback loops, not as a chain of directly related events or errors as found in traditional accident causality models. Less successful actions are a natural part of the search by operators for optimal performance [164].

Consider again figure 2.9. Operators are often provided with procedures to follow by designers. But designers are dealing with their own models of the controlled process, which may not reflect the actual process as constructed and changed over time. Human controllers must deal with the system as it exists. They update their process models using feedback, just as in any control loop. Sometimes humans use experimentation to understand the behavior of the controlled system and its current state and use that information to change their control algorithm. For example, after picking up a rental car, drivers may try the brakes and the steering system to get a feel for how they work before driving on a highway.

If human controllers suspect a failure has occurred in a controlled process, they may experiment to try to diagnose it and determine a proper response. Humans also use experimentation to determine how to optimize system performance. The driver's control algorithm may change over time as the driver learns more about

the automated system and learns how to optimize the car's behavior. Driver goals and motivation may also change over time. In contrast, automated controllers by necessity must be designed with a single set of requirements based on the designer's model of the controlled process and its environment.

Thomas provides an example [199] using cruise control. Designers of an automated cruise control system may choose a control algorithm based on their model of the vehicle (such as weight, engine power, response time), the general design of roadways and vehicle traffic, and basic engineering design principles for propulsion and braking systems. A simple control algorithm might control the throttle in proportion to the difference between current speed (monitored through feedback) and desired speed (the goal).

Like the automotive cruise control designer, the human driver also has a process model of the car's propulsion system, although perhaps simpler than that of the automotive control expert, including the approximate rate of car acceleration for each accelerator position. This model allows the driver to construct an appropriate control algorithm for the current road conditions (slippery with ice or clear and dry) and for a given goal (obeying the speed limit or arriving at the destination at a required time). Unlike the static control algorithm designed into the automated cruise control, the human driver may dynamically change his or her control algorithm over time based on changes in the car's performance, in goals and motivation, or driving experience.

The differences between automated and human controllers lead to different requirements for hazard analysis and system design. Simply identifying human "failures" or errors is not enough to design safer systems. Hazard analysis must identify the specific human behaviors that can lead to the hazard. In some cases, it may be possible to identify why the behaviors occur. In either case, we are not able to "redesign" humans. Training can be helpful, but not nearly enough—training can do only so much in avoiding human error even when operators are highly trained and skilled. In many cases, training is impractical or minimal, such as automobile drivers. The only real solution lies in taking the information obtained in the hazard analysis about worst-case human behavior and using it in the design of the other system components and the system as a whole to eliminate, reduce, or compensate for that behavior. Chapter 9 discusses why we need human operators in systems and how to design to eliminate or reduce human errors.

STPA as currently defined provides much more useful information about the cause of human errors than traditional hazard analysis methods, but augmenting STPA could provide more information for designers. Stringfellow has suggested some additions to STPA for human controllers [195]. In general, engineers need better tools for including humans in hazard analyses in order to cope with the unique aspects of human control.

8.6 Using STPA on Organizational Components of the Safety Control Structure

The examples above focus on the lower levels of safety control structures, but STPA can also be used on the organizational and management components. Less experimentation has been done on applying it at these levels, and, once again, more needs to be done.

Two examples are used in this section: one was a demonstration for NASA of risk analysis using STPA on a new management structure proposed after the *Columbia* accident. The second is pharmaceutical safety. The fundamental activities of identifying system hazards, safety requirements and constraints, and of documenting the safety control structure were described for these two examples in chapter 7. This section starts from that point and illustrates the actual risk analysis process.

8.6.1 Programmatic and Organizational Risk Analysis

The Columbia Accident Investigation Board (CAIB) found that one of the causes of the *Columbia* loss was the lack of independence of the safety program from the Space Shuttle program manager. The CAIB report recommended that NASA institute an Independent Technical Authority (ITA) function similar to that used in SUBSAFE (see chapter 14), and individuals with SUBSAFE experience were recruited to help design and implement the new NASA Space Shuttle program organizational structure. After the program was designed and implementation started, a risk analysis of the program was performed to assist in a planned review of the program's effectiveness. A classic programmatic risk analysis, which used experts to identify the risks in the program, was performed. In parallel, a group at MIT developed a process to use STAMP as a foundation for the same type of programmatic risk analysis to understand the risks and vulnerabilities of this new organizational structure and recommend improvements [125].[3] This section describes the STAMP-based process and results as an example of what can be done for other systems and other emergent properties. Laracy [108] used a similar process to examine transportation system security, for example.

The STAMP-based analysis rested on the basic STAMP concept that most major accidents do not result simply from a unique set of proximal, physical events but from the migration of the organization to a state of heightened risk over time as safeguards and controls are relaxed due to conflicting goals and tradeoffs. In such a high-risk state, events are bound to occur that will trigger an accident. In both the *Challenger* and *Columbia* losses, organizational risk had been increasing to unacceptable levels for quite some time as behavior and decision-making evolved in

3. Many people contributed to the analysis described in this section, including Nicolas Dulac, Betty Barrett, Joel Cutcher-Gershenfeld, John Carroll, and Stephen Friedenthal.

response to a variety of internal and external performance pressures. Because risk increased slowly, nobody noticed, that is, the *boiled frog* phenomenon. In fact, confidence and complacency were increasing at the same time as risk due to the lack of accidents.

The goal of the STAMP-based analysis was to apply a classic system safety engineering process to the analysis and redesign of this organizational structure. Figure 8.9 shows the basic process used, which started with a preliminary hazard analysis to identify the system hazards and the safety requirements and constraints. In the second step, a STAMP model of the ITA safety control structure was created (as designed by NASA; see figure 7.4) and a gap analysis was performed to map the identified safety requirements and constraints to the assigned responsibilities in the safety control structure and identify any gaps. A detailed hazard analysis using STPA was then performed to identify the system risks and to generate recommendations for improving the designed new safety control structure and for monitoring the implementation and long-term health of the new program. Only enough of the modeling and analysis is included here to allow the reader to understand the process. The complete modeling and analysis effort is documented elsewhere [125].

