Enforcing feature availability

Avi Harel https://avi.har-el.com/

Ergolight consulting ergolight@gmail.com

Abstract

The article suggests a model, based on four case studies, of monitor-controller-server interaction for describing the risks of over-constraining the accessibility to critical controls. The design goal proposed is to eliminate the risks of uncoordinated activity. The conclusion is that the operation should be based on carefully adjusted rules for ensuring the availability of critical features. A protocol of scenario-based interaction may be employed to ensure that the interaction is always coordinated.

Objective

Typically, accidents are commonly attributed to decision errors made by the human operators. The article presents a model of operational errors, and a framework for eliminating these errors by integration engineering.

Case studies

The model was obtained by analysis of investigation reports of four well investigated and published accidents: TMI (1979), Torrey Canyon (1967), AF 296 (1988), and PL603 (1996)

Three Miles Island (TMI) -1979

The case study is of the backup pump which was not available.

The TMI accident was a partial meltdown of a reactor of the Three Mile Island nuclear power station in Pennsylvania. The trigger for the exceptional condition was cutting off the backup pump, and subsequently, forgetting to reopen it. The valve was closed but nobody noticed it. The impact on the industry of nuclear power stations was dramatically slowing down of the industry for years.

The second trigger, two days later, was a contamination of the line control while trying to release a stuck valve. Due to the contamination, the control of the main loop failed, and the primary pump stopped working. The system was designed such that the backup pumps automatically turn on in case of such events. However, the pump was not available when needed, as it was disconnected from the system.

At this stage, the Pressure Operated Release Valve (PORV) was automatically open to release the pressure, but due to a malfunction, it did not close back. The operators became confused because the indication of the PORV state was misleading.

This case study was reported by Harel (A, 2024) based on published information.

Torrey Canyon - 1967

The case study is of the supertanker rudder control which was not available.

The Torrey Canyon oil spill was one of the world's most serious oil spills. The supertanker SS Torrey Canyon ran aground on the Pollard's Rock off the southwest coast of the United Kingdom in 1967, spilling an estimated 25–36 million gallons of crude oil.

The accident, one of the first major oil tanker accidents, had a devastating impact on the environment, killing thousands of seabirds and other marine life. It also led to significant changes in maritime law and oil spill responses.

The accident can be traced back to several contributing factors. The primary cause was a course deviation due to a drift, followed by failure to access the rudder control. The failure was due to a slip, of setting a special control state instead of the manual control state. In the special control state, intended to be used for maintenance, the rudder was disconnected from the wheel.

This case study was reported by Harel (B, 2024) based on published information.

AF296 – 1988

The case study is of the disabled thrust control

Air France Flight 296Q was a chartered flight of a new Airbus A320-111. On 26 June 1988, the plane crashed at the Habsheim airshow while making a low pass over the airfield. This was the first fly-by-wire.

The low-speed flyover, with landing gear down, was supposed to take place at an altitude of 100 feet; instead, due to a problematic flight protection envelope, the booster was disabled for eight seconds. Consequently, the plane performed the flyover at 30 ft, skimmed the treetops of the forest at the end of the runway and crashed. All 136 passengers survived the initial impact, but 3 then died of smoke inhalation from the subsequent fire.

Official reports concluded that the pilots flew too low, too slow, failed to see the forest, and accidentally flew into it. The captain, Michel Asseline, disputed the report and claimed an error in the fly-by-wire computer prevented him from applying thrust and pulling up. Five individuals, including the captain and first officer, were later found guilty of involuntary manslaughter. Captain Asseline, who maintained his innocence, went on to serve ten months in prison and a further ten months of probation.

This case study was reported by Harel (C, 2024) based on published information.

PL603 – 1996

Aeroperú 603 (PL603/PLI603) was a scheduled passenger flight from Miami, Florida, to Santiago, Chile, with stopovers in Quito, Ecuador, and Lima, Peru. On October 2, 1996, the Boeing 757-23A aircraft flying the final leg of the flight crashed, killing all 70 people aboard.

Flying over water, at night, with no visual references, the pilots were unaware of their true altitude, and struggled to control and navigate the aircraft. The investigation determined that the air data computers were unable to show correct airspeed and altitude on cockpit displays because a maintenance worker had failed to remove tape covering the pitot-static system ports on the aircraft exterior.

