

Safety Assessment of Complex, Software-Intensive Systems

Nancy G. Leveson, Cody Harrison Fleming, Melissa Spencer and John Thomas
Massachusetts Institute of Technology

Chris Wilkinson
Honeywell Advanced Technology

ABSTRACT

This paper presents a new methodology for the safety assessment of complex software intensive systems such as is envisioned for the coming major upgrade of the air traffic management system known as NextGen. This methodology is based on a new, more inclusive model of accident causation called Systems Theoretic Accident Model and Process (STAMP) [1]. STAMP includes not just the standard component failure mechanisms but also the new ways that software and humans contribute to accidents in complex systems. A new hazard analysis method, called Systems Theoretic Process Analysis (STPA), is built on this theoretical foundation. The STPA is based on systems theory rather than reliability theory; it treats safety as a control problem rather than a failure problem with interactive and possibly nested control loops that may include humans. In this methodology, safety is assured by closed loop control of safety parameters.

In the NextGen Concept of Operations, [2] many diverse ground and air systems will be tightly coupled leading to a greatly increased potential for the occurrence of safety critical events. The process described in this paper provides a rigorous, integrated and traceable safety analysis that improves upon the present somewhat ad-hoc multi-layered approach commonly used today. This process also improves upon the human-system interaction aspect of safety assessment, a topic that is not well covered in present certification practice.

We illustrate the effectiveness of this new methodology by an analysis of the NextGen "In-Trail Procedure in Oceanic Airspace" (ITP) that is specified in RTCA DO-312 [3]. We show how STPA derives some additional safety requirements beyond those in the Operational Safety Analysis (OSA) of DO-312.

CITATION: Leveson, N., Fleming, C., Spencer, M., Thomas, J. et al., "Safety Assessment of Complex, Software-Intensive Systems," *SAE Int. J. Aerosp.* 5(1):2012, doi:10.4271/2012-01-2134.

INTRODUCTION

Traditional safety analysis methods focus on component failure using Probabilistic Risk Assessment (PRA) techniques such as Fault and Event Tree Analysis (FTA/ETA) and Failure Mode and Effect Analysis (FMEA) in which failures of hardware, software or humans are assigned a probability, the value of which is often based on no more than expert judgment of how often components will fail to meet specification requirements. However, software often contributes to accidents by behaving exactly as specified and humans can contribute to accidents without violating procedures. Conversely, deviating from procedures does not guarantee that an accident will result. In fact, human operators may need to deviate from procedures to prevent an accident. This makes the standard PRA techniques of limited usefulness on human or software-intensive systems especially when many such systems are tightly coupled. Failure is

neither a necessary nor sufficient condition for entering an unsafe state; an unsafe state may arise because of unsafe control actions occurring independently of failure.

Our studies of accident reports indicate that the underlying cause can often be traced to missing or incomplete system safety requirements so that subsequent verification qualifies an inherently unsafe system. In current certification practice [4], safety assessment is often carried out at the sub-system level so that complex interactions between subsystems and between sub-system and humans are missed, or common-cause effects are overlooked. In this new methodology, we include all contributors to unsafe control actions that can lead to an unsafe state or event and derive verifiable safety requirements in a systematic way that can be allocated to component sub-systems and passed down to formal or other methods of verification in accordance with current design assurance and certification standards. To illustrate this method, we analyze a particular implementation

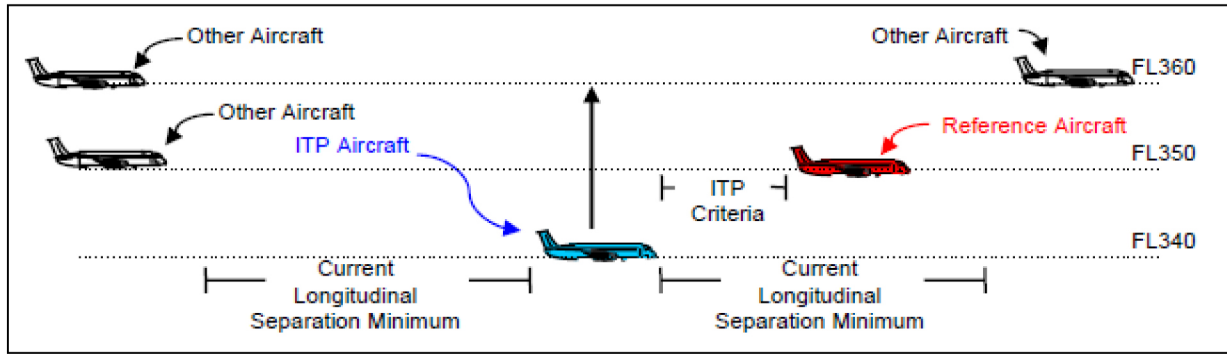


Figure 1. ITP Following-Climb [3]

of the NextGen airspace overhaul called ITP (In-Trail Procedure).