The hazard identification, system safety requirements, and safety control structure for this example are described in section 7.4.1, so the example starts from this basic information.

8.6.2 Gap Analysis

In analyzing an existing organizational or social safety control structure, one of the first steps is to determine where the responsibility for implementing each requirement rests and to perform a *gap analysis* to identify holes in the current design, that is, requirements that are not being implemented (enforced) anywhere. Then the safety control structure needs to be evaluated to determine whether it is potentially effective in enforcing the system safety requirements and constraints.

A mapping was made between the system-level safety requirements and constraints and the individual responsibilities of each component in the NASA safety control structure to see where and how requirements are enforced. The ITA program was at the time being carefully defined and documented. In other situations, where such documentation may be lacking, interview or other techniques may need to be used to elicit how the organizational control structure actually works. In the end, complete documentation should exist in order to maintain and operate the system safely. While most organizations have job descriptions for each employee, the safety-related responsibilities are not necessarily separated out or identified, which can lead to unidentified gaps or overlaps.

As an example, in the ITA structure the responsibility for the system-level safety requirement:

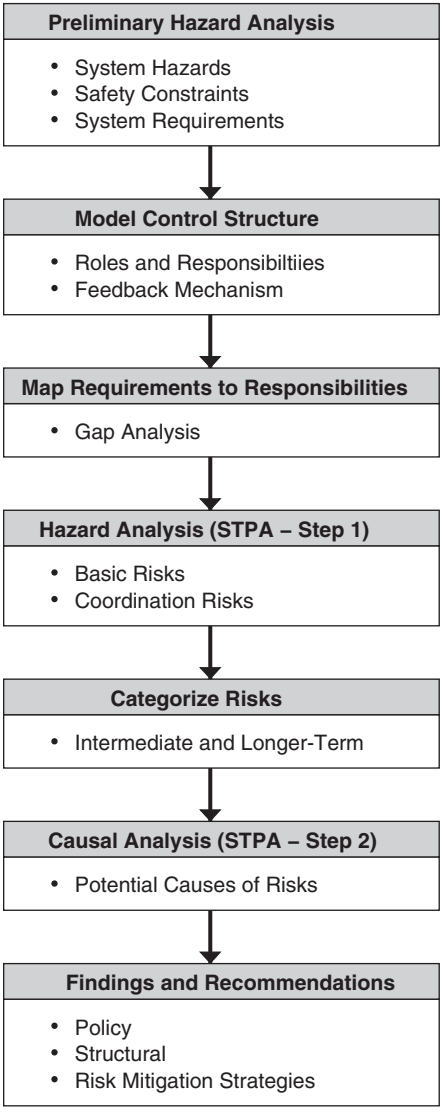

Figure 8.9
The basic process used in the NASA ITA risk analysis.

1a. State-of-the art safety standards and requirements for NASA missions must be established, implemented, enforced, and maintained that protect the astronauts, the workforce, and the public

was assigned to the NASA Chief Engineer but the Discipline Technical Warrant Holders, the Discipline Trusted Agents, the NASA Technical Standards Program, and the headquarters Office of Safety and Mission Assurance also play a role in implementing this Chief Engineer responsibility. More specifically, system requirement *1a* was implemented in the control structure by the following responsibility assignments:

- **Chief Engineer:** Develop, monitor, and maintain technical standards and policy.
- **Discipline Technical Warrant Holders:**
  - Recommend priorities for development and updating of technical standards.
  - Approve all new or updated NASA Preferred Standards within their assigned discipline (the NASA Chief Engineer retains Agency approval)
  - Participate in (lead) development, adoption, and maintenance of NASA Preferred Technical Standards in the warranted discipline.
  - Participate as members of technical standards working groups.
- **Discipline Trusted Agents:** Represent the Discipline Technical Warrant Holders on technical standards committees
- **NASA Technical Standards Program:** Coordinate with Technical Warrant Holders when creating or updating standards
- **NASA Headquarters Office Safety and Mission Assurance:**
  - Develop and improve generic safety, reliability, and quality process standards and requirements, including FMEA, risk, and the hazard analysis process.
  - Ensure that safety and mission assurance policies and procedures are adequate and properly documented.

Once the mapping is complete, a gap analysis can be performed to ensure that each system safety requirement and constraint is embedded in the organizational design and to find holes or weaknesses in the design. In this analysis, concerns surfaced, particularly about requirements not reflected in the defined ITA organizational structure.

As an example, one omission detected was appeals channels for complaints and concerns about the components of the ITA structure itself that may not function appropriately. All channels for expressing what NASA calls "technical conscience" go through the warrant holders, but there was no defined way to express

concerns about the warrant holders themselves or about aspects of ITA that are not working well.

A second example was the omission in the documentation of the ITA implementation plans of the person(s) who was to be responsible to see that engineers and managers are trained to use the results of hazard analyses in their decision making. More generally, a distributed and ill-defined responsibility for the hazard analysis process made it difficult to determine responsibility for ensuring that adequate resources are applied; that hazard analyses are elaborated (refined and extended) and updated as the design evolves and test experience is acquired; that hazard logs are maintained and used as experience is acquired; and that all anomalies are evaluated for their hazard potential. Before ITA, many of these responsibilities were assigned to each Center's Safety and Mission Assurance Office, but with much of this process moving to engineering (which is where it should be) under the new ITA structure, clear responsibilities for these functions need to be specified. One of the basic causes of accidents in STAMP is multiple controllers with poorly defined or overlapping responsibilities.

A final example involved the ITA program assessment process. An assessment of how well ITA is working is part of the plan and is an assigned responsibility of the chief engineer. The official risk assessment of the ITA program performed in parallel with the STAMP-based one was an implementation of that chief engineer's responsibility and was planned to be performed periodically. We recommended the addition of specific organizational structures and processes for implementing a continual learning and improvement process and making adjustments to the design of ITA itself when necessary outside of the periodic review.

