STAMP/STPA分析入門
～ 例題を用いた分析手順の解説 ～

一般社団法人組込みシステム技術協会（JASA）
技術本部　安全性向上委員会

一般社団法人 組込みシステム技術協会 Japan Embedded Systems Technology Association

はじめに

- STAMP/STPAの分析手順を、例題を使って解説します。
- 実際の分析の流れを、途中のQ&Aで具体的に考えていただきながら、模擬的に体験していただきます。
- 分析の流れに沿って、STAMP/STPA分析の支援ツールである「STAMP Workbench」の画面もご紹介します。
  - STAMP Workbenchの主な機能や利用イメージも簡単にご紹介します。

「STAMPガイドブック ～システム思考による安全分析～」の公開 ～STAMPの本質を理解してさらなる有効活用を～
https://www.ipa.go.jp/ikc/reports/20190329.html
付録：ガイドブックで紹介する教材 https://www.ipa.go.jp/files/000072492.zip

STAMP/STPA分析支援ツール STAMP Workbench

https://www.ipa.go.jp/sec/tools/stamp_workbench.html

一般社団法人 組込みシステム技術協会 Japan Embedded Systems Technology Association

本日の内容

- 例題の説明
- STAMP/STPA分析
  - Step 0 準備1：アクシデント、ハザードの識別
  - Step 0 準備2：コントロールストラクチャー図の作成
  - Step 1　　：UCAの識別
  - Step 2　　：UCAの要因分析

例題：アクセス・コントロール・ゲート

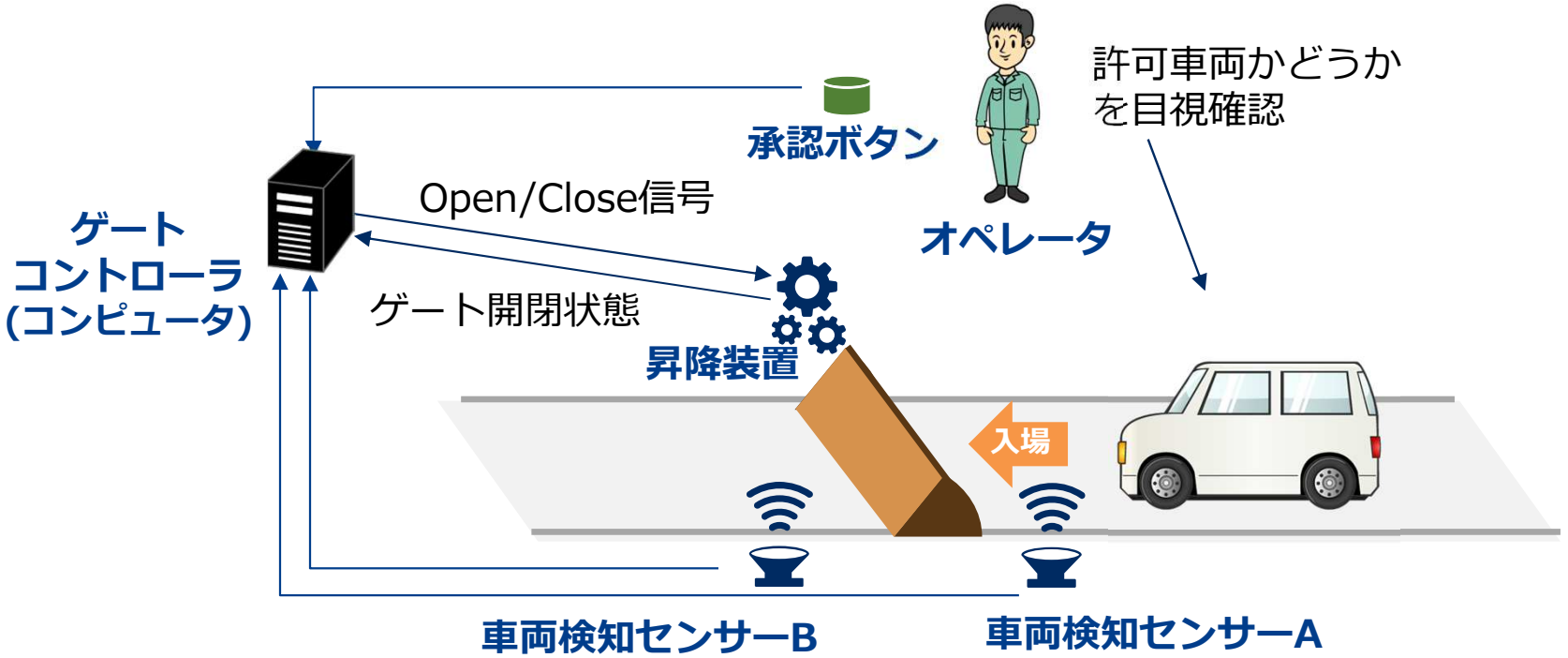
許可された車両だけを入場させ、
許可のない車両の進入を物理的にブロックする
システム

※本例題は下記資料を参照し、一部引用して作成した。
“STPA Exercise”, 第2回 STAMPワークショップ, 2017
https://www.ipa.go.jp/files/000063044.pdf

例題：アクセス・コントロール・ゲート

Aさんは、このシステムの設計を任され、下図のような仕組みで実現することを考えました。
このシステムの安全性に関する要件を明確にするために、STAMP/STPAで分析することにしました。

【概要】
- 入場車両が来ると、オペレータが目視で許可を確認して承認ボタンを押す。
- ゲートコントローラは、センサーによって車両を検知し、オペレータの指示にしたがってOpen, Close信号を発行する。



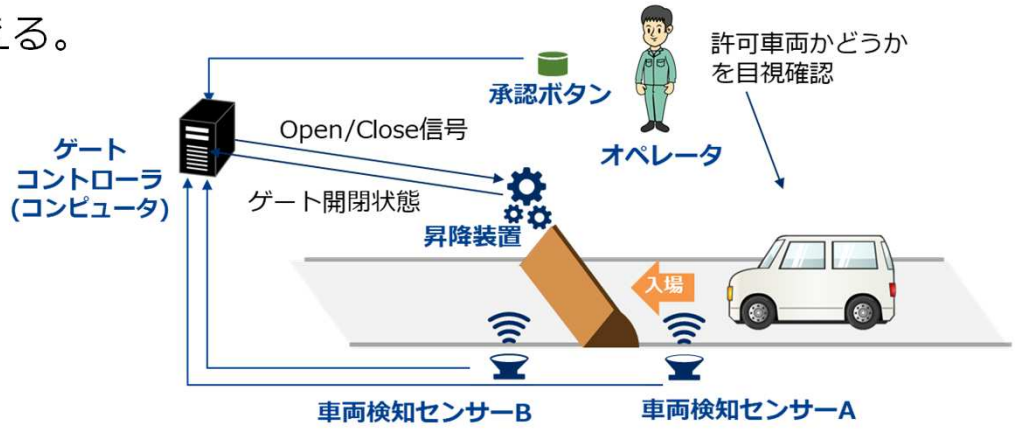
例題：Aさんが考えた仕様

■ 【オペレータ】
- 入場する車両の入場許可の有無を確認し、許可があれば承認ボタンを押す。

■ 【車両検知センサー】
- 金属検知方式により、車両検知の有無をゲートコントローラに通知する。

■ 【ゲートコントローラ（コンピュータ）】
- センサーAが車両を検知し、「承認」を受信すると、ゲートが完全に下がりきるまで昇降装置にOpen信号を送信する。
- A,B両方のセンサーから「検知無し」を受信すると、車両が通過完了と判断し、ゲートが完全に上がりきるまで昇降装置にClose信号を送信する。

■ 【昇降装置】
- 電気モーターと歯車により、ゲートを昇降させる。
- Open信号を受信している間はゲートを下降させるように動作し、Close信号を受信している間はゲートを上昇させるように動作する。
- ゲートの開閉状態をゲートコントローラに伝える。



STAMP/STPAの分析手順

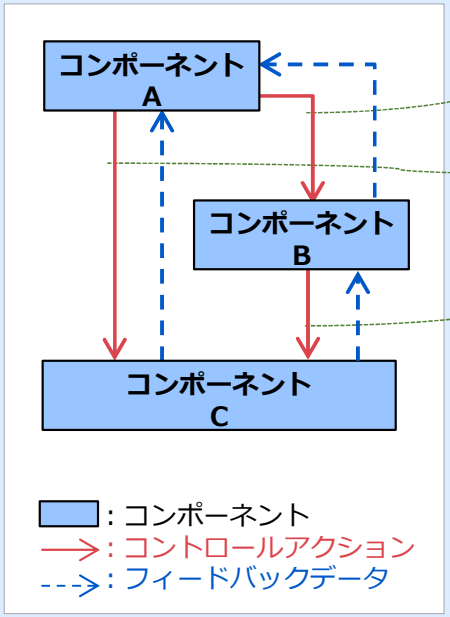
【Step0 】準備1

アクシデント、
ハザード、
安全制約の識別

【Step0】準備2

コンポーネント間の
制御関係を表すモデル※
の構築

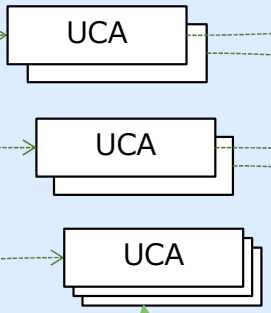
※コントロールストラクチャー



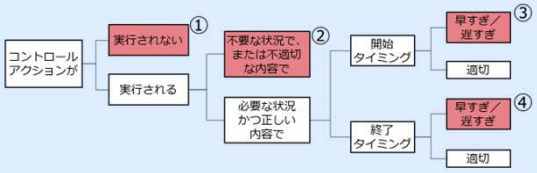
【Step1】

ハザードにつながる
コントロールアクション※
の識別

※UCA
(Unsafe Control Action)



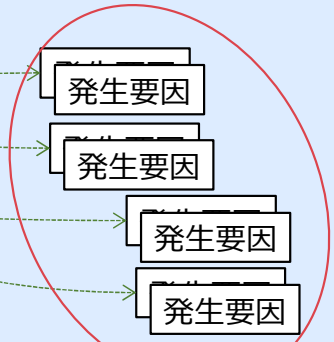
危険な制御の
パターンを利用して分析



【Step2】

UCAの発生要因※
の分析

※HCF
(Hazard Causal Factor)