This case study was reported by Harel (D, 2024) based on published information.

Anatomy of operational failure

Traditional root-cause analysis (RCA)

Traditional RCA is reactive, namely, specific to an incident. The goal of reactive failure RCA is to generate a model describing the root cause of the failure. In reactive RCA we look for a single main important cause of the specific accident, a unique source for the unfortunate event. The goal is to ensure that the failure does not repeat. The conclusions are often circumstantial, justified by quoting Murphy's Law.

The failure in the case studies was that the service was not available when it was needed in emergency:

- In the TMI accident, the controller was the production control unit, and the service was the operation of the backup pump. The failure was in the pump readiness for the emergency
- In the Torrey Canon accident, the controller was the helm, and the service was the rudder. The failure was in the rudder availability while in navigation
- In the AF296 accident, the controller was the side stick, and the service was the thrust. The failure was in the thrust availability to the pilot
- In the PL603 accident, the controller was the pilot, and the service was the altimeter. The failure was in the availability of the altimeter data.

In these case studies, the controller experience was of Loss of Control (LOC). The engineering challenge about LOC accidents is to enforce the service availability by design. These case studies suggest the need for a scenario-based availability plan. In proactive failure RCA we define models based on the analysis, intended for availability assurance by design. The impact of the models is demonstrated and evaluated by hypothetical application to the case studies.

Exceptions

In the case studies, the failure was due to crossing the safety boundaries. The safety boundaries define the transition to exceptional situations:

- In the TMI accident, the exception was disabled backup pump during energy production
- In the Torrey Canyon accident, the exception was rudder disconnected from helm during navigation
- In the AF296 accident, the exception was disabled thrust at low altitude
- In the PL603 accident, the exception was disabled altimeter following takeoff.

The problem in the case studies was that the exceptions were not defined explicitly, and the performance boundaries were fuzzy.

Invisible risks

When the exceptions are fuzzy, it is possible to divert to the exceptions, and such diversion is unnoticed. In most of the systems, most of the failures are unknown, because neither the designers nor the customers can notice them, and therefore they are not aware of them. Only when the costs are high do we bother to look for ways to prevent repeating the diversion. Only a small part of the diversions is noticed, namely, when their costs are noticeable.

In case of an incident, when the designers notice a failure, they often attribute it to an operator's error. They are obliged to pay attention to a failure only in case of an accident, namely, when the costs are extremely high.

We should assume that the number of risky situations is huge, because we can see only those that are costly, and because we do not bother to detect and investigate low-cost events.

Operational errors

Analysis of many accidents has shown that the term human error is just a name for operational failure that the human operator was not able to prevent (cf. Dekker, 2007). Examples:

- TMI – The disabling of the backup pump was regarded as a human error, instead of a design mistake, of lack of warning about the slip
- Torrey Canyon – The changing to maintenance control was regarded as a human error, instead of a design mistake, of lack of warning about the slip
- AF296 – Flying over the field was considered a pilot error, instead of a design mistake, disabling the thrust
- PL603 – The takeoff with the altimeter masked may be considered a pilot error, instead of a design mistake, of lack of warning about the slip.

To eliminate human errors, we need to understand how the operation fails. The challenge is to get enough evidence to understand how errors are generated.

Error proofing

To be on the safe side, we should protect the system from all risky situations, because we cannot tell when one of them might be disastrous. Practically, this implies that error proofing ought to be a key topic of systems engineering.

Modeling the system integration

The challenge of proactive RCA is to elicit common attributes of the case studies to obtain a model of failure which applies to other industries.

In each of the case studies, the system had two components: a human controller and a technological subsystem. In each of them, the controller could not activate a critical service, which was required in emergency.

The conclusions from the reactive failure RCA may be used proactively, to obtain a model of system failure, and subsequently, to obtain a methodology for ensuring feature availability. In the proactive version of failure RCA, we look for the enablers of the problematic sources, and for the ways of the situation diversion from normal to exceptional.

A model for describing systems such as in the case studies may comprise at least two layers. In the context of availability challenges, the top layer comprises a human supervisor and a technological subsystem, and the technological subsystem comprises a controller and services.