8.6.3 Hazard Analysis to Identify Organizational and Programmatic Risks

A risk analysis to identify ITA programmatic risks and to evaluate these risks periodically had been specified as one of the chief engineer's responsibilities. To accomplish this goal, NASA identified the programmatic risks using a classic process using experts in risk analysis interviewing stakeholders and holding meetings where risks were identified and discussed. The STAMP-based analysis used a more formal, structured approach.

Risks in STAMP terms can be divided into two types: (1) basic inadequacies in the way individual components in the control structure fulfill their responsibilities and (2) risks involved in the coordination of activities and decision making that can lead to unintended interactions and consequences.

Basic Risks

Applying the four types of inadequate control identified in STPA and interpreted for the hazard, which in this case is unsafe decision-making leading to an accident, ITA has four general types of risks:

1. Unsafe decisions are made or approved by the chief engineer or warrant holders.
2. Safe decisions are disallowed (e.g., overly conservative decision making that undermines the goals of NASA and long-term support for ITA).
3. Decision making takes too long, minimizing impact and also reducing support for the ITA.
4. Good decisions are made by the ITA, but do not have adequate impact on system design, construction, and operation.

The specific potentially unsafe control actions by those in the ITA safety control structure that could lead to these general risks are the ITA programmatic risks. Once identified, they must be eliminated or controlled just like any unsafe control actions.

Using the responsibilities and control actions defined for the components of the safety control structure, the STAMP-based risk analysis applied the four general types of inadequate control actions, omitting those that did not make sense for the particular responsibility or did not impact risk. To accomplish this, the general responsibilities must be refined into more specific control actions.

As an example, the chief engineer is responsible as the ITA for the technical standards and system requirements and all changes, variances, and waivers to the requirements, as noted earlier. The control actions the chief engineer has available to implement this responsibility are:

- To develop, monitor, and maintain technical standards and policy.
- In coordination with programs and projects, to establish or approve the technical requirements and ensure they are enforced and implemented in the programs and projects (ensure the design is compliant with the requirements).
- To approve all changes to the initial technical requirements.
- To approve all variances (waivers, deviations, exceptions to the requirements.
- Etc.

Taking just one of these, the control responsibility to develop, monitor, and maintain technical standards and policy, the risks (potentially inadequate or unsafe control actions) identified using STPA step 1 include:

1. General technical and safety standards are not created.
2. Inadequate standards and requirements are created.
3. Standards degrade over time due to external pressures to weaken them. The process for approving changes is flawed.
4. Standards are not changed over time as the environment changes.

As another example, the chief engineer cannot perform all these duties himself, so he has a network of people below him in the hierarchy to whom he delegates or "warrants" some of the responsibilities. The chief engineer retains responsibility for ensuring that the warrant holders perform their duties adequately as in any hierarchical management structure.

The chief engineer responsibility to approve all variances and waivers to technical requirements is assigned to the System Technical Warrant Holder (STWH). The risks or potentially unsafe control actions of the STWH with respect to this responsibility are:

- An unsafe engineering variance or waiver is approved.
- Designs are approved without determining conformance with safety requirements. Waivers become routine.
- Reviews and approvals take so long that ITA becomes a bottleneck. Mission achievement is threatened. Engineers start to ignore the need for approvals and work around the STWH in other ways.

Although a long list of risks was identified in this experimental application of STPA to a management structure, many of the risks for different participants in the ITA process were closely related. The risks listed for each participant are related to his or her particular role and responsibilities and therefore those with related roles or responsibilities will generate related risks. The relationships were made clear in the earlier step tracing from system requirements to the roles and responsibilities for each of the components of the ITA.

Coordination Risks

Coordination risks arise when multiple people or groups control the same process. The types of unsafe interactions that may result include: (1) both controllers assume that the other is performing the control responsibilities, and as a result nobody does, or (2) controllers provide conflicting control actions that have unintended side effects.

Potential coordination risks are identified by the mapping from the system requirements to the component requirements used in the gap analysis described earlier. When similar responsibilities related to the same system requirement are identified, the potential for new coordination risks needs to be considered.

As an example, the original ITA design documentation was ambiguous about who had the responsibility for performing many of the safety engineering functions. Safety engineering had previously been the responsibility of the Center Safety and Mission Assurance Offices but the plan envisioned that these functions would shift to the ITA in the new organization leading to several obvious risks.

Another example involves the transition of responsibility for the production of standards to the ITA from the NASA Headquarters Office of Safety and Mission Assurance (OSMA). In the plan, some of the technical standards responsibilities were retained by OSMA, such as the technical design standards for human rating spacecraft and for conducting hazard analyses, while others were shifted to the ITA without a clear demarcation of who was responsible for what. At the same time, responsibilities for the assurance that the plans are followed, which seems to logically belong to the mission assurance group, were not cleanly divided. Both overlaps raised the potential for some functions not being accomplished or conflicting standards being produced.

8.6.4 Use of the Analysis and Potential Extensions

While risk mitigation and control measures could be generated from the list of risks themselves, the application of step 2 of STPA to identify causes of the risks will help to provide better control measures in the same way STPA step 2 plays a similar role in physical systems. Taking the responsibility of the System Technical Warrant Holder to approve all variances and waivers to technical requirements in the example above, potential causes for approving an unsafe engineering variance or waiver include: inadequate or incorrect information about the safety of the action, inadequate training, bowing to pressure about programmatic concerns, lack of support from management, inadequate time or resources to evaluate the requested variance properly, and so on. These causal factors were generated using the generic factors in figure 8.6 but defined in a more appropriate way. Stringfellow has examined in more depth how STPA can be applied to organizational factors [195].

The analysis can be used to identify potential changes to the safety control structure (the ITA program) that could eliminate or mitigate identified risks. General design principles for safety are described in the next chapter.

A goal of the NASA risk analysis was to determine what to include in a planned special assessment of the ITA early in its existence. To accomplish the same goal, the MIT group categorized their identified risks as (1) immediate, (2) long-term, or (3) controllable by standard ongoing processes. These categories were defined in the following way:

Immediate concern: An immediate and substantial concern that should be part of a near-term assessment.