OVERVIEW OF THE IN-TRAIL PROCEDURE

Because of the limited or non-existent radar coverage in oceanic and other remote airspace, air traffic controllers (ATC) have historically relied on procedural separation rules to ensure safe traffic flow and minimum separation between aircraft. Aircraft in these sectors generally fly along pre-defined flight paths within an organized track system at constant speed and with typically 10 minute longitudinal separation. ATC have limited capability for monitoring the exact positions or separations of aircraft in a sector since position reporting by the crew is sporadic or delayed. To compensate for the limitations, these sections of the airspace often have much larger separation requirements than those applied to continental airspace, which has greater surveillance and communication coverage, creating throughput limits on these oceanic tracks. Changing flight levels allows for greater fuel efficiency due to changes in aircraft weight during the course of a flight (as fuel is burned) or because of differing optimal flight levels for different aircraft. Because of the large separation requirements, a desired flight level might often not be available due to the presence of “blocking” aircraft in intervening flight levels that fall within the standard minimum longitudinal separation distance. The ITP will allow many of these previously blocked flight level changes to occur. ITP enables either leading or following “Same Track” aircraft to perform a climb or descent to a requested flight level through an intervening flight level. Level changes are currently restricted to two flight levels. The crew will use information derived on the aircraft to determine if the criteria for executing the ITP are met with respect to one or two Reference Aircraft at intervening flight levels. Note that the standard separation minimum between aircraft does not hold during the ITP maneuver but the ITP equipment provides information to the flight crew for them to ensure that the ITP-defined reduced separation minimum are observed.

In the ITP, the flight crew first determines if the criteria defined in DO-312 for an ITP clearance request are met. If they are, and a standard clearance is not possible because the standard separations are not met, the flight crew may request a special ITP clearance, identifying the Reference Aircraft(s) in the request. ATC verifies that the reported criteria and requested Reference Aircraft(s) are valid and issues or denies the ITP clearance request. ATC does not determine or verify the actual separation distance from the Reference Aircraft. An example of the six potential ITP maneuver geometries is shown in Figure 1. In this diagram, the ITP aircraft wants to climb and pass behind the leading Reference Aircraft. In this scenario, there is only one Reference Aircraft; the trailing (other) aircraft exceeds the standard longitudinal separation minimum but other scenarios permit an ITP aircraft to designate both a leading and a trailing aircraft as Reference aircraft when both separations are below the standard minimum.

The ITP defined by DO-312 consists of four phases: the initiation phase, instruction phase, execution phase, and termination phase.

1. **Initiation phase:** The preparation for performing the application consists of realizing the desire and assessing the appropriateness for requesting an ITP maneuver by the flight crew. This includes the identification of the Reference Aircraft in the procedure and transmission of the ITP request to the ground controller.
2. **Instruction phase:** The ITP clearance is issued by the controller and reevaluated by the flight crew.
3. **Execution phase:** The cleared ITP Aircraft performs the ITP maneuver, maintaining the required rate of climb/descent and speed as directed by the ITP clearance. Conducting an ITP maneuver is similar operationally to standard climbing/descending maneuvers except that the crew must also monitor nearby aircraft.
4. **Termination phase:** The procedure is terminated once the ITP Aircraft has achieved the requested Flight Level (FL360 in our example) or an abnormal event results in premature termination of the ITP maneuver.

