安全と管理手法
～２種類のエラーと対策～

独立行政法人自動車技術総合機構交通安全環境研究所　鉄道認証室 主席研究員

森　崇

登場人物

カバ興業　社長
座右の銘：技術と直感

カバ興業　営業　カバお
「怒られてナンボの毎日が生命の危機です」

カバ鉄道　電気課長
口癖：安くてエエもん持って来い！

カバ興業　技術　オタかば
「面白くなければ技術じゃない」

謎のフリーコンサル
なぞカバ
「知識は力！管理は必然」

カバ興業　プログラマ
ハッキングカバ
「俺しかできないことをやる」

大蒲教授
「ソフトウェアは作法である！」

カバ興業の協力会社社員

「請負は、請けたら負け」

カバ興業設計課長　カバ実
「みんなできるようになりましょう」

説明したいこと

最後ですので、よく誤解されている、安全と管理手法の関係について説明します。

IEC 62278 4.7.1

5-1安全の対策

社長はシステムが危険な状態に遷移しないようにするには、どのような対策を考えていますか？

よっしゃ。答えはな「フェールセーフの考え方」や。もしウチの安全関連システムが故障を検知したらすぐ止めて安全側を維持するんや。

素晴らしいですね。他の対策はないですか？

おまえそんな。。もし故障検知してすぐ止められたら、オレは安全が担保されていると考えるで。これがウチのSIL 4システムの思想やで。

IEC 62278 4.7.1

5-1安全の対策

二つの言葉の使い方の間違いがあります。まず、カバ興業のフェールセーフの考え方ではすべての事象を解決できません。2つ目は、SILという言葉の使い方が違います。

もし御社の製品に多くのバグがあったら、それはちゃんと動きますかね。もしそれが安全関連の機能だったら、安全ですかね？

ウチの技術陣はすごく優秀やねんで。その問題はないで。

技術陣の能力！それ対策じゃないですかね。

5-1鉄道事業者の皆さんがよくやってしまう会話

とにかくSIL 4やで、ウチは人命を預かっとるんや。連動装置はSIL 4でつくれ！

さすが課長さん。当社もその思想、大事にさせてもらいますわ！

アホか社長！全部SIL 4の管理と技術で作ったらナンボかかる思てるねん！自動進路装置までSIL 4で作るんか‼安全に関係ないで!

規格はリスクを解析し、必要な安全に対してメリハリをつけてリソースを割り当てることになっています。

許容度は10^{-9}/hやで、ウチは人命を預かっとるんや。全部証拠出してくれ!

ソフトウェアのバグがどのくらい発生するかって。。困りましたね。。

じゃあソフトはエエわ。ハードウェアは頼むで。

ソフトウェアも何らかの基準は必要そうですね。。

IEC 62278 4.7.1

5-2 2つの故障種別

安全の完全性は数値化できる要素(一般的にハードウェアに関係し、例えばランダム故障)と数値化できない要素(一般的にソフトウェア、仕様、文書、プロセスなどのシステマティック故障の組み合わせとしてとらえられる。

-IEC 62278 4.7.1

ランダム故障

- 運用モードの異常遷移
- 環境による故障
- 耐性の低下による故障
- 摩耗による故障
- 耐性を超える事象による故障
- その他

ソフトウェア、仕様、文書、プロセスなどのシステマティック故障

- 要求仕様の誤り
- 設計及び実現手法の不適正
- 製造不適正
- 固有の弱点
- ソフトウェアエラー
- 運用マニュアルの不適正
- 指示書の不適正
- ヒューマンエラー
- その他

From Figure 5 – Factors Influencing Railway RAMS, IEC 62278

思い当たる節が。。。

IEC 62278 4.7.1

5-2 2つの故障種別

ランダムエラーか、システマティックエラーか、両方か答えてください。

ロジックボードが予期しない高温で壊れた！

ロジックボードが通常の温度で壊れた！

カバおがシステムの条件を理解できなかったせいで、カバ鉄道さんに合った仕様にならなかった。

テストケースがすべての機能を満たしておらず、その製品がお客様に渡ってしまった。

システムAとBの間のエラーレートが高くなって、システムが故障してしまった！

IEC 62278 4.7.1

5-2 2つの故障種別

ランダムエラーか、システマティックエラーか、両方か答えてください。

ロジックボードが予期しない高温で壊れた！

「しっかりカバ鉄道の状況把握してよ」もしくは「そんな温度で使うな」ですよね。起こるべくして起こった故障。部品のせいじゃありません。これはシステマティックエラーといえそうです。