Longer-term concern: A substantial longer-term concern that should potentially be part of future assessments; as the risk will increase over time or cannot be evaluated without future knowledge of the system or environment behavior.

Standard process: An important concern that should be addressed through standard processes, such as inspections, rather than an extensive special assessment procedure.

This categorization allowed identifying a manageable subset of risks to be part of the planned near-term risk assessment and those that could wait for future assessments or could be controlled by on-going procedures. For example, it is important to assess immediately the degree of "buy-in" to the ITA program. Without such support, ITA cannot be sustained and the risk of dangerous decision making is very high. On the other hand, the ability to find appropriate successors to the current warrant holders is a longer-term concern identified in the STAMP-based risk analysis that would be difficult to assess early in the existence of the new ITA control structure. The performance of the current technical warrant holders, for example, is one factor that will have an impact on whether the most qualified people will want the job in the future.

8.6.5 Comparisons with Traditional Programmatic Risk Analysis Techniques

The traditional risk analysis performed by NASA on ITA identified about one hundred risks. The more rigorous, structured STAMP-based analysis—done independently and without any knowledge of the results of the NASA process—identified about 250 risks, all the risks identified by NASA plus additional ones. A small part of the difference was related to the consideration by the STAMP group of more components in the safety control structure, such as the NASA administrator, Congress, and the Executive Branch (White House). There is no way to determine whether the other additional risks identified by the STAMP-based process were simply missed in the NASA analysis or were discarded for some reason.

The NASA analysis did not include a causal analysis of the risks and thus no comparison is possible. Their goal was to determine what should be included in the upcoming ITA risk assessment process and thus was narrower than the STAMP demonstration risk analysis effort.

8.7 Reengineering a Sociotechnical System: Pharmaceutical Safety and the Vioxx Tragedy

The previous section describes the use of STPA on the management structure of an organization that develops and operates high-tech systems. STPA and other types of analysis are potentially also applicable to social systems. This section provides an example using pharmaceutical safety.

Couturier has performed a STAMP-based causal analysis of the incidents associated with the introduction and withdrawal of Vioxx [43]. Once the causes of such losses are determined, changes need to be made to prevent a recurrence. Many suggestions for changes as a result of the Vioxx losses (for example, [6, 66, 160, 190]) have been proposed. After the Vioxx recall, three main reports were written by the Government Accountability Office (GAO) [73], the Institute of Medicine (IOM) [16], and one commissioned by Merck. The publication of these reports led to two waves of changes, the first initiated within the FDA and the second by Congress in

the form of a new set of rules called FDAAA (FDA Amendments Act). Couturier [43, 44], with inputs from others,[4] used the Vioxx events to demonstrate how these proposed and implemented policy and structural changes could be analyzed to predict their potential effectiveness using STAMP.

8.7.1 The Events Surrounding the Approval and Withdrawal of Vioxx

Vioxx (Rofecoxib) is a prescription COX-2 inhibitor manufactured by Merck. It was approved by the Food and Drug Administration (FDA) in May 1999 and was widely used for pain management, primarily from osteoarthritis. Vioxx was one of the major sources of revenue for Merck while on the market: It was marketed in more than eighty countries with worldwide sales totaling $2.5 billion in 2003.

In September 2004, Merck voluntarily withdrew the drug from the market because of safety concerns: The drug was suspected to increase the risk of cardiovascular events (heart attacks and stroke) for the patients taking it long term at high dosages. Vioxx was one of the most widely used drugs ever to be withdrawn from the market. According to an epidemiological study done by Graham, an FDA scientist, Vioxx has been associated with more than 27,000 heart attacks or deaths and may be the "single greatest drug safety catastrophe in the history of this country or the history of the world" [76].

The important question to be considered is how did such a dangerous drug get on the market and stay there so long despite warnings of problems and how can this type of loss be avoided in the future.

The major events that occurred in this saga start with the discovery of the Vioxx molecule in 1994. Merck sought FDA approval in November 1998.

In May 1999 the FDA approved Vioxx for the relief of osteoarthritis symptoms and management of acute pain. Nobody had suggested that the COX-2 inhibitors are more effective than the classic NSAIDS in relieving pain, but their selling point had been that they were less likely to cause bleeding and other digestive tract complications. The FDA was not convinced and required that the drug carry a warning on its label about possible digestive problems. By December, Vioxx had more than 40 percent of the new prescriptions in its class.

In order to validate their claims about Rofecoxib having fewer digestive system complications, Merck launched studies to prove their drugs should not be lumped with other NSAIDS. The studies backfired.

In January 1999, before Vioxx was approved, Merck started a trial called VIGOR (Vioxx Gastrointestinal Outcomes Research) to compare the efficacy and adverse

4. Many people provided input to the analysis described in this section, including Stan Finkelstein, John Thomas, John Carroll, Margaret Stringfellow, Meghan Dierks, Bruce Psaty, David Wierz, and various other reviewers.

effects of Rofecoxib and Naproxen, an older nonsteroidal anti-inflammatory drug or NSAID. In March 2000, Merck announced that the VIGOR trial had shown that Vioxx was safer on the digestive tract than Naproxen, but it doubled the risk of cardiovascular problems. Merck argued that the increased risk resulted not because Vioxx caused the cardiovascular problems but that Celebrex (the Naproxen used in the trial) protected against them. Merck continued to minimize unfavorable findings for Vioxx up to a month before withdrawing it from the market in 2004.

Another study, ADVANTAGE, was started soon after the VIGOR trial. ADVANTAGE had the same goal as VIGOR, but it targeted osteoarthritis, whereas VIGOR was for rheumatoid arthritis. Although the ADVANTAGE trial did demonstrate that Vioxx was safer on the digestive track than Naproxen, it failed to show that Rofecoxib had any advantage over Naproxen in terms of pain relief. Long after the report on ADVANTAGE was published, it turned out that its first author had no involvement in the study until Merck presented him with a copy of the manuscript written by Merck authors. This turned out to be one of the more prominent recent examples of ghostwriting of journal articles where company researchers wrote the articles and included the names of prominent researchers as authors [178].