コンポーネント
が満たすべき
安全要件を導出

一般社団法人
組込みシステム技術協会
Japan Embedded Systems Technology Association

Step0 準備1：
アクシデント、ハザード、安全制約の識別

【Step0 】
準備1

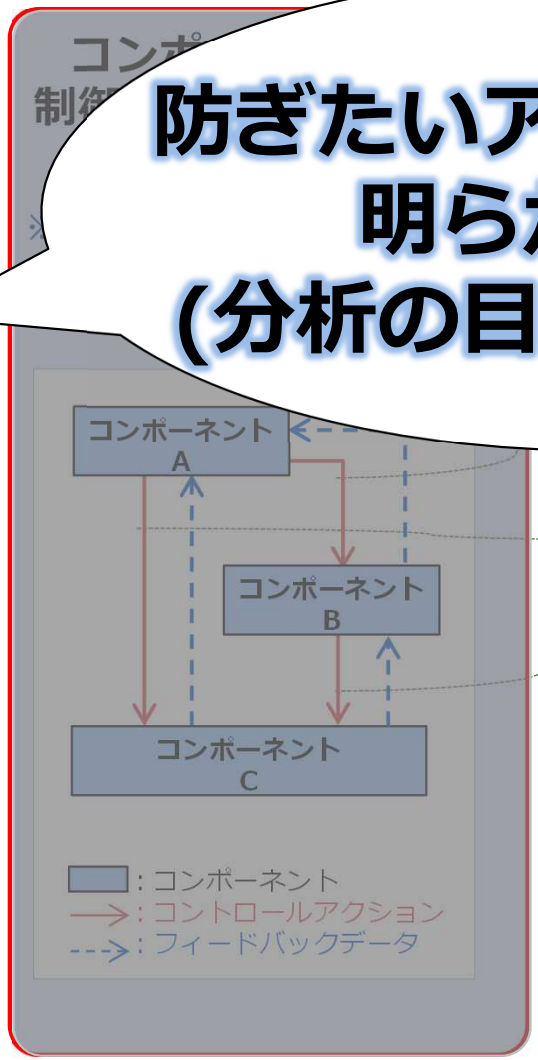
【Step0】
準備2

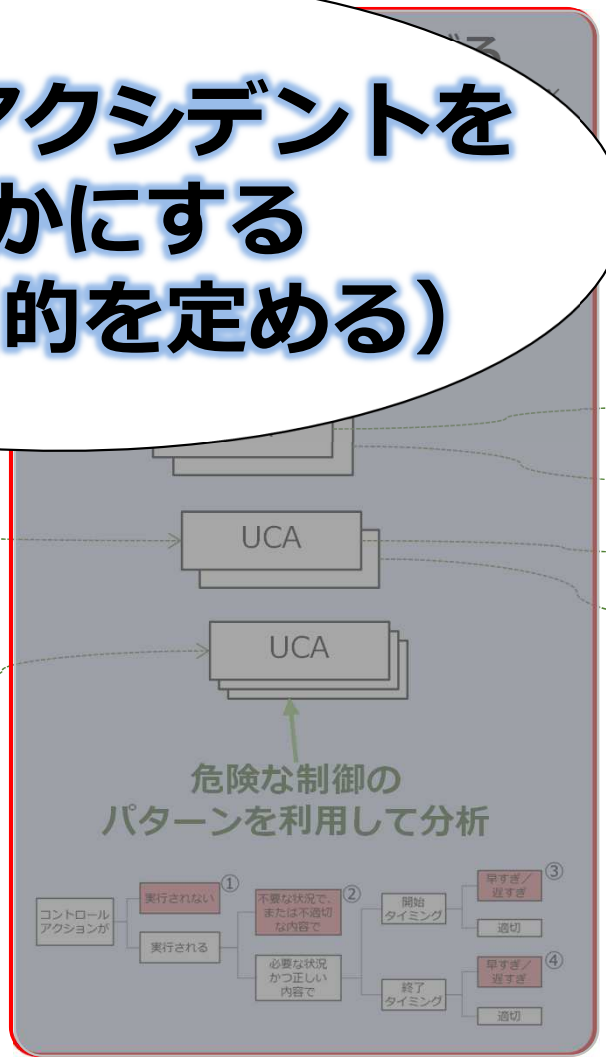
【Step1】

【Step2】

アクシデント、
ハザード、
安全制約の識別

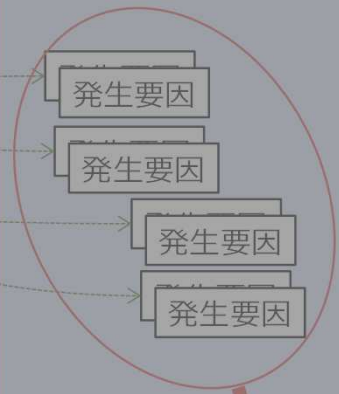
防ぎたいアクシデントを
明らかにする
（分析の目的を定める）





UCAの発生要因※
の分析

※HCF
（Hazard Causal Factor）



コンポーネント
が満たすべき
安全要件を導出

一般社団法人
組込みシステム技術協会
Japan Embedded Systems Technology Association

Step0 – 準備１：
アクシデント、ハザード、安全制約の識別

アクシデント：ステークホルダーにとって許容できない**損失**を伴う事象

※ 「**損失**」は、人命の損失や身体の損傷だけでなく、
所有物の毀損、経済的損失、ミッションの未達なども該当する

アクシデント（損失）の例

- 自動車の自動運転システムで、歩行者や乗員が**怪我**をする。
- 自動車の自動運転システムで、車両や建造物が**破損**する。
- オンライン決済システムで、利用者が**経済的損失**を被る。
- 列車の乗客が目的の駅に**たどりつけない**。
- 情報通信システムの利用者が、情報を**送受信できない**。

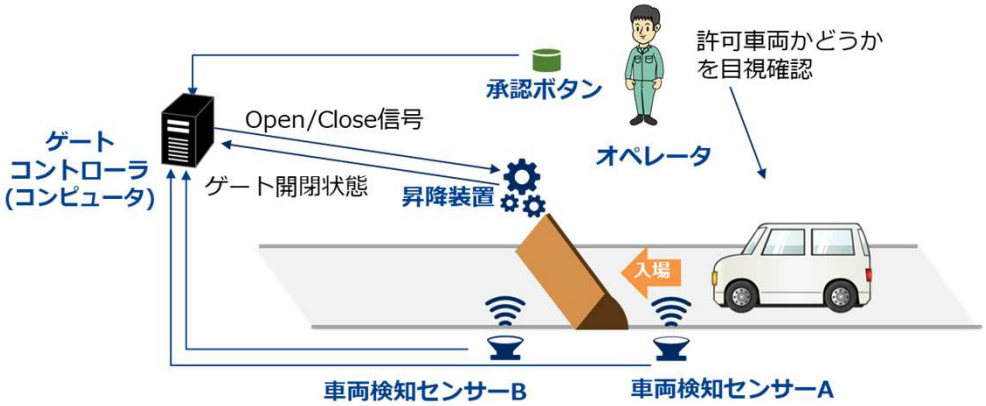
Step0 – 準備１：
アクシデント、ハザード、安全制約の識別

アクシデント：ステークホルダーにとって許容できない**損失**を伴う事象

問題

アクセス・コントロール・ゲートのSTAMP/STPA分析を行う場合、次のうちアクシデントとして適当なのはどれでしょうか？

1. **入場する車両が損傷する**
2. **歩行者が怪我をする**
3. **許可のない車両が入場してしまう**
4. **許可のある車両が入場できない**
5. **ゲートが動かない**
6. **オペレータが許可車両を見誤る**



Step0 – 準備１：
アクシデント、ハザード、安全制約の識別

アクシデント：ステークホルダーにとって許容できない**損失**を伴う事象

問題

アクセス・コントロール・ゲートのSTAMP/STPA分析を行う場合、次のうちアクシデントとして適当なのはどれでしょうか？

	1. 入場する車両が損傷する	→所有物の毀損
	2. 歩行者が怪我をする	→身体の損傷
	3. 許可のない車両が入場してしまう	→ミッション未達
	4. 許可のある車両が入場できない	→ミッション未達
✕	5. ゲートが動かない	→損失を表していない
✕	6. オペレータが許可車両を見誤る	→損失を表していない

Step0 – 準備１：
アクシデント、ハザード、安全制約の識別

アクシデント：ステークホルダーにとって許容できない**損失**を伴う事象

問題　アクセス・コントロール　場合、
次のうちアクシデ

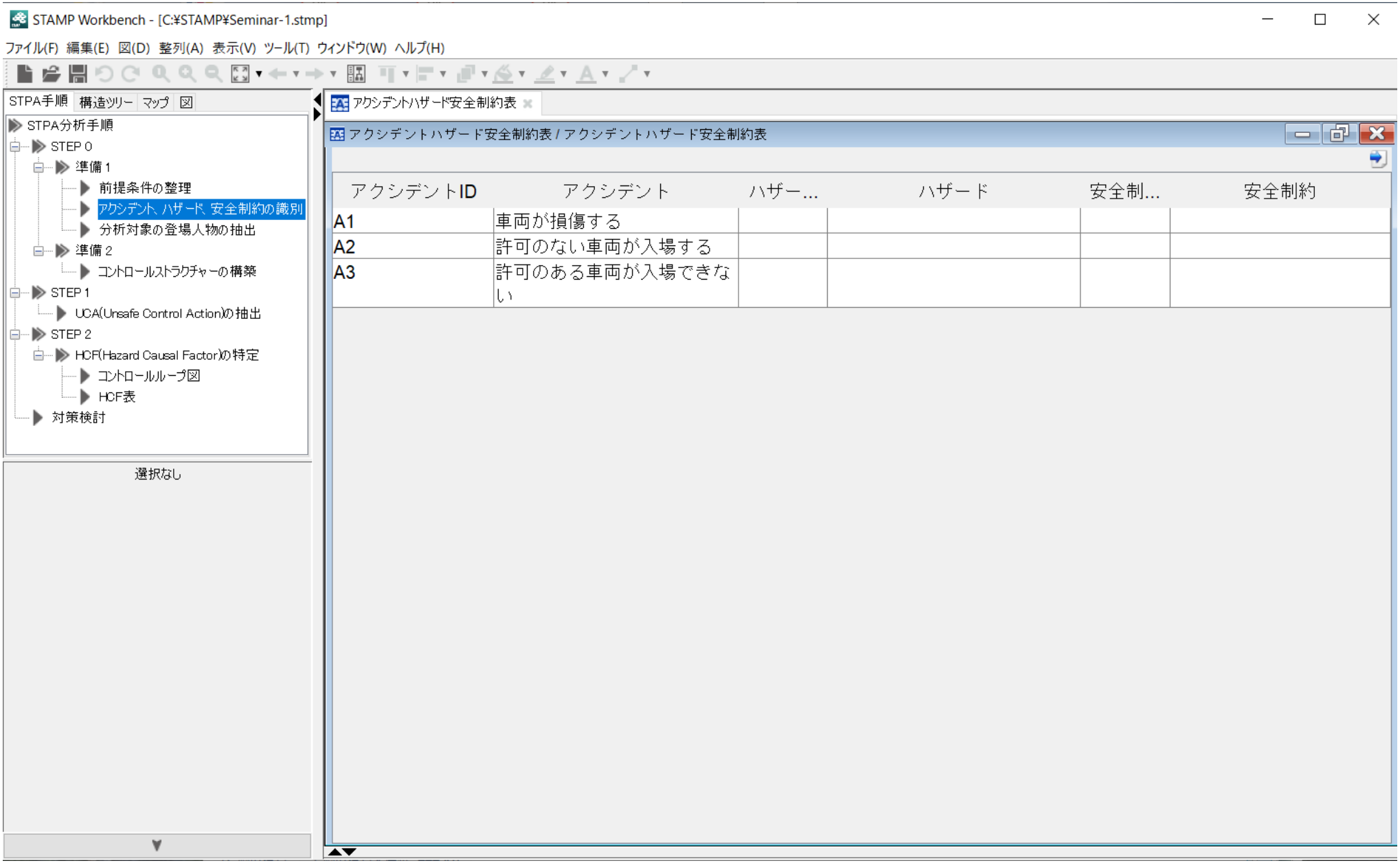
本日のセミナーでは、この3つを
分析対象の「アクシデント」とします

1. 入場する車両が損傷する　→所有物の毀損
2. 歩行者が怪我をする　→身体の損傷
3. 許可のない車両が入場してしまう　→ミッション未達
4. 許可のある車両が入場できない　→ミッション未達