The top layer

The top layer comprises concrete entities and data flowing between the entities.

The top-level entities include a human supervisor and a technological subsystem. In the case studies above, the technological subsystems are:

- TMI- the power generation subsystem
- Torrey Canyon – the navigation subsystem
- AF296 – the Airbus 320A aircraft
- PL603 – the Boeing 757 socio-technical system.

The top-level data includes functions applicable to the supervisor, scenarios defined by the supervisor, and tasks defined by the supervisor, applicable to the technological subsystem.

The supervisor functions in the case studies are:

- TMI - production control
- Torrey Canyon – manual navigation
- AF296 – pull up
- PL603 – takeoff from Peru.

The scenarios defined by the supervisors are:

- TMI – normal production vs. maintenance
- Torrey Canyon – navigation vs. maintenance
- AF296 – pull up vs. protected mode
- PL603 – Takeoff at a foggy night

The subsystem tasks applicable for the case studies are direct reflections of the supervisor functions. This top-level model is typical of many case studies.

The technological subsystem

The technological subsystem comprises entities, processes, situations, and activities. In the case studies above, the entities include a controller and a server. In the case studies they are:

- TMI controller – production unit; server – backup pump
- Torrey Canyon controller – wheel at the helm; server – rudder
- AF296 controller – side stick; server – thrust
- PL603 controller – the pilot; server – altimeter.

Rule-driven coordination

A rule-based model describing proper operation may help with this task and may facilitate the implementation. In normal situations the controller and the service should be coordinated, according to safety rules. The rules are obtained in hindsight, based on the observation obtained in reactive RCA. For example, the rules defining hypothetical safe operation applicable to the case studies may be:

- TMI: in normal energy production the backup pump should be available
- Torrey Canyon: in normal navigation the rudder should be connected to the wheel
- AF296: in pulling up the side stick, the thrust should be available
- PL603: during flight, the altimeter should be functional.

To enforce the service availability when in need, the design should constrain the situations, to comply with the scenario.

Operational risks are associated with diversion from normal to exceptional situations. Exceptional situations are situations which are not supported by the design of normal operation.

Diversions

In normal operation, the system units are coordinated. A diversion from a normal situation to exceptional is called a coordination slip. An availability diversion is a change from coordinated to uncoordinated situation between a controller and a service.

A coordination slip may occur by a trigger, or through a lapse or drift. A trigger is just a name for an action diverting the operational situation from normal to exceptional. A trigger may result from a human slip, from a hardware failure, or from a software bug. In the case studies, the accidents are attributed to coordination slips due to triggers:

- Disabling a critical feature, by the human operators – in the TMI accident
- Unintentional changing the control to a maintenance-only state – in the Torrey Canyon accident
- Disabling the thrust in low altitude due to a design mistake – in the AF 296 accident.
- Disabling the altimeter due to maintenance error – in the PL603 accident.

The challenges are to prevent diversions, and to detect the diversion and to notify about it to the supervisor at the time it is generated.

Operational risks

Diversions may be classified as either expected or unexpected. Initially, before the accident, they are unexpected. In hindsight, they are expected and predictable. The challenge is to develop a model of predictable diversions and a model of unexpected diversions.

The system behavior depends on the context of the activity. For example, a control, such as a button, may activate one feature or another, depending on the state of a selector. In normal operation, we need to constrain the activity to suit the desired behavior. Operational risks are often associated with operator errors due to improper constraining:

- Over constraining might result in inability to perform a desired function (Alpha errors). The effect is LOC
- Sub constraining might enable unintentional activation of functions intended to be used in specific situations, such as at setup or in maintenance (Beta errors). The effect is unpredictable.

Both types of errors should be attributed to design mistakes. In the case studies, the first slip was due to sub constraining (Beta error), and the effect was diversion to operating in over constrained conditions (Alpha error), in which a safety feature was not available. The case studies demonstrate the effect of wrong constraint:

- In the TMI accident, the activity was sub constrained, enabling the cutting of the backup pump for maintenance
- In the Torrey Canyon accident, the activity was sub constrained, enabling doing maintenance-only activity while navigating at sea
- In the AF 296 accident the activity was over constrained, preventing the thrust control
- In the PL603 accident the activity was sub constrained, enabling the maintenance error.