In addition, Merck documents later came to light that appear to show the ADVANTAGE trial emerged from the Merck marketing division and was actually a "seeding" trial, designed to market the drug by putting "its product in the hands of practicing physicians, hoping that the experience of treating patients with the study drug and a pleasant, even profitable interaction with the company will result in more loyal physicians who prescribe the drug" [83].

Although the studies did demonstrate that Vioxx was safer on the digestive track than Naproxen, they also again unexpectedly found that the COX-2 inhibitor doubled the risk of cardiovascular problems. In April 2002, the FDA required that Merck note a possible link to heart attacks and strokes on Vioxx's label. But it never ordered Merck to conduct a trial comparing Vioxx with a placebo to determine whether a link existed. In April 2000 the FDA recommended that Merck conduct an animal study with Vioxx to evaluate cardiovascular safety, but no such study was ever conducted.

For both the VIGOR and ADVANTAGE studies, claims have been made that cardiovascular events were omitted from published reports [160]. In May 2000 Merck published the results from the VIGOR trial. The data included only seventeen of the twenty heart attacks the Vioxx patients had. When the omission was later detected, Merck argued that the events occurred after the trial was over and therefore did not have to be reported. The data showed a four times higher risk of heart attacks compared with Naproxen. In October 2000, Merck officially told the FDA about the other three heart attacks in the VIGOR study.

Merck marketed Vioxx heavily to doctors and spent more than $100 million a year on direct-to-the-consumer advertising using popular athletes including Dorothy Hamill and Bruce Jenner. In September 2001, the FDA sent Merck a letter warning the company to stop misleading doctors about Vioxx's effect on the cardiovascular system.

In 2001, Merck started a new study called APPROVe (Adenomatous Polyp PRevention On Vioxx) in order to expand its market by showing the efficacy of Vioxx on colorectal polyps. APPROVe was halted early when the preliminary data showed an increased relative risk of heart attacks and strokes after eighteen months of Vioxx use. The long-term use of Rofecoxib resulted in nearly twice the risk of suffering a heart attack or stroke compared to patients receiving a placebo.

David Graham, an FDA researcher, did an analysis of a database of 1.4 million Kaiser Permanente members and found that those who took Vioxx were more likely to suffer a heart attack or sudden cardiac death than those who took Celebrex, Vioxx's main rival. Graham testified to a congressional committee that the FDA tried to block publication of his findings. He described an environment "where he was 'ostracized'; 'subjected to veiled threats' and 'intimidation.'" Graham gave the committee copies of email that support his claims that his superiors at the FDA suggested watering down his conclusions [178].

Despite all their efforts to deny the risks associated with Vioxx, Merck withdrew the drug from the market in September 2004. In October 2004, the FDA approved a replacement drug for Vioxx by Merck, called Arcoxia.

Because of the extensive litigation associated with Vioxx, many questionable practices in the pharmaceutical industry have come to light [6]. Merck has been accused of several unsafe "control actions" in this sequence of events, including not accurately reporting trial results to the FDA, not having a proper control board (DSMB) overseeing the safety of the patients in at least one of the trials, misleading marketing efforts, ghostwriting journal articles about Rofecoxib studies, and paying publishers to create fake medical journals to publish favorable articles [45]. Post-market safety studies recommended by the FDA were never done, only studies directed at increasing the market.

8.7.2 Analysis of the Vioxx Case

The hazards, system safety requirements and constraints, and documentation of the safety control structure for pharmaceutical safety were shown in chapter 7. Using these, Couturier performed several types of analysis.

He first traced the system requirements to the responsibilities assigned to each of the components in the safety control structure, that is, he performed a gap analysis as described above for the NASA ITA risk analysis. The goal was to check that at least one controller was responsible for enforcing each of the safety requirements, to identify when multiple controllers had the same responsibility, and to study each

of the controllers independently to determine if they are capable of carrying out their assigned responsibilities.

In the gap analysis, no obvious gaps or missing responsibilities were found, but multiple controllers are in charge of enforcing some of the same safety requirements. For example, the FDA, the pharmaceutical companies, and physicians are all responsible for monitoring drugs for adverse events. This redundancy is helpful if the controllers work together and share the information they have. Problems can occur, however, if efforts are not coordinated and gaps occur.

The assignment of responsibilities does not necessarily mean they are carried out effectively. As in the NASA ITA analysis, potentially inadequate control actions can be identified using STPA step 1, potential causes identified using step 2, and controls to protect against these causes designed and implemented. Contextual factors must be considered such as external or internal pressures militating against effective implementation or application of the controls. For example, given the financial incentives involved in marketing a blockbuster drug—Vioxx in 2003 provided $2.5 billion, or 11 percent of Merck's revenue [66]—it may be unreasonable to expect pharmaceutical companies to be responsible for drug safety without strong external oversight and controls or even to be responsible at all: Suggestions have been made that responsibility for drug development and testing be taken away from the pharmaceutical manufacturers [67].

Controllers must also have the resources and information necessary to enforce the safety constraints they have been assigned. Physicians need information about drug safety and efficacy that is independent from the pharmaceutical company representatives in order to adequately protect their patients. One of the first steps in performing an analysis of the drug safety control structure is to identify the contextual factors that can influence whether each component's responsibilities are carried out and the information required to create an accurate process model to support informed decision making in exercising the controls they have available to carry out their responsibilities.

Couturier also used the drug safety control structure, system safety requirements and constraints, the events in the Vioxx losses, and STPA and system dynamics models (see appendix D) to investigate the potential effectiveness of the changes implemented after the Vioxx events to control the marketing of unsafe drugs and the impact of the changes on the system as a whole. For example, the Food and Drug Amendments Act of 2007 (FDAAA) increased the responsibilities of the FDA and provided it with new authority. Couturier examined the recommendations from the FDAAA, the IOM report, and those generated from his STAMP causal analysis of the Vioxx events.