❌ 5. ゲートが動かない　→損失を表していない

❌ 6. オペレータが許可車両を見誤る　→損失を表していない

STAMP Workbenchの利用イメージ



Step0 – 準備１：
アクシデント、ハザード、安全制約の識別

アクシデント：ステークホルダーにとって許容できない**損失**を伴う事象

アクシデントの例

- 自動車が、他の車両と衝突し、乗員や車両が損傷する
- 電車の乗客が線路に転落して怪我をする

ハザード：アクシデントにつながる可能性が高い**制御可能なシステムの状態**

「ハザード」の例

- 時速△△km以上で走行する自動車が、前方車両との距離が○○m未満の状態
- 電車のドアが開いたまま走行する状態

Step0 – 準備１：
アクシデント、ハザード、安全制約の識別

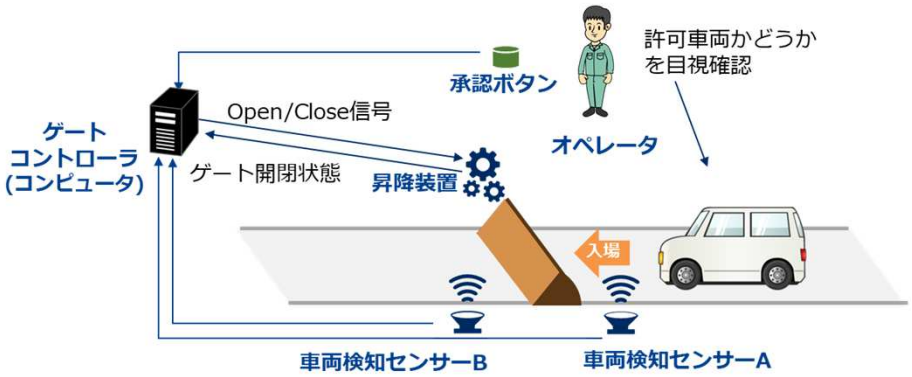
ハザード：アクシデントにつながる可能性が高い**制御可能なシステムの状態**

問題

次のうち、3つのアクシデントに対するハザードとして、適切なのはどれでしょうか？

ID	アクシデント
A1	入場する車両が損傷する
A2	許可のない車両が入場してしまう
A3	許可のある車両が入場できない

1. **車両がゲートの上にいるときにゲートが上昇する**
2. **センサーが車両を検知しない**
3. **車両が100km/hで入場してくる**
4. **許可のない車両がゲート前にいるときにゲートが開いている**
5. **許可を持つ車両がゲート前にいるときにゲートが開かない**



Step0 – 準備１：
アクシデント、ハザード、安全制約の識別

ハザード：アクシデントにつながる可能性が高い**制御可能な**
システムの状態

問題

次のうち、3つのアクシデントに対するハザードとして、適切なのはどれでしょうか？

ID	アクシデント
A1	入場する車両が損傷する
A2	許可のない車両が入場してしまう
A3	許可のある車両が入場できない

1. **車両がゲートの上にいるときにゲートが上昇する** → **A1, A3につながるハザード**

△ 2. **センサーが車両を検知しない** → **要素の状態であり、システム全体の状態ではない**

✕ 3. **車両が100km/hで入場してくる** → **直接制御できない（システム外の状態）**

4. **許可のない車両がゲート前にいるときにゲートが開いている** → **A2につながるハザード**

5. **許可を持つ車両がゲート前にいるときにゲートが開かない** → **A3につながるハザード**

Step0 – 準備１：
アクシデント、ハザード、安全制約の識別

ハザード：アクシデントにつながる可能性が高い**制御可能なシステムの状態**

問題

次のうち、3つのアクシデントに対するハザードとして、適切なのはどれでしょうか？

ID	アクシデント
A1	入場する車両が損傷する
A2	許可のない車両が入場してしまう
A3	許可のある車両が入場できない

1. **車両がゲートの上にいるときにゲートが上昇する** →A1, A3につながるハザード

△ 2. センサーが車両を検知しない

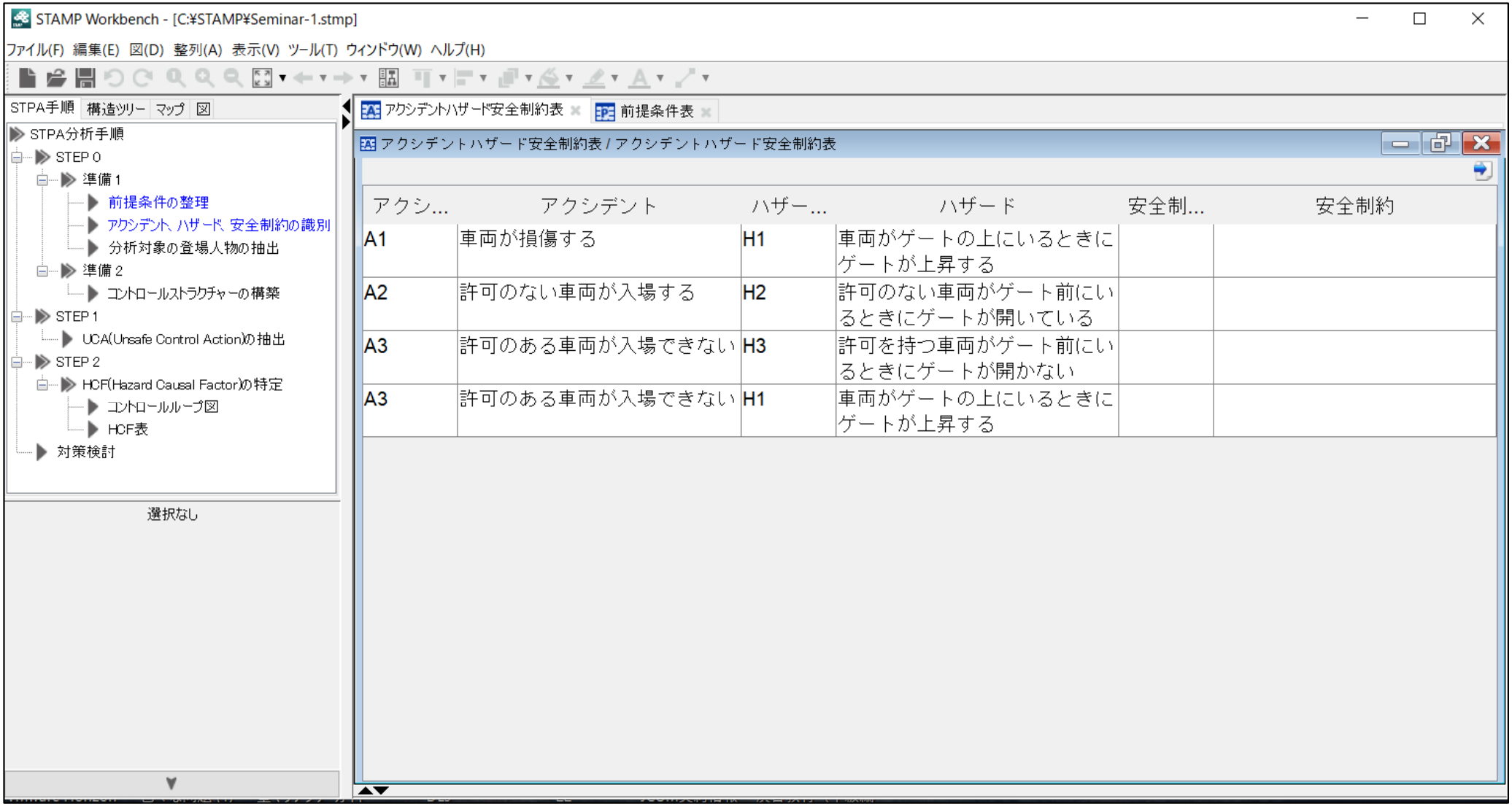
✕ 3. 車両が100km/hで入場してくる

本日のセミナーでは、この3つを分析対象の「ハザード」とします

4. **許可のない車両がゲート前にいるときにゲートが開いている** →A2につながるハザード

5. **許可を持つ車両がゲート前にいるときにゲートが開かない** →A3につながるハザード

STAMP Workbenchの利用イメージ



アクシ...	アクシデント	ハザー...	ハザード	安全制...	安全制約
A1	車両が損傷する	H1	車両がゲートの上にいるときにゲートが上昇する		
A2	許可のない車両が入場する	H2	許可のない車両がゲート前にいるときにゲートが開いている		
A3	許可のある車両が入場できない	H3	許可を持つ車両がゲート前にいるときにゲートが開かない		
A3	許可のある車両が入場できない	H1	車両がゲートの上にいるときにゲートが上昇する		