ちゃんと温度設計していたのに、ロジックボードが通常の温度で壊れた！

これは部品がある一定の確率で壊れることから来た故障なのでランダムエラーといえそうです。

カバおがシステムの条件を理解できなかったせいで、カバ鉄道さんに合った仕様にならなかった。

「しっかりカバ鉄道の状況把握してよ」ということになりそうですが、そもそも技術的な判断が出来ない営業に全てを任せた社長に問題がありそうです。これはシステマティックエラーといえそうです。

テストケースがすべての機能を満たしておらず、その製品がお客様に渡ってしまった。

「試験内容の漏れ」ということで、部品の故障は関係ありません。これはシステマティックエラーといえそうです。

システムAとBの間のエラーレートが高くなって、システムが故障してしまった！

なんともいえません。設計上のミスかも知れませんし、部品が壊れてエラーになったのかも知れません。もう少し追加の原因究明が必要です

IEC 62278 4.7.1

5-2 2つの故障種別

信号の錯誤現示による脱線頻度は10^{-9}/h以下にしなければいかんと決めたで！

社長命令からの割り当て

機能 A: TFFR 10^{-9}/h

random failures

機能Aを実現するボードの故障頻度 10^{-9}/h

ランダム故障は数値目標を立てることができる

systematic failures in software, specification, documents, processes

機能を実現する設計の失敗頻度:???

機能Aを実現するソフトのエラー頻度:???

機能Aの検査のエラー頻度:???

:

もし書類のエラー率を決めたとしても、それをどう解決するかなんて、どうしようもないやろ。

TFFR: Tolerable Functional Failure Rate

IEC 62278 4.7.1

5-2 2つの故障種別

信号の錯誤現示による脱線頻度は10^{-9}/h以下にしなければいかんと決めたで！

社長命令からの割り当て

機能 A: TFFR 10^{-9}/h

社長の命令に基づいた管理手法と対策

random failures

機能Aを実現するボードの故障頻度 10^{-9}/h

systematic failures

機能を実現する設計:厳密な対処

機能Aを実現するソフト:厳密な対処

機能Aの検査:厳密な対処

TFFR: Tolerable Functional Failure Rate

EN 50126-2 10.2.12

5-3 SILの誤った使い方

カバ鉄道さんが、SIL 4の処理システムを準備できるか？と聞いて来たで。
そんなモン、「Yes, We Can!」しかないやろ。

おまえそんなこと言うたんか？　そんなこと言うてええんか？
アメリカ大統領選挙でも出るつもりかいな。

ウチのめっちゃスゴイ処理システム Hippo-Xは世界一なんや！誰もおいつかれへんで！　おい、ハッキングカバ！おまえはウチの商品信用してないんか。

考えてみてくださいよ。もし誰かがHippo-Xに想定処理能力を超えるアプリケーションプログラムを入れたら、ちゃんと動きますかね？

そんなアホなソフト入れるヤツは辞めてもらうで。
ウチのHippo-Xはしっかり設計されたソフトしか受け入れへんで！

おい社長、もっとカネくれたら、ソフトの品質空まで上がるで。

SILはシステムの属性として、使用してはいけないと言われています。

EN 50126-2 10.2.12

5-3 SILは機能に割り当て

SILは機能に割り当てるんやで。
システムとかサブシステムに割り当てるんちゃうで。

困るやん！ ウチは「**SIL4 システムが貴社の安全運用を守ります！**
カバさんはいつもあなたのそばに。」って広告してるんやで。。。。

カバが人間のそばにずっといるなんて、少し怖くないですかね。。。

SIL 4を要求する機能Aに合致するアプリケーションソフト Hippo-S

Hippo-Xのアプリケーション条件

SIL 4を要求する機能Aに合致する処理系ハードウェア Hippo-S

実現
方法

SIL 4 機能
機能 A

この機能が失敗すると、
重大な結果が起こる。

こういったラインアップで、
"SIL 4 機能A"を守るんやで。

5-3 安全機能とSILの割り当て

オレらいっつも「この機能はSIL 4」とか言ってるけど、どうやって決まってるんや。

多分ですけど、お客様のご意見とか、営業戦略で。。。。

そんなん本質ちゃうやろ！

EN 50126-2 10.2.2.にやり方がありますね。

5-3 コンポーネントにおけるSIL機能と対応 1

Risk assessment

TFFRとSILを決める

software SILを決める

4.3 The required software safety integrity level shall be decided and assessed at system level, on the basis of the system safety integrity level and the level of risk associated with the use of the software in the system.