System dynamics modeling was used to show the relationship among the contextual factors and unsafe control actions and the reasons why the safety control structure migrated toward ineffectiveness over time. Most modeling techniques provide

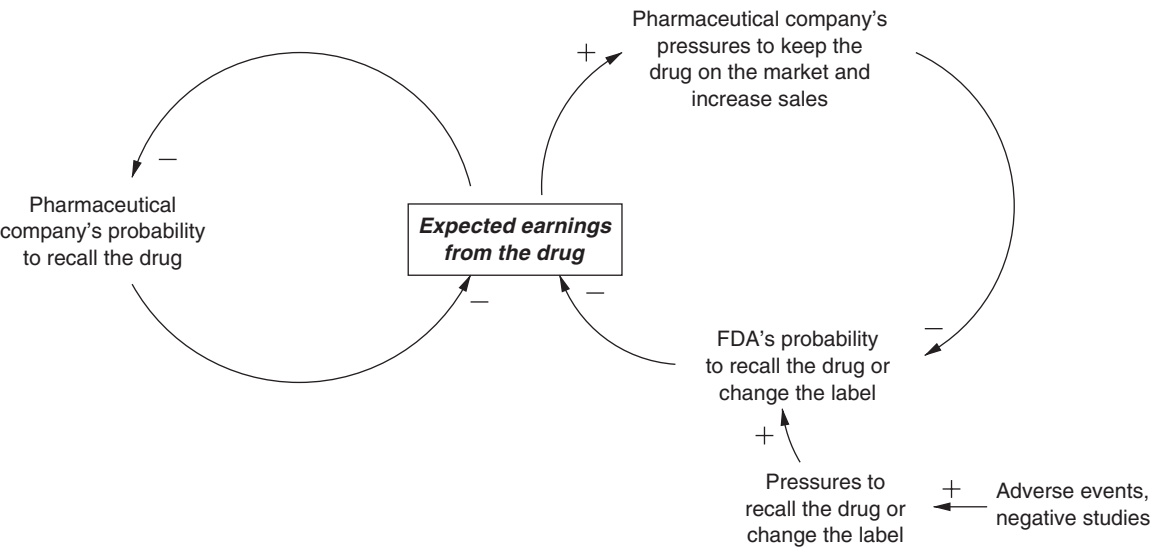

Figure 8.10
Overview of reinforcing pressures preventing the recall of drugs.

only direct relationships (arrows), which are inadequate to understand the indirect relationships between causal factors. System dynamics provides a way to show such indirect and nonlinear relationships. Appendix D explains this modeling technique.

First, system dynamics models were created to model the contextual influences on the behavior of each component (patients, pharmaceutical companies, the FDA, and so on) in the pharmaceutical safety control structure. Then the models were combined to assist in understanding the behavior of the system as a whole and the interactions among the components. The complete analysis can be found in [43] and a shorter paper on some of the results [44]. An overview and some examples are provided here.

Figure 8.10 shows a simple model of two types of pressures in this system that militate against drugs being recalled. The loop on the left describes pressures within the pharmaceutical company related to drug recalls while the loop on the right describes pressures on the FDA related to drug recalls.

Once a drug has been approved, the pharmaceutical company, which invested large resources in developing, testing, and marketing the drug, has incentives to maximize profits from the drug and keep it on the market. Those pressures are accentuated in the case of expected blockbuster drugs where the company's financial well-being potentially depends on the success of the product. This goal creates a reinforcing loop within the company to try to keep the drug on the market. The company also has incentives to pressure the FDA to increase the number of approved

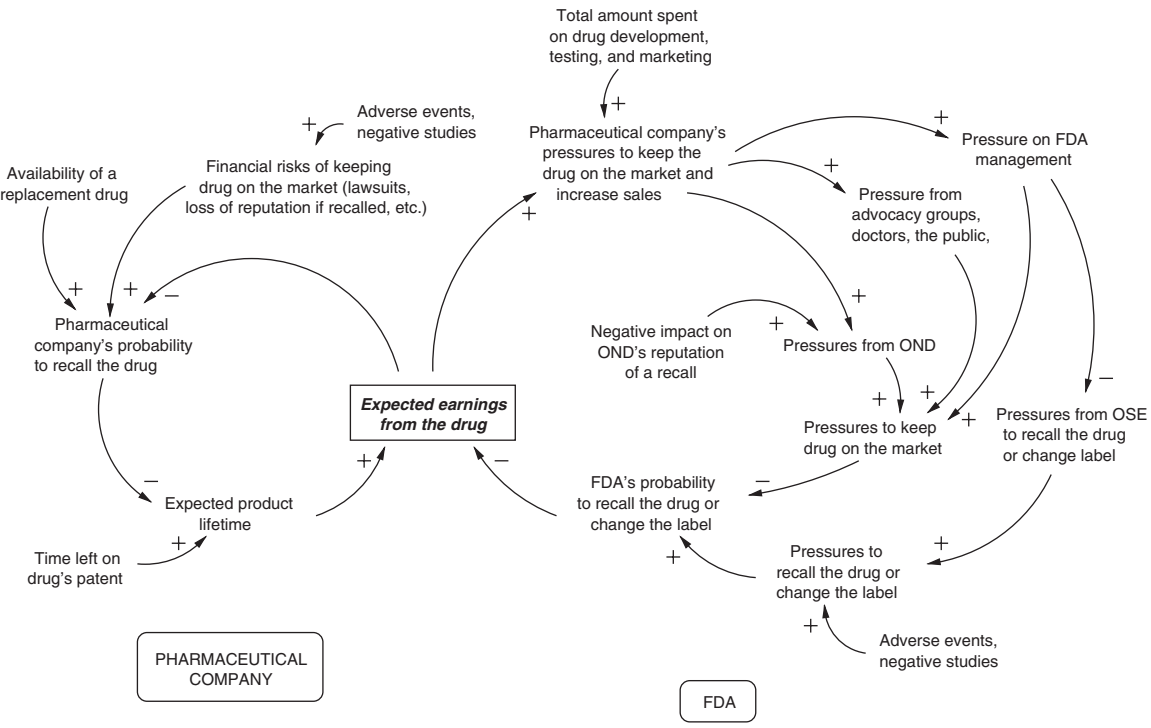

Figure 8.11
A more detailed model of reinforcing pressures preventing the recall of drugs.

indications, and thus purchasers, resist label changes, and prevent drug recalls. If the company is successful at preventing recalls, the expectations for the drug increase, creating another reinforcing loop. External pressures to recall the drug limit the reinforcing dynamics, but they have a lot of inertia to overcome.