Step0 – 準備１：
アクシデント、ハザード、安全制約の識別

アクシデント：ステークホルダーにとって許容できない**損失**を伴う事象

アクシデントの例

- 自動車が、他の車両と衝突し、乗員や車両が損傷する
- 電車の乗客が線路に転落して怪我をする

ハザード：アクシデントにつながる可能性が高い**制御可能なシステムの状態**

「ハザード」の例

- 時速△△km以上で走行する自動車が、前方車両との距離が○○m未満の状態
- 電車がホーム以外にいるときにドアが開く

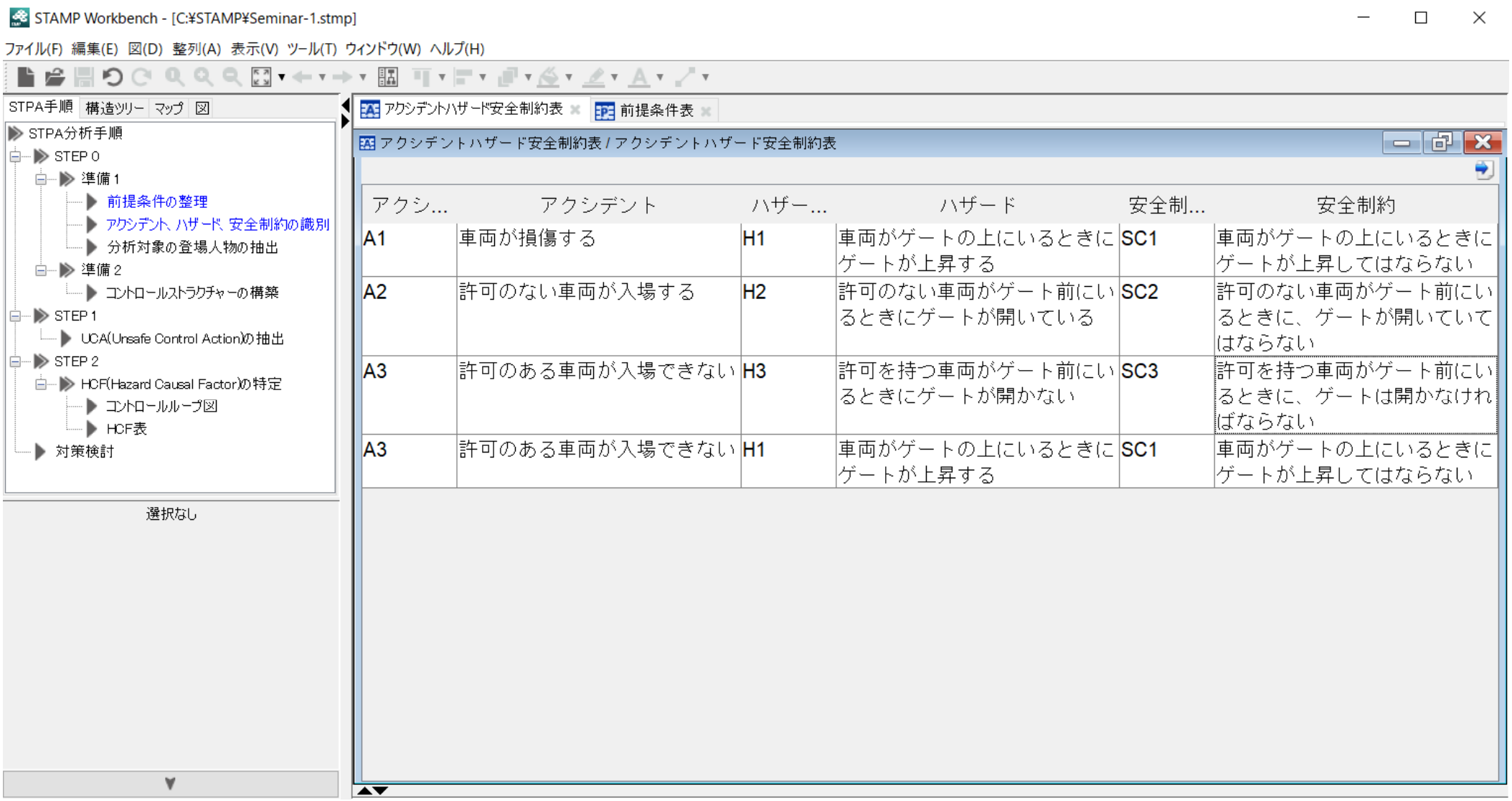
安全制約：ハザードを防ぐためにシステムが満たすべき条件

ハザードの裏返し

「安全制約」の例

- 時速△△km以上で走行する自動車は、前方車両との距離が○○m以上でなければならない
- 電車がホーム以外にいるときにドアが開いてはならない

STAMP Workbenchの利用イメージ



アクシ...	アクシデント	ハザー...	ハザード	安全制...	安全制約
A1	車両が損傷する	H1	車両がゲートの上にいるときにゲートが上昇する	SC1	車両がゲートの上にいるときにゲートが上昇してはならない
A2	許可のない車両が入場する	H2	許可のない車両がゲート前にいるときにゲートが開いている	SC2	許可のない車両がゲート前にいるときに、ゲートが開いていてはならない
A3	許可のある車両が入場できない	H3	許可を持つ車両がゲート前にいるときにゲートが開かない	SC3	許可を持つ車両がゲート前にいるときに、ゲートは開かなければならない
A3	許可のある車両が入場できない	H1	車両がゲートの上にいるときにゲートが上昇する	SC1	車両がゲートの上にいるときにゲートが上昇してはならない

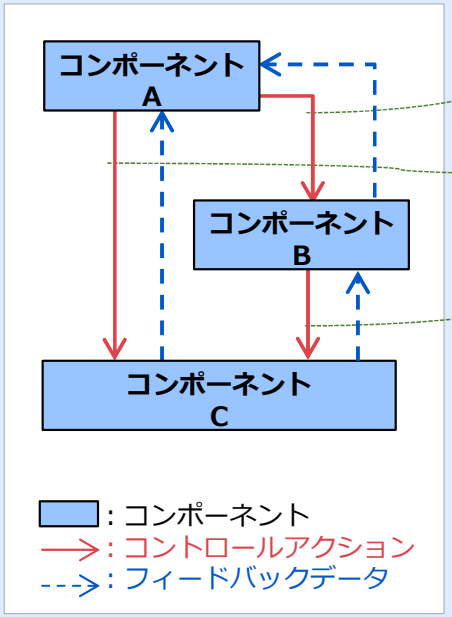
Step0 – 準備２：
コントロールストラクチャー図の作成

【Step0-準備1】 ➡ 【Step0-準備2】 ➡ 【Step1】 ➡ 【Step2】

システムレベルの
アクシデント、ハ
ザード、安全制約
の識別

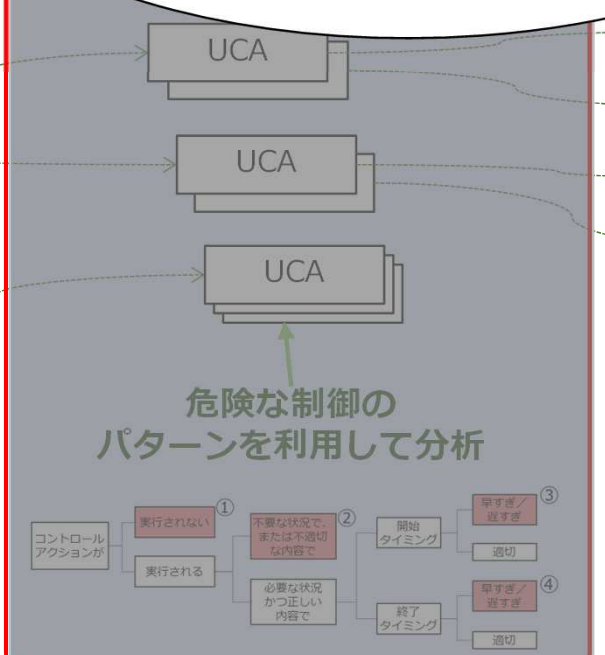
コンポーネント間の
制御関係を表すモデル※
の作成

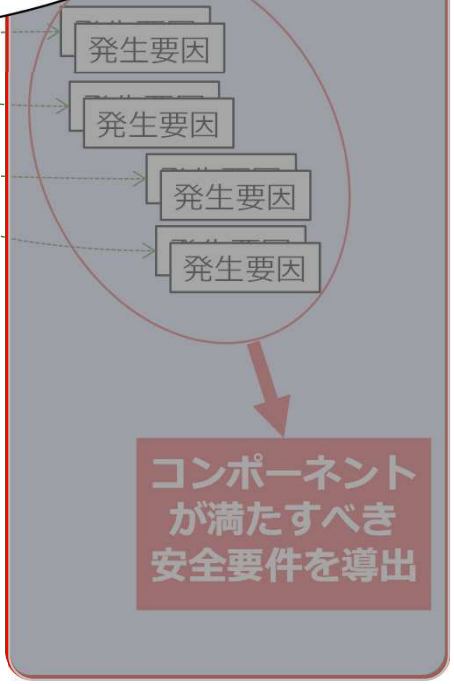
※コントロールストラクチャ



**ハザードを防ぐため
の制御の仕組みを
絵に描く**

の発生要因※
分析
sal Factor）

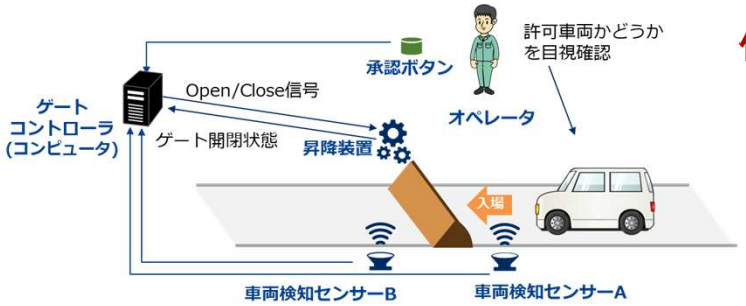




Step0 – 準備２：
コントロールストラクチャー図の作成

- コントロールストラクチャー図（制御構造図）は、STAMP/STPA分析で用いる**「モデル」**です。
- モデルとは、対象物を**関心のある観点で抽象化**したもので、正解は一つに限りません。
- コントロールストラクチャー図（制御構造図）は、**ハザードを防ぐための制御の仕組みはどうなっているか**、の観点でモデル化したものです。
- 今日のセミナーでは、コントロールストラクチャー図を作成するときの考え方の一つをご紹介します。