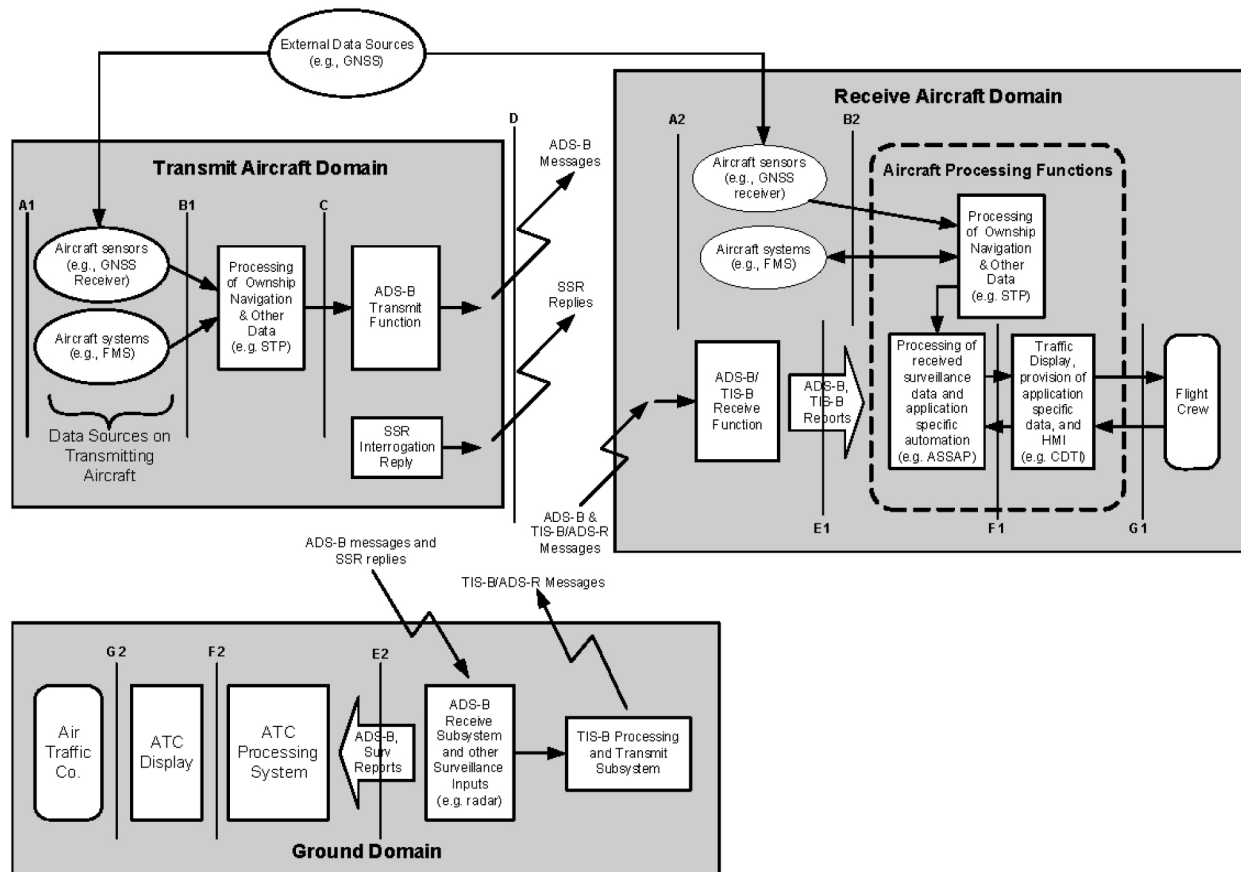


Figure 2. Surveillance Functional Architecture for ITP [3]

Figure 2 shows the surveillance functional architecture for ITP. Aircraft broadcast GPS derived position and velocity state data via an ADS-B 1090MHz transponder along with data quality indicators to all surrounding aircraft. The Receive (ITP) aircraft uses this data and its own position and velocity state data (received from its own GPS receiver) to compute the ITP criteria and to perform checks on the received data accuracy and integrity. The ITP criteria are displayed graphically to the flight crew on cockpit mounted display devices. In the currently certified Honeywell implementation ITP data is displayed on side mounted Electronic Flight Bags (EFBs) for the first officer and co-pilot. ITP criteria computations are performed by an on-board ITP function embedded within an upgraded version of the existing the traffic collision and avoidance (TCAS) computer. TCAS is a function that is mandated on all revenue flights. It provides the crew with resolution and traffic advisories on the primary flight displays if it detects a collision or near miss threat from another aircraft on an intersecting course. The combination of EFBs and ITP/TCAS is collectively the "ITP Equipment." An ITP request is transmitted to and the clearance received from the ATC ground controller using satellite-based Controller-Pilot Data Link Communications equipment (CPDLC).

OVERVIEW OF THE STAMP METHOD

The STAMP model used in our analysis [1, 5] extends the types of accidents and causes that can be considered by including non-linear, indirect and feedback relationships among events. In this way, the traditional causality model is extended to consider new types of accident causality brought about by component interactions (rather than just component failures), cognitively complex human mistakes, management and organizational errors and software errors (particularly requirements errors). Accidents or unacceptable losses can result not only from system component failures but also from interactions among system components, both physical and social, that violate system safety constraints.

In systems theory, emergent properties (like safety) associated with a set of components are related to constraints upon the degree of freedom of those components' behavior. System safety, then, can be reformulated as a system control problem rather than a component reliability problem: accidents or losses occur when component failures, external disturbances, and/or dysfunctional interactions among system components are not handled adequately or controlled; where controls may be managerial, organizational, physical, operational, or manufacturing.