The engineering challenges about LOC accidents is to enforce the service availability by design. The case studies demonstrate the need to prevent any mistakes in constraining feature availability. Such mistakes may be detected in validation testing.

Enforcing feature availability

Cross industry rules

The circumstances of the case studies above are different from each other. Still, these case studies have much in common. The impact was loss of control (LOC). These commonalities should be sufficient to enable cross-domain learning, by investigation of one of them, and through abstraction, to apply the conclusions to the others. The generic rules applicable to many industries are:

- The need: In safety-critical scenarios, the relevant safety features should be available
- The risk: Due to a diversion, the relevant safety feature might not be available when required
- Protection: In emergency scenarios, the safety features should be activated automatically.

These accidents suggest the need for scenario-based situation coordination.

Why, then, was the lesson not learnt?

Principles

The design challenge is to enforce the availability of critical features. This is a special case of a more general problem of controller service coordination: the service mode should comply with the controller scenario. A feature which is critical for the operation of a unit in a certain scenario must be coordinated with that unit, in that scenario.

The model of availability failure may be used as a baseline for a methodology for availability assurance. The case studies may demonstrate hypothetical realization of the protection principles and methods, which may be applied to systems like those in the case studies.

The desired response to the trigger does not necessarily be the same for the two sources. It may depend on urgency in resuming coordinated activity. The scenario is superior to the service mode because it reflects the task imposed by the monitor. In a scenario-based design, the service mode should adapt to the scenario. In the case studies, if the scenario has changed, the controller may activate the service automatically. If the service operators disabled it, the service should notify the controller about the change in the availability.

Diversion control

Availability diversion may result from a trigger originated by the controller or by the service. The challenge is to detect the trigger and to notify about it to the supervisor. A method used to design the coordination between processes is based on the principle of multiple layer defense, as demonstrated using the Swiss Cheese illustration. The layers of availability assurance are:

1. Preventing coordination slips
2. Exploratory decision making
3. Alerting on predictable coordination slips
4. Availability awareness
5. Rebounding from slips
6. Warning on unexpected coordination slips
7. Recovery
8. Escalation preview
9. Sustaining the slip.

Preventing coordination slips

Availability diversion may result from a trigger originated by the controller or by the service. A trigger is an action by the controller or the service, that eventually results in a diversion. The safest way to manage risks is to avoid them. To manage the risks of coordination slips, we may ensure that the system is coordinated. To prevent a diversion, we need to prevent that trigger. The design challenge is to direct

the impact of the action such that it does not result in a trigger. We may constrain the operation to comply with coordination rules. Hypothetical examples from the case studies:

- In the TMI case: on changing from maintenance to production, enforce pump availability
- In the Torrey Canyon case: on changing from maintenance to navigation, enforce rudder connection
- In the AF296 case: do not apply the protection envelope while the pilot is using the site stick
- In the PL603 case: the pilot should be aware of the altimeter situation before takeoff.

Exploratory decision making

The motivation to prefer human control over automation is the operator's advantage in decision making. However, to enable proper decisions, the operators need to understand the risks of approaching the protection envelope, and the potential impact of the optional choices. Preview information may help the operators to decide on the safe option.

Preview information may help the decision of the controller operators, by prompting to avoid the diversion and by informing them of the time remaining before an action is required. Hypothetical examples from the case studies:

- The Torrey Canyon case: On unintentional changing the navigation mode from Automated navigation to maintenance, the system could inform the captain about approaching the rock
- The AF296 case: the system could inform the pilot about the time remaining until the thrust is disabled
- The PL603 case: the control system could detect that the altimeter readings are exceptional and inform the pilot about the risks of taking off before fixing the problem.

Alerting on predictable coordination slips

Not all coordination slips may be avoided. For example, according to the human factors version of Murphy's Law, if the system enables human errors, then the operators are likely to err. To detect expected coordination slips, we need to include special probes in the activity design. The probes may be designed based on a model of the system coordination. Examples of hypothetical detection of expected slips from the case studies:

- The TMI case: on disabling the backup pump, the control system could warn the operators about the risks of operating with the backup pump disabled
- The Torrey Canyon case: On unintentional changing the navigation mode from Automated navigation to maintenance, the system could inform the captain about the risks of operating in maintenance mode
- The AF296 case: the system could inform the pilot about the risks of operating with disabled thrust
- The PL603 case: the control system could detect that the altimeter readings are exceptional and inform the pilot about the risks of taking off before fixing the problem.