Aさんの設計



「ハザードを防ぐための制御の仕組みはどうなっているか」の観点で抽象化

モデル

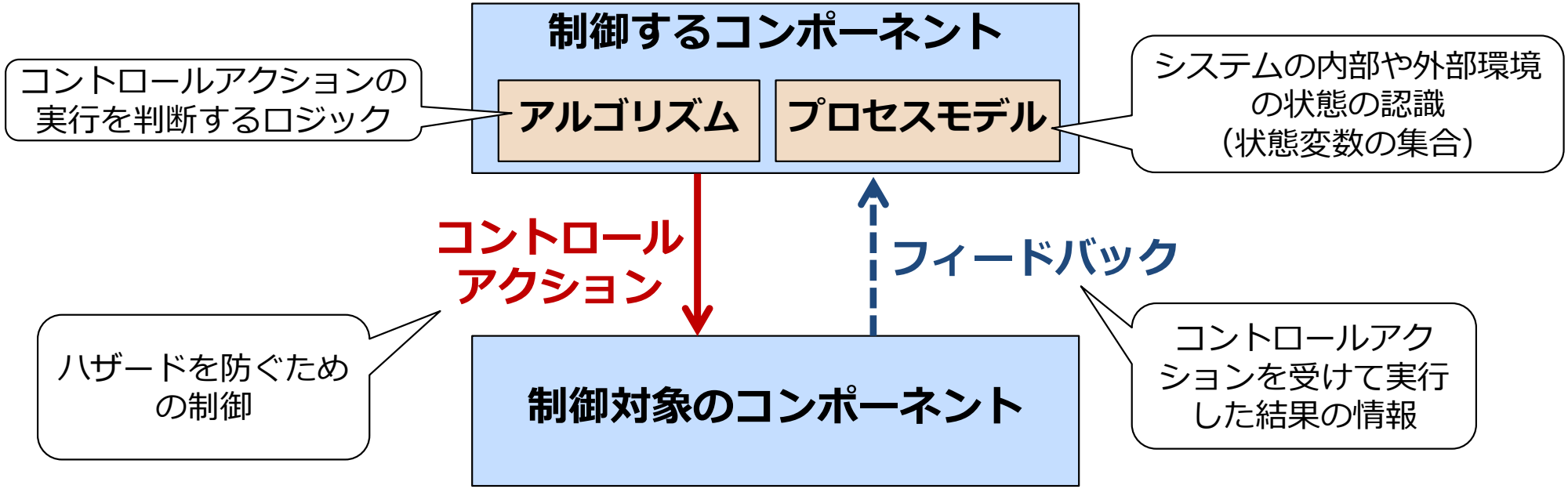
Step0 – 準備２：
コントロールストラクチャー図の作成

ハザードを防ぐ（安全制約を守る）ために

誰が、誰を、どう制御しているか

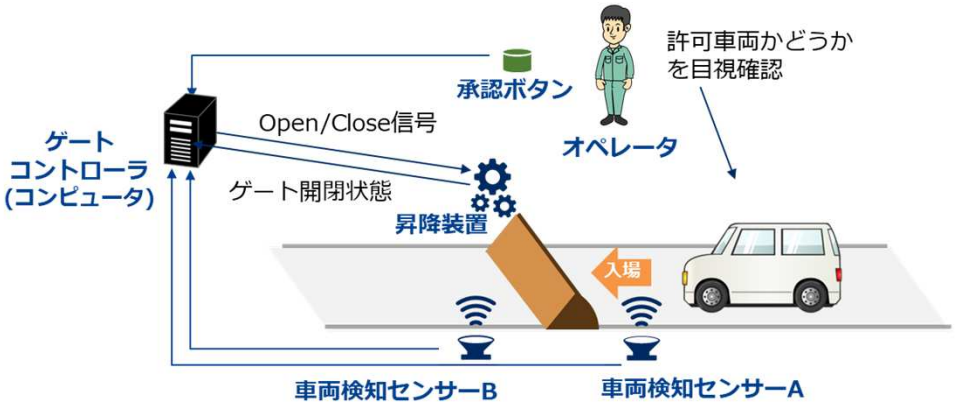
を描く

STAMPで使うモデルの基本



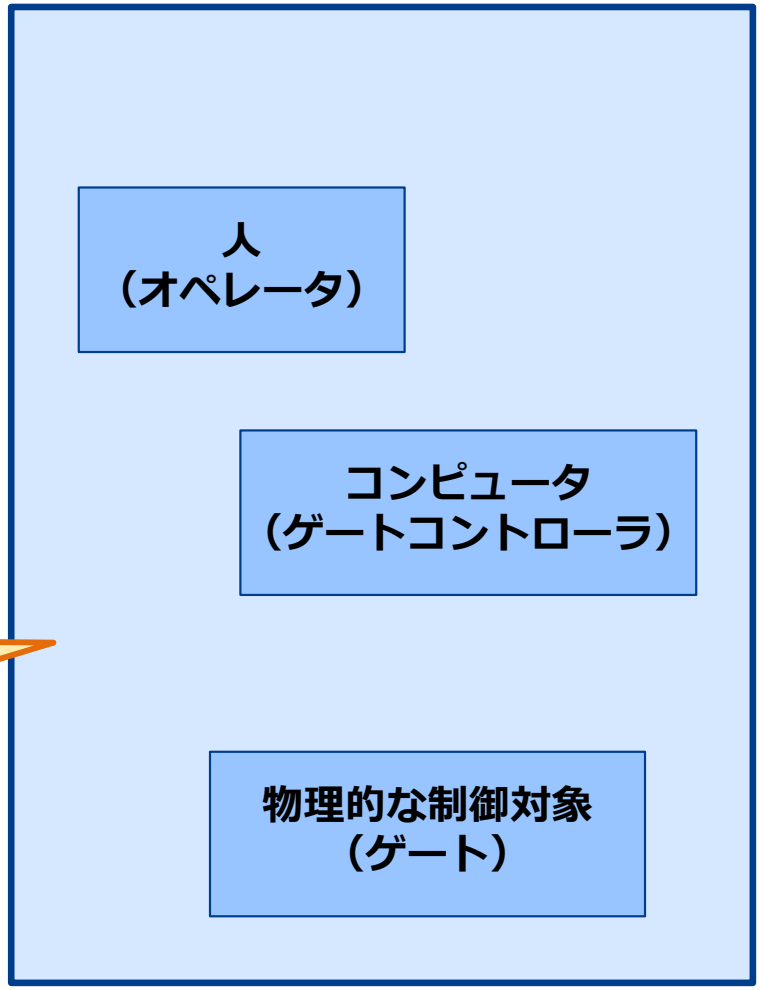
Step0 – 準備２：
コントロールストラクチャー図の作成

Aさんの設計



抽象化

ハザードを防ぐために
誰が誰をどう制御しているか？

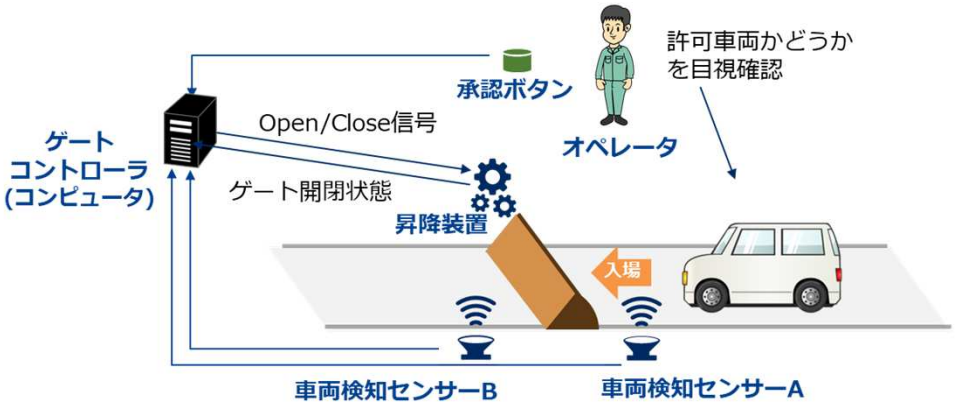


まず、基本的な要素を置いてみると、考えやすい場合がある

物理的な対象物を
人や**コンピュータ**が制御する
場合の基本的な要素

Step0 – 準備２：
コントロールストラクチャー図の作成

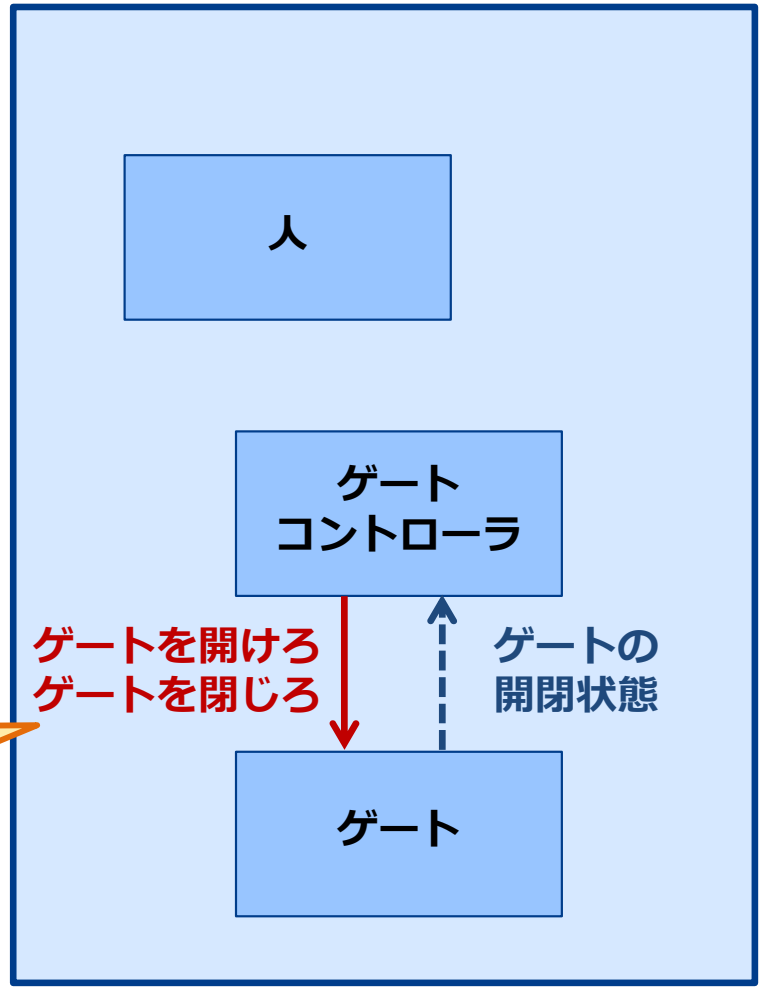
Aさんの設計



抽象化

ハザードを防ぐために
誰が誰をどう制御しているか？

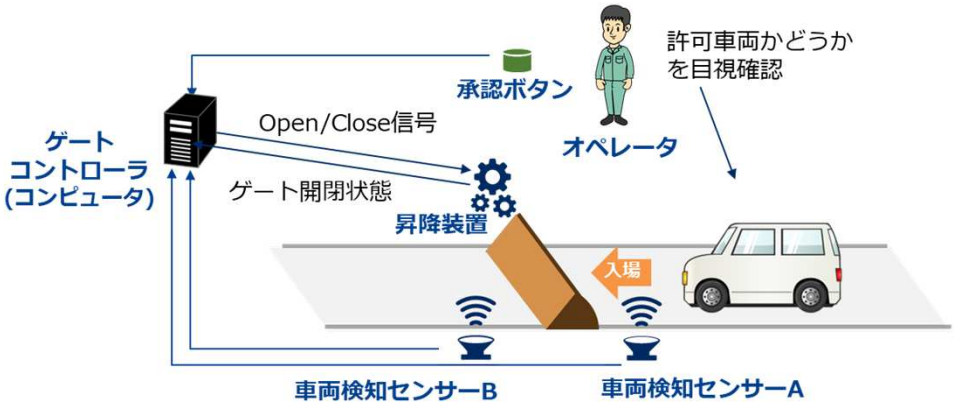
ID	ハザード
H1	車両がゲートの上にいるときにゲートが上昇する
H2	許可のない車両がゲート前にいるときにゲートが開いている
H3	許可を持つ車両がゲート前にいるときにゲートが開かない