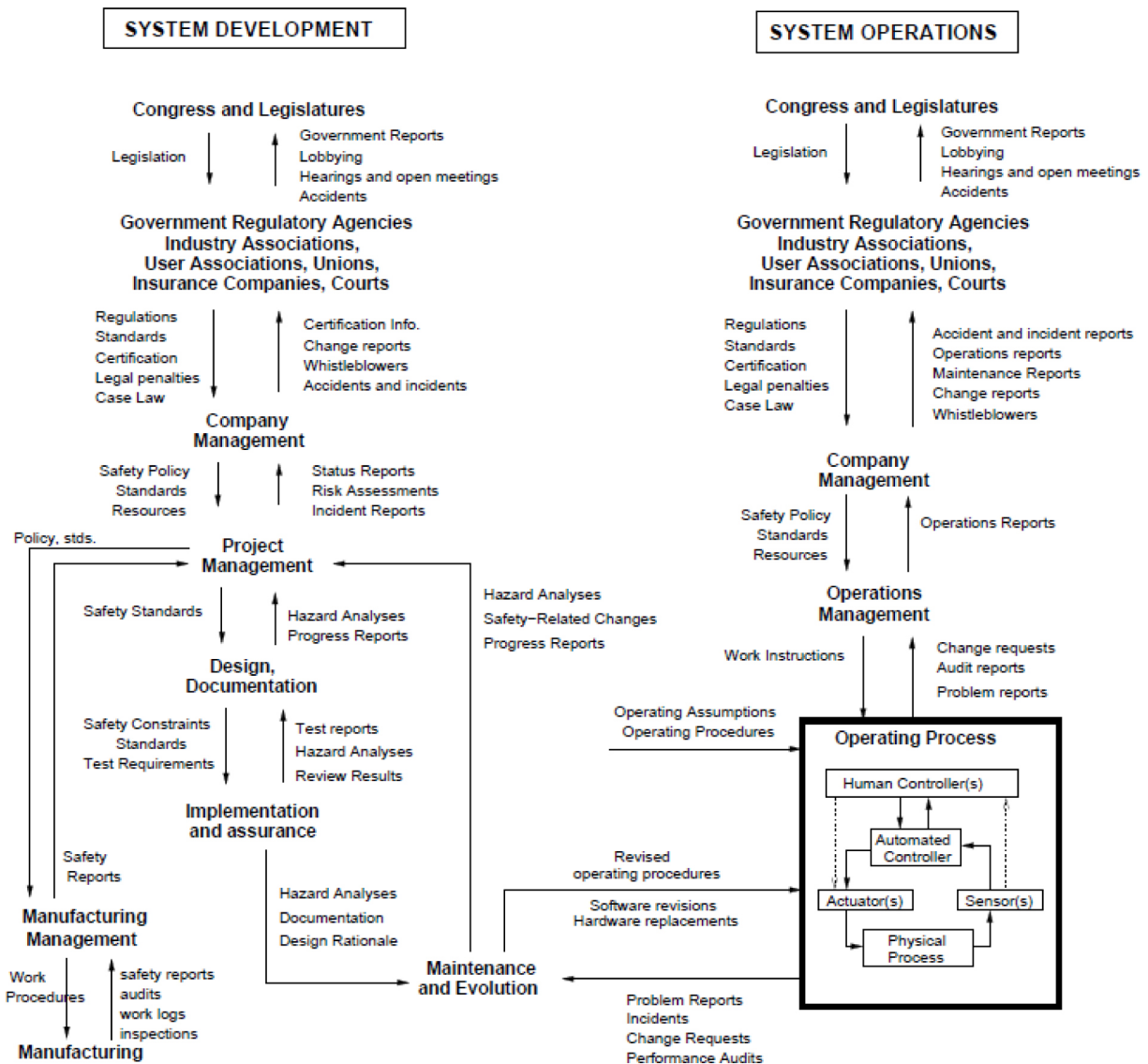


Figure 3. Example Model of a Socio-Technical Safety Control Structure

In a systems-theoretic view of safety, the emergent safety properties are controlled or enforced by a set of safety constraints related to the behavior of the system components. Safety constraints specify those relationships among system variables or components that constitute the non-hazardous or safe system states: for example, the power must never be on when the access door to the high-power source is open; two aircraft must never violate minimum separation requirements; pilots in a combat zone must be able to identify targets as hostile or friendly; and the public health system must prevent the exposure of the public to contaminated water and food products. Accidents result from interactions among system components that violate these constraints; in other words, from a lack of appropriate constraints (i.e. requirements) on component and system behavior.

STPA is the hazard analysis procedure used to identify the system constraints necessary to ensure safe development and

operation of complex socio-technical systems. The results are shown using a model-based specification method, called Intent Specifications [6, 7] that was partially developed for the certification of TCAS II [8]. Intent Specifications capture the results of the hazard analysis in a readable, reviewable way by people from multiple disciplines.

OVERVIEW OF THE STPA METHOD

STPA (System Theoretic Process Analysis) is a hazard analysis technique built on STAMP. As described above, accidents are viewed in STAMP as resulting from inadequate enforcement of constraints on system behavior. Figure 3 shows a generic safety control structure in place to enforce safety constraints. Each hierarchical level of the control structure represents a control process and control loop with actions and feedback. Two control structures are shown; system development (on the left) and system operations (on