The coordination may be validated continuously; however, it is more practical if it is validated only in an event of change of the controller scenario or of the service availability.

Availability awareness

It might not be sufficient that the system detects and warns about the hazard. Any delay in detecting constraint violation might enable an accident. The discipline for assuring that the operators are aware of the hazard is HCD. The case studies demonstrate the costs of detection delay:

- TMI accident, the operators were not aware of the backup pump being disabled for two days before the accident
- Torry Canyon accident, the captain was not aware of the rudder being disconnected while approaching the Isles of Scilly
- AF296, the pilot was not aware of the airplane being in the protection mode, in which the thrust was delayed
- PL603, the pilot was not aware of not having the altimeter reading before takeoff.

The design challenges are to notify about a diversion at the time it is generated.

Rebounding from slips

Not all expected coordination slips can possibly be prevented. Rebounding from exceptional situations means automated or manual reverting back to the recent normal situation. This feature may apply to human-originated slips, namely, human errors. Examples from the case studies:

- In the TMI case: on disabling the backup pump while in production, the system control may enforce reverting to the state of enabled backup pump
- In the Torrey Canyon case: on changing from navigation to maintenance, system control may revert the operational mode to navigation.

Warning on unexpected coordination slips

Not all coordination slips are expected. Diversions may be prevented only if they are predictable. When they are not predictable, the system may still detect coordination slips, and notify the operators about them. Hypothetical examples from the case studies:

- The TMI case: on unexpected disability the pump, notify the operators and the controller about the risky situation
- The Torrey Canyon case: on unintentional changing the navigation mode from Automated navigation to maintenance, the system may notify the operators when the rudder is not connected to the helm
- The AF296 case: the system may notify the pilot when the protection envelope is applied, disabling the pilot control
- The PL603 case: the system may notify the operator when the altimeter reading does not comply with the scenario.

A method for detecting unexpected situations is by risk indicator, based on segmentation of continuous system variables, such as performance variables, or time measurement of process execution or state transition. For example, in the PL603 case: the control system could detect that the altimeter readings are exceptional and inform the pilot about the risks of taking off before fixing the problem.

Recovery

Occasionally, the operators might fail to rebound from the exceptional situation. In case of a coordination failure both the controller and the service may react, depending on the source.

The reaction may be spontaneous recovery, by rebounding to the original coordinated state, or by enforcing a corrective change in the partner.

The recovery methods are by applying troubleshooting and recovery procedures, and by collaboration between the operators and the system, while in safe-mode operation.

Escalation preview

When operation in exceptional situations, it is important to notify the operators about it. In response the operators need to look for ways to recover from the exceptions. The time frame for recovery may be limited, and knowledge of the time frame is important for deciding well about the recovery procedure.

Sustaining the slip

Sometimes, the system design does not include sufficient means for troubleshooting, and the coordination practically fails. For these cases, the system should apply a last protection layer, which is by employing resilience procedures, such as safe-mode operation.

Lessons

The challenge is that safety-critical features should be available when they are needed in case of hazard. The conclusion is that it is possible to enforce the availability of safety-critical features. The design should protect from disconnecting safety critical features and should provide warning when these features are disconnected.

References

Dekker, S 2007. Just Culture: Balancing Justice with Accountability.
Harel, A 2024. (A) Enforcing feature availability: the TMI case study,
DOI: 10.13140/RG.2.2.23687.61607
Harel, A 2024. (B) Enforcing feature availability: the Torrey Canyon case study,
DOI: 10.13140/RG.2.2.25365.33766
Harel, A 2024. (C) Enforcing feature availability: the AF 296 case study,
DOI: 10.13140/RG.2.2.19703.02725
Harel, A 2024. (D) Enforcing feature availability: the PL 603 case study,
DOI: 10.13140/RG.2.2.34802.52165