ハザードを防ぐための
最も基本的な制御

Step0 – 準備２：
コントロールストラクチャー図の作成

Aさんの設計

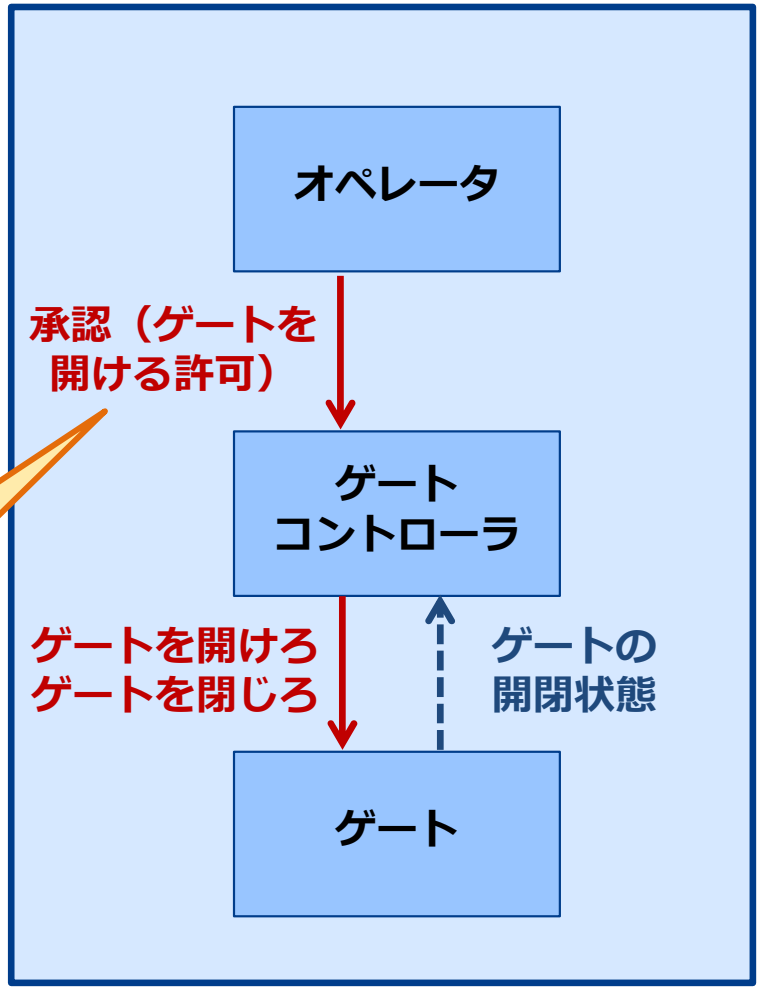


抽象化

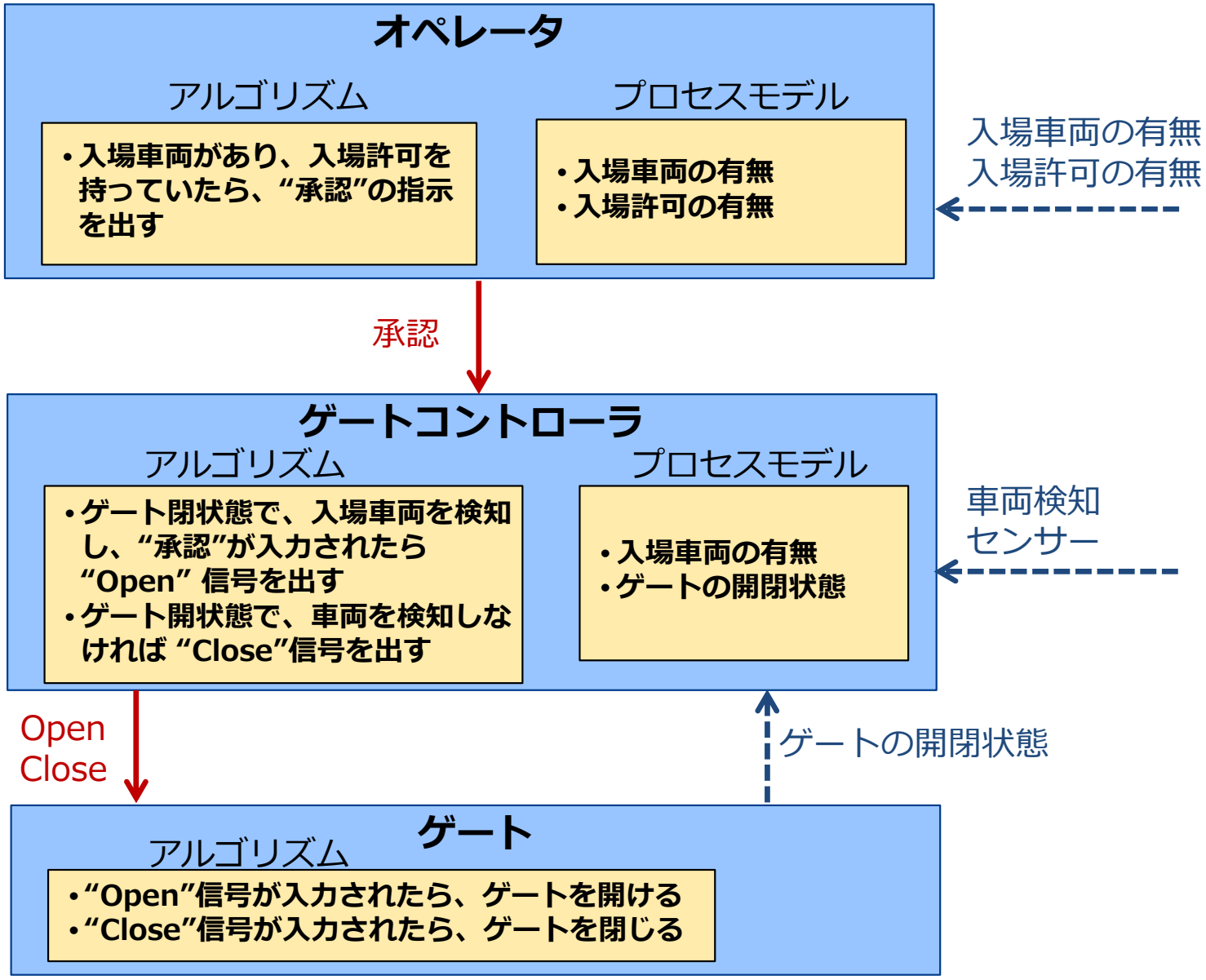
ハザードを防ぐために
誰が誰をどう制御しているか？

ID	ハザード
H1	車両がゲートの上にいるときにゲートが上昇する
H2	許可のない車両がゲート前にいるときにゲートが開いている
H3	許可を持つ車両がゲート前にいるときにゲートが開かない