Figure 8.11 includes more details, more complex feedback loops, and more outside pressures, such as the availability of a replacement drug, the time left on the drug's patent, and the amount of time spent on drug development. Pressures on the FDA from the pharmaceutical companies are elaborated including the pressures on the Office of New Drugs (OND) through PDUFA fees,[5] pressures from advisory boards

5. The Prescription Drug Use Fee Act (PDUFA) was first passed by Congress in 1992. It allows the FDA to collect fees from the pharmaceutical companies to pay the expenses for the approval of new drugs. In return, the FDA agrees to meet drug review performance goals. The main goal of PDUFA is to accelerate the drug review process. Between 1993 and 2002, user fees allowed the FDA to increase by 77 percent the number of personnel assigned to review applications. In 2004, more than half the funding for the CDEH was coming from user fees [148]. A growing group of scientists and regulators have expressed fears that in allowing the FDA to be sponsored by the pharmaceutical companies, the FDA has shifted its priorities to satisfying the companies, its "client," instead of protecting the public.

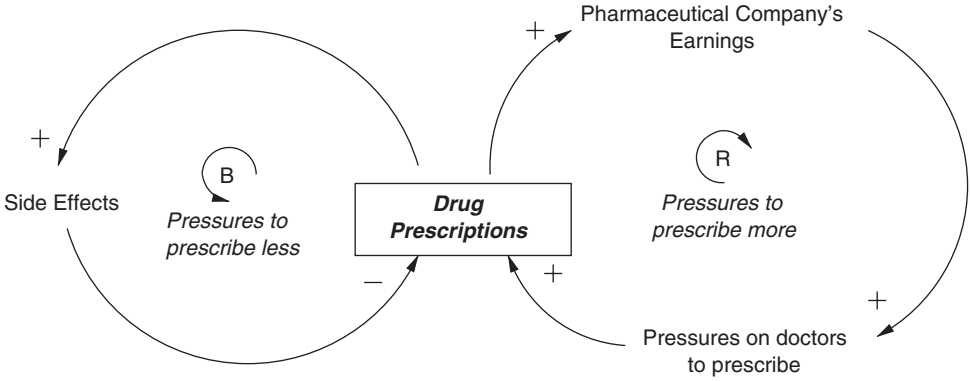

Figure 8.12
Overview of influences on physician prescriptions.

to keep the drug (which are, in turn, subject to pressures from patient advocacy groups and lucrative consulting contracts with the pharmaceutical companies), and pressures from the FDA Office of Surveillance and Epidemiology (OSE) to recall the drug.

Figures 8.12 and 8.13 show the pressures leading to overprescribing drugs. The overview in figure 8.12 has two primary feedback loops. The loop on the left describes pressures to lower the number of prescriptions based on the number of adverse events and negative studies. The loop on the right shows the pressures within the pharmaceutical company to increase the number of prescriptions based on company earnings and marketing efforts.

For a typical pharmaceutical product, more drug prescriptions lead to higher earnings for the drug manufacturer, part of which can be used to pay for more advertising to get doctors to continue to prescribe the drug. This reinforcing loop is usually balanced by the adverse effects of the drug. The more the drug is prescribed, the more likely is observation of negative side effects, which will serve to balance the pressures from the pharmaceutical companies. The two loops then theoretically reach a dynamic equilibrium where drugs are prescribed only when their benefits outweigh the risks.

As demonstrated in the Vioxx case, delays within a loop can significantly alter the behavior of the system. By the time the first severe side effects were discovered, millions of prescriptions had been given out. The balancing influences of the side-effects loop were delayed so long that they could not effectively control the reinforcing pressures coming from the pharmaceutical companies. Figure 8.13 shows how additional factors can be incorporated including the quality of collected data, the market size, and patient drug requests.