機能をサブシステムやコンポーネントに割り当てる

- failure rateのコンポーネントへの割り当て
- FRに相当するSILの技術的手法との整合確保

5-3 コンポーネントにおけるSIL機能と対応 2

機能をサブシステムやコンポーネントに割り当てる

7.3.4.9 Where the software consists of components of different software safety integrity levels then all of the software components shall be treated as belonging to the highest of these levels unless there is evidence of independence between the higher software safety integrity level components and the lower software safety integrity level components.

5-3 まとめ:そんなに難しいことではありません

安全第一！お客様の死亡事故は考えられる限りなくさなアカンで！
これはカバ鉄道全員の思想や！

さすが課長さん。当社もその思想、大事にさせてもらいますわ！

じゃあ、装置の機能がミスったら乗客が死ぬようなやつは、10-100万年に一回もあったらアカンくらいの設計がいるんやな！10^{-8}~10^{-9}/hくらいということや。

じゃあ、装置の機能が失われると大変なことになる機能を、安全面と稼働率の面から、洗い出して報告して！

解析したで!鎖錠論理と信号現示論理、各装置のインターフェース機能がミスると脱線するかもしれん。稼働率は、記録装置以外ほぼ全部が影響するな!

じゃあシステム全体として、危険となる場合が5×10^{-9}/h以下となるように、各機能がどの程度の許容度があるか解析してちょうだい!

5-3まとめ:そんなに難しいことではありません

じゃあシステム全体として、危険となる場合が5×10^{-9}/h以下となるように、各機能がどの程度の許容度があるか解析してちょうだい!

かば実課長、やり手の矢継ぎ早の指示やな! 安全にかかわる機能は一時間当たりの危険発生回数は1×10^{-9}回/hとすればいけるで。これを実現できるハードウェア設計することになるな。

次にポカミスとか、バグとか、仕様書の誤りがないような管理手法や技術手法の選択をお願いするわね。危険発生が非常に少ないシステムに合ったものになるわよね。

カバおは野放しではアブナイで。ちゃんとチェックしないとな。
でも一番危ないのは、気軽になんでも客に約束してくる社長やな。課長どうする？

。。。。。

5-3まとめ:そんなに難しいことではありません

コンセプト

安全第一！お客様の死亡事故は考えられる限りなくさなアカンで！
これはカバ鉄道全員の思想や！

営業トーク(キケンだが必要?)

さすが課長さん。当社もその思想、大事にさせてもらいますわ！

じゃあ、装置の機能がミスったら乗客が死ぬようなやつは、10-100万年に一回もあったらアカンくらいの設計がいるんやな！10^{-8}~10^{-9}/hくらいということや。

システムのリスク許容程度(THR)

じゃあ、装置の機能が失われると大変なことになる機能を、安全面と稼働率の面から、洗い出して報告して！

リスクの解析

解析したで!鎖錠論理と信号現示論理、各装置のインターフェース機能がミスると脱線するかもしれん。稼働率は、記録装置以外ほぼ全部が影響するな!

じゃあシステム全体として、危険となる場合が5×10^{-9}/h以下となるように、各機能がどの程度の許容度があるか解析してちょうだい!

機能へのリスク割り当て

5-3まとめ：そんなに難しいことではありません

じゃあシステム全体として、危険となる場合が5×10^{-9}/h以下となるように、各機能がどの程度の許容度があるか解析してちょうだい!

かば実課長、やり手の矢継ぎ早の指示やな! 安全にかかわる機能は一時間当たりの危険発生回数は1×10^{-9}回/hとすればいけるで。これを実現できるハードウェア設計することになるな。

機能へのリスク許容程度(TFFR)

次にポカミスとか、バグとか、仕様書の誤りがないような管理手法や技術手法の選択をお願いするわね。危険発生が非常に少ないシステムに合ったものになるわよね。

安全度と管理・技術の割り当て(SIL)

カバおは野放しではアブナイで。ちゃんとチェックしないとな。
でも一番危ないのは、気軽になんでも客に約束してくる社長やな。課長どうする？

能力と役割の割り当て

。。。。。

終わりに

一年間ありがとうございました。
またどこかでお会いしましょう！