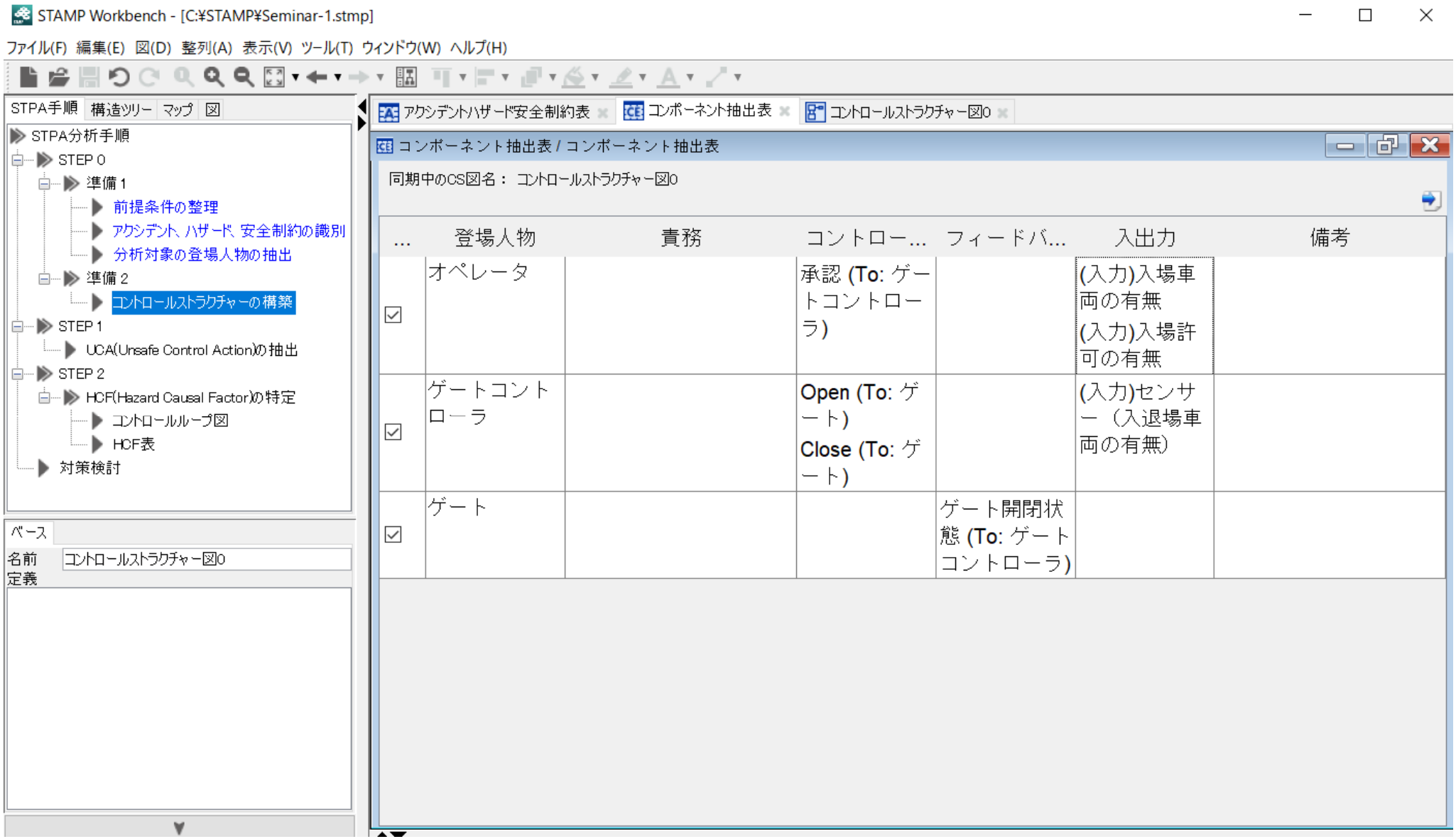
人による制御



Step0 – 準備２：
コントロールストラクチャー図の作成（完成）

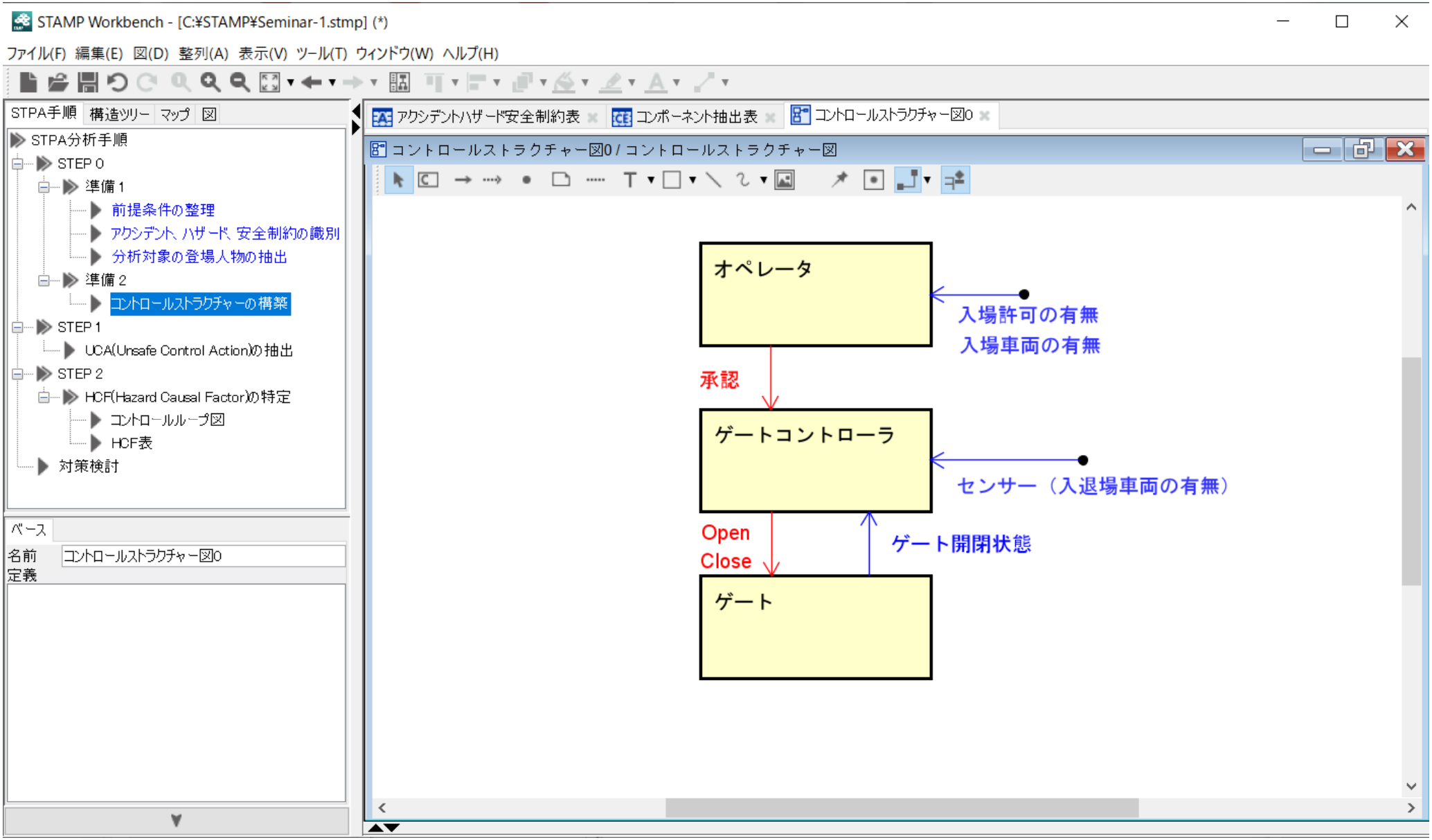


STAMP Workbenchの利用イメージ



…	登場人物	責務	コントロー…	フィードバ…	入出力	備考
☑	オペレータ		承認 (To: ゲートコントローラ)		(入力)入場車両の有無 (入力)入場許可の有無	
☑	ゲートコントローラ		Open (To: ゲート) Close (To: ゲート)		(入力)センサー（入退場車両の有無）	
☑	ゲート			ゲート開閉状態 (To: ゲートコントローラ)		

STAMP Workbenchの利用イメージ



Step1：UCAの識別

【Step0-準備1】 【Step0-準備2】 ➡ 【Step1】 ➡ 【Step2】

コントロールアクションの乱れがハザードにつながる状況を洗い出す

システム

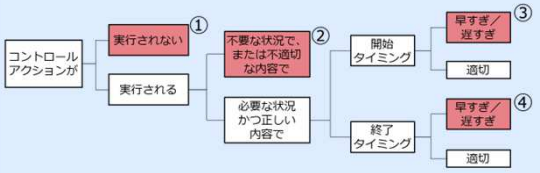
コンポーネントA
コンポーネントB
コンポーネントC
：コンポーネント
：コントロールアクション
：フィードバックデータ

ハザードにつながるコントロールアクション※の識別

※UCA
（Unsafe Control Action）

UCA
UCA
UCA

危険な制御のパターンを利用して分析



UCAの発生要因※の分析

※HCF
（Hazard Causal Factor）

発生要因
発生要因
発生要因
発生要因

コンポーネントが満たすべき安全要件を導出

Step1：UCAの識別

UCA（Unsafe Control Action）とは
「ある特定の状況でハザードにつながるコントロールアクション」

例：自動車というシステムの「加速」というコントロールアクションを考える。
「加速」自体は、安全とも危険とも判断できない。

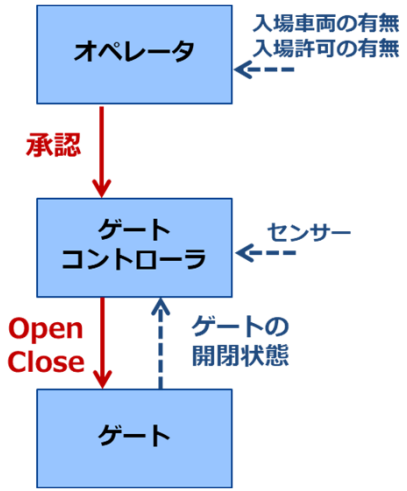
しかし、

- 「路面が滑りやすく、前方車両と近づきつつある」という状況では、「加速」の実行は危険な状態につながる。
- 高速道路に合流時、周りが時速80Kmで流れていて自車が時速40Kmの状況では、「加速」が実行されないと危険な状態につながる。

このような
ハザードにつながる「コントロールアクションと状況の組み合わせ」を識別する

Step1：UCAの識別

コントロールアクション



「承認」

「Open」

「Close」

×

4つのガイドワード

（コントロールアクションの振る舞いが
ハザードにつながるパターン）

実行されずハザード			
実行される	不要/不正なものが実行されてハザード		
	必要・正しいもの	早すぎ、遅すぎ、誤順序で実行されてハザード	
		正しいタイミング	実行が長過ぎ、短過ぎてハザード

組み合わせの一つ一つについて、
ハザードにつながる「状況」を考える

（UCAが起きる要因や現実性は、まだ考えない）

Step1：UCAの識別

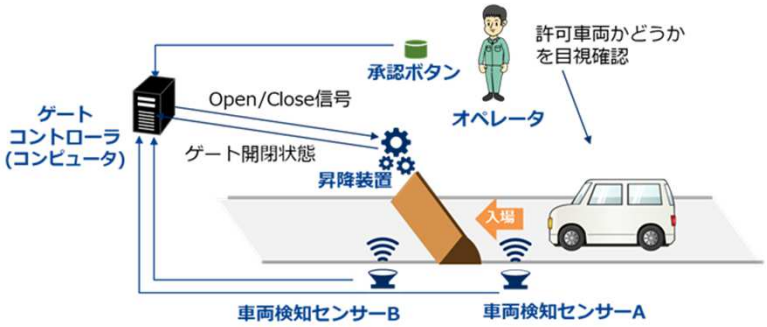
例

「Close」について、ガイドワードを当てはめ、ハザードにつながる状況を考える。

ガイドワード

コントロールアクション	実行されない	不要/不正なものが実行される	早すぎ/遅すぎ/誤順序で実行	長すぎ/短すぎる実行
Close	「Close」が実行されないため、ハザードになる	どんな状況で？	?	?

ID	ハザード
H1	車両がゲートの上にいるときにゲートが上昇する
H2	許可のない車両がゲート前にいるときにゲートが開いている
H3	許可を持つ車両がゲート前にいるときにゲートが開かない



Step1：UCAの識別

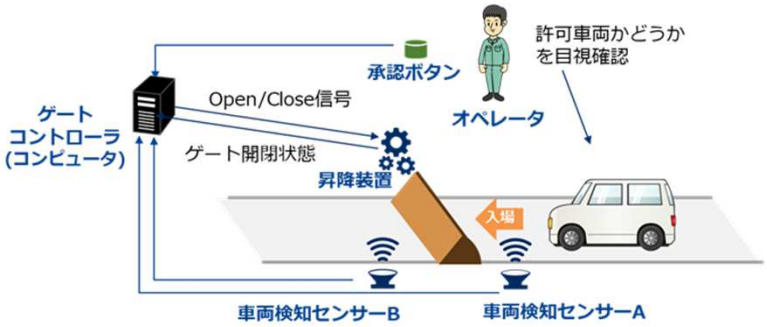
例

「Close」について、ガイドワードを当てはめ、ハザードにつながる状況を考える。

ガイドワード

コントロールアクション	実行されない	不要/不正なものが実行される	早すぎ/遅すぎ/誤順序で実行	長すぎ/短すぎる実行
Close	**ゲートが開いていて許可のない車両がゲート前にいる時、「Close」が実行されず、ハザードになる[H-2]**	?	?	?