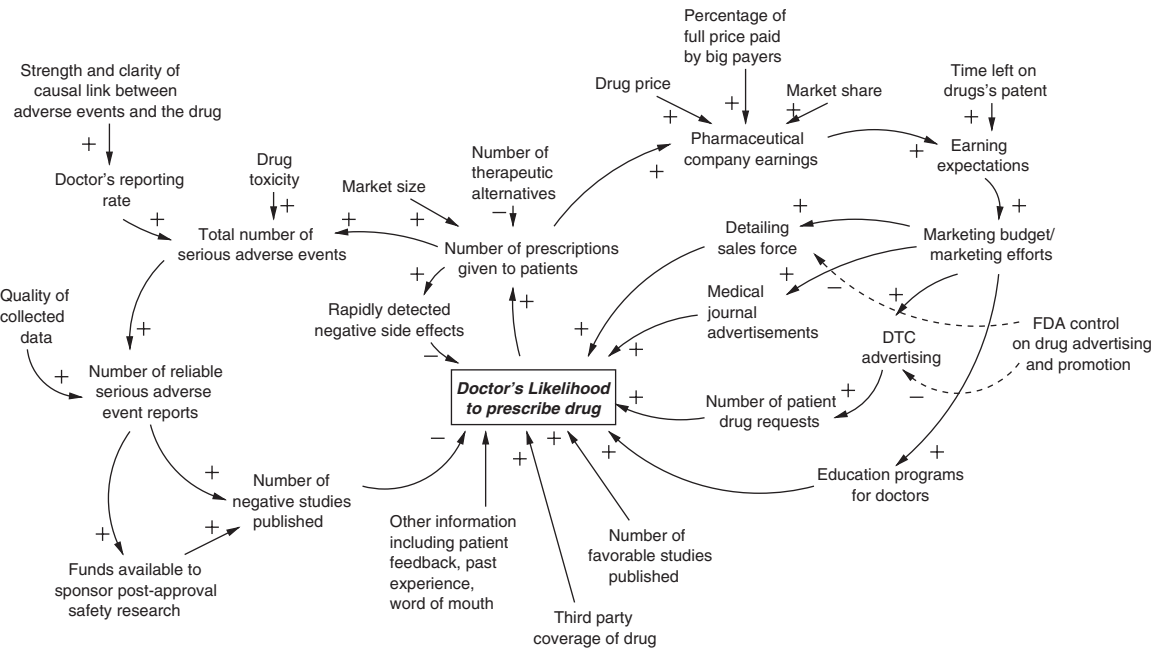

Figure 8.13
A more detailed model of physician prescription behavior.

Couturier incorporated into the system dynamics models the changes that were proposed by the IOM after the Vioxx events, the changes actually implemented in FDAAA, and the recommendations coming out of the STAMP-based causal analysis. One major difference was that the STAMP-based recommendations had a broader scope. While the IOM and FDAAA changes focused on the FDA, the STAMP analysis considered the contributions of all the components of the pharmaceutical safety control structure to the Vioxx events and the STAMP causal analysis led to recommendations for changes in nearly all of them.

Couturier concluded, not surprisingly, that most of the FDAAA changes are useful and will have the intended effects. He also determined that a few may be counterproductive and others need to be added. The added ones come from the fact that the IOM recommendations and the FDAAA focus on a single component of the system (the FDA). The FDA does not operate in a vacuum, and the proposed changes do not take into account the safety role played by other components in the system, particularly physicians. As a result, the pressures that led to the erosion of the overall system safety controls were left unaddressed and are likely to lead to changes in the system static and dynamic safety controls that will undermine the improvements implemented by FDAAA. See Couturier [43] for the complete results.

A potential contribution of such an analysis is the ability to consider the impact of multiple changes within the entire safety control structure. Less than effective controls may be implemented when they are created piecemeal to fix a current set of adverse events. Existing pressures and influences, not changed by the new procedures, can defeat the intent of the changes by leading to unintended and counterbalancing actions in the components of the safety control structure. STAMP-based analysis suggest how to reengineer the safety control structure as a whole to achieve the system goals, including both enhancing the safety of current drugs while at the same time encouraging the development of new drugs.

8.8 Comparison of STPA with Traditional Hazard Analysis Techniques

Few formal comparisons have been made yet between STPA and traditional techniques such as fault tree analysis and HAZOP. Theoretically, because STAMP extends the causality model underlying the hazard analysis, non-failures and additional causes should be identifiable, as well as the failure-related causes found by the traditional techniques. The few comparisons that have been made, both informal and formal, have confirmed this hypothesis.

In the use of STPA on the U.S. missile defense system, potential paths to inadvertent launch were identified that had not been identified by previous analyses or in extensive hazard analyses on the individual components of the system [BMDS]. Each element of the system had an active safety program, but the complexity and coupling introduced by their integration into a single system created new subtle and complex hazard scenarios. While the scenarios identified using STPA included those caused by potential component failures, as expected, scenarios were also identified that involved unsafe interactions among the components without any components actually failing—each operated according to its specified requirements, but the interactions could lead to hazardous system states. In the evaluation of this effort, two other advantages were noted:

1. The effort was bounded and predictable and assisted the engineers in scoping their efforts. Once all the control actions have been examined, the assessment is complete.
2. As the control structure is developed and the potential inadequate control actions are identified, they were able to prioritize required changes according to which control actions have the greatest role in keeping the system from transitioning to a hazardous state.

A paper published on this effort concluded:

> The STPA safety assessment methodology . . . provided an orderly, organized fashion in which to conduct the analysis. The effort successfully assessed safety risks arising from the

integration of the Elements. The assessment provided the information necessary to characterize the residual safety risk of hazards associated with the system. The analysis and supporting data provided management a sound basis on which to make risk acceptance decisions. Lastly, the assessment results were also used to plan mitigations for open safety risks. As changes are made to the system, the differences are assessed by updating the control structure diagrams and assessment analysis templates.

Another informal comparison was made in the ITA (Independent Technical Authority) analysis described in section 8.6. An informal review of the risks identified by using STPA showed that they included all the risks identified by the informal NASA risk analysis process using the traditional method common to such analyses. The additional risks identified by STPA appeared on the surface to be as important as those identified by the NASA analysis. As noted, there is no way to determine whether the less formal NASA process identified additional risks and discarded them for some reason or simply missed them.

A more careful comparison has also been made. JAXA (the Japanese Space Agency) and MIT engineers compared the use of STPA on a JAXA unmanned spacecraft (HTV) to transfer cargo to the International Space Station (ISS). Because human life is potentially involved (one hazard is collision with the International Space Station), rigorous NASA hazard analysis standards using fault trees and other analyses had been employed and reviewed by NASA. In an STPA analysis of the HTV used in an evaluation of the new technique for potential use at JAXA, all of the hazard causal factors identified by the fault tree analysis were identified also by STPA [88]. As with the BMDS comparison, additional causal factors were identified by STPA alone. These additional causal factors again involved those related to more sophisticated types of errors beyond simple component failures and those related to software and human errors.

Additional independent comparisons (not done by the author or her students) have been made between accident causal analysis methods comparing STAMP and more traditional methods. The results are described in chapter 11 on accident analysis based on STAMP.

8.9 Summary

Some new approaches to hazard and risk analysis based on STAMP and systems theory have been suggested in this chapter. We are only beginning to develop such techniques and hopefully others will work on alternatives and improvements. The only thing for sure is that applying the techniques developed for simple electromechanical systems to complex, human and software-intensive systems without fundamentally changing the foundations of the techniques is futile. New ideas are desperately needed if we are going to solve the problems and respond to the changes in the world of engineering described in chapter 1.