ID	ハザード
H1	車両がゲートの上にいるときにゲートが上昇する
H2	許可のない車両がゲート前にいるときにゲートが開いている
H3	許可を持つ車両がゲート前にいるときにゲートが開かない



Step1：UCAの識別

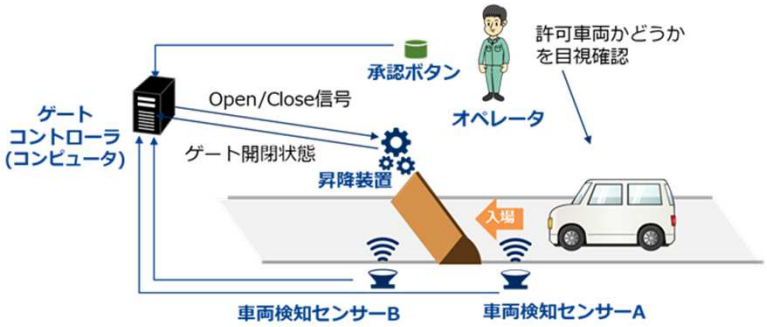
例

「Close」について、ガイドワードを当てはめ、ハザードにつながる状況を考える。

ガイドワード

コントロールアクション	実行されない	不要/不正なものが実行される	早すぎ/遅すぎ/誤順序で実行	長すぎ/短すぎる実行
Close	**ゲートが開いていて許可のない車両がゲート前にいる時、「Close」が実行されず、ハザードになる[H-2]**	**「Close」が実行されてハザードになる**	**どんな状況で？**	？

ID	ハザード
H1	車両がゲートの上にいるときにゲートが上昇する
H2	許可のない車両がゲート前にいるときにゲートが開いている
H3	許可を持つ車両がゲート前にいるときにゲートが開かない



Step1：UCAの識別

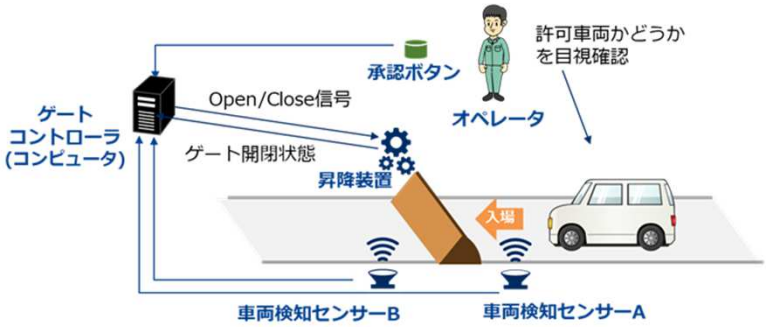
例

「Close」について、ガイドワードを当てはめ、ハザードにつながる状況を考える。

ガイドワード

コントロールアクション	実行されない	不要/不正なものが実行される	早すぎ/遅すぎ/誤順序で実行	長すぎ/短すぎる実行
Close	**ゲートが開いていて許可のない車両がゲート前にいる時、「Close」が実行されず、ハザードになる[H-2]**	**車両がゲートの上にいる時、「Close」が実行されてハザードになる[H-1]**	?	?

ID	ハザード
H1	車両がゲートの上にいるときにゲートが上昇する
H2	許可のない車両がゲート前にいるときにゲートが開いている
H3	許可を持つ車両がゲート前にいるときにゲートが開かない



Step1：UCAの識別

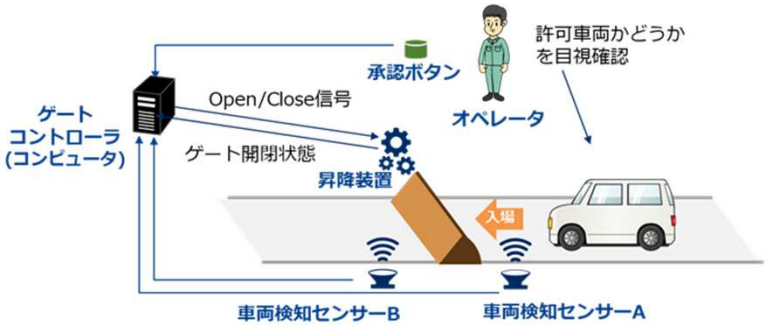
例

「Close」について、ガイドワードを当てはめ、ハザードにつながる状況を考える。

ガイドワード

コントロールアクション	実行されない	不要/不正なものが実行される	早すぎ/遅すぎ/誤順序で実行	長すぎ/短すぎる実行
Close	**ゲートが開いていて許可のない車両がゲート前にいる時、「Close」が実行されず、ハザードになる[H-2]**	**車両がゲートの上にいる時、「Close」が実行されてハザードになる[H-1]**	**車両がゲートを通過し終わる前にCloseが実行され、ハザードになる[H-1]**	?

ID	ハザード
H1	車両がゲートの上にいるときにゲートが上昇する
H2	許可のない車両がゲート前にいるときにゲートが開いている
H3	許可を持つ車両がゲート前にいるときにゲートが開かない



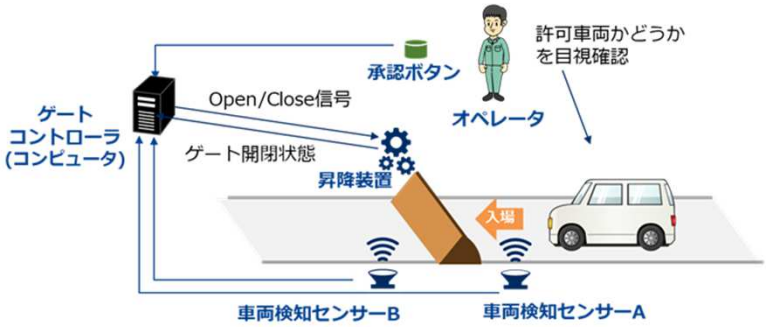
Step1：UCAの識別

問題 **「Open」について、「実行される」のガイドワードを当てはめたとき、識別されるUCAとして適当なのはどれでしょうか？**

コントロールアクション	実行されない	不要/不正なものが実行される	早すぎ/遅すぎ/誤順序で実行	長すぎ/短すぎる実行
Open	?	**実行されて、ハザードになる**	?	?

1. **許可のない車両がゲート前にいるときに Open が実行される**
2. **車両がいないときに Open が実行される**
3. **子供がゲート近くにいるときに Open が実行される**

ID	ハザード
H1	車両がゲートの上にいるときにゲートが上昇する
H2	許可のない車両がゲート前にいるときにゲートが開いている
H3	許可を持つ車両がゲート前にいるときにゲートが開かない



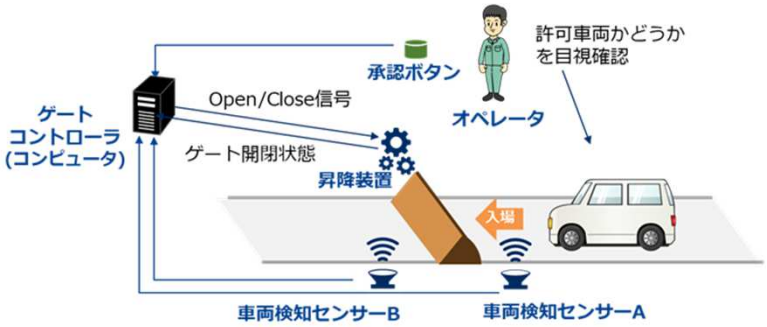
Step1：UCAの識別

問題 「Open」について、「実行される」のガイドワードを当てはめたとき、識別されるUCAとして適当なのはどれでしょうか？

コントロールアクション	実行されない	不要/不正なものが実行される	早すぎ/遅すぎ/誤順序で実行	長すぎ/短すぎる実行
Open	?	実行されて、ハザードになる	?	?

1. 許可のない車両がゲート前にいるときに Open が実行される　←H2につながる
2. ✕ 車両がいないときに Open が実行される　←3つのハザードにはつながらない
3. ✕ 子供がゲート近くにいるときに Open が実行される　←3つのハザードにはつながらない

ID	ハザード
H1	車両がゲートの上にいるときにゲートが上昇する
H2	許可のない車両がゲート前にいるときにゲートが開いている
H3	許可を持つ車両がゲート前にいるときにゲートが開かない



Step1：UCAの識別

コントロールアクション	実行されない	不要/不正なものが実行される	早すぎ/遅すぎ/誤順序で実行	長すぎ/短すぎる実行
承認	**UCA1**: 許可のある車両がゲート前にいる時に、「承認」が実行されない。[H-3]	**UCA2:** 許可のない車両がゲート前にいる時に、「承認」が実行される。[H-2]	**UCA3**: 車両の入場許可を確認するより前に「承認」が実行される。[H-2]	---
Open	**UCA4**: 許可のある車両がゲート前にいる時に[Open」が実行されない。[H-3]	**UCA5**: 許可のない車両がゲート前にいる時に、「Open」が実行される。[H-2]	---	**UCA6**: 許可のある車両がゲート前にいる時に「Open」の実行が短すぎる。[H-3]
Close	**UCA7**: ゲート開放時、許可のない車両がゲート前にいるときに「Close」が実行されない。[H-2]	**UCA8**: 車両がゲートの上にいる時に、「Close」が実行される。[H-1]	**UCA9**: 許可車両がゲートを通過後、「Close」が遅すぎ、ゲートが開いた状態で後続の無許可車両が来る。[H-2] **UCA10**: 車両がゲートを通過し終わるより前に「Close」が実行される。[H-1]	**UCA11**: 許可車両がゲートを通過後、「Close」の実行が短すぎ、ゲートがほぼ開いた状態で、後続の無許可車両が来る。[H-2]

Step1：UCAの識別

コントロールアクション	実行されない	不要/不正なものが実行される	早すぎ/遅すぎ/誤順序で実行	長すぎ/短すぎる実行
承認	**UCA1**: 許可のある車両がゲート前にいる時に、「承認」が実行されない。[H-3]	**UCA2:** 許可のない車両がゲート前にいる時に、「承認」が実行される。	**UCA3**: 車両の入場許可を確認するより前に「承認」が実行される。[H-2]	---
Open	**UCA** ゲー [Open [H-3]		---	**UCA6**: 許可のある車両がゲート前にいる時に「Open」の実行が短すぎる。[H-3]
Close	**UCA7**: ゲート開放時、許可のない車両がゲート前にいるときに「Close」が実行されない。[H-2]	**UCA8**: 車両がゲートの上にいる時に、「Close」が実行される。[H-1]	**UCA9**: 許可車両がゲートを通過後、「Close」が遅すぎ、ゲートが開いた状態で後続の無許可車両が来る。[H-2] **UCA10**: 車両がゲートを通過し終わるより前に「Close」が実行される。[H-1]	**UCA11**: 許可車両がゲートを通過後、「Close」の実行が短すぎ、ゲートがほぼ開いた状態で、後続の無許可車両が来る。[H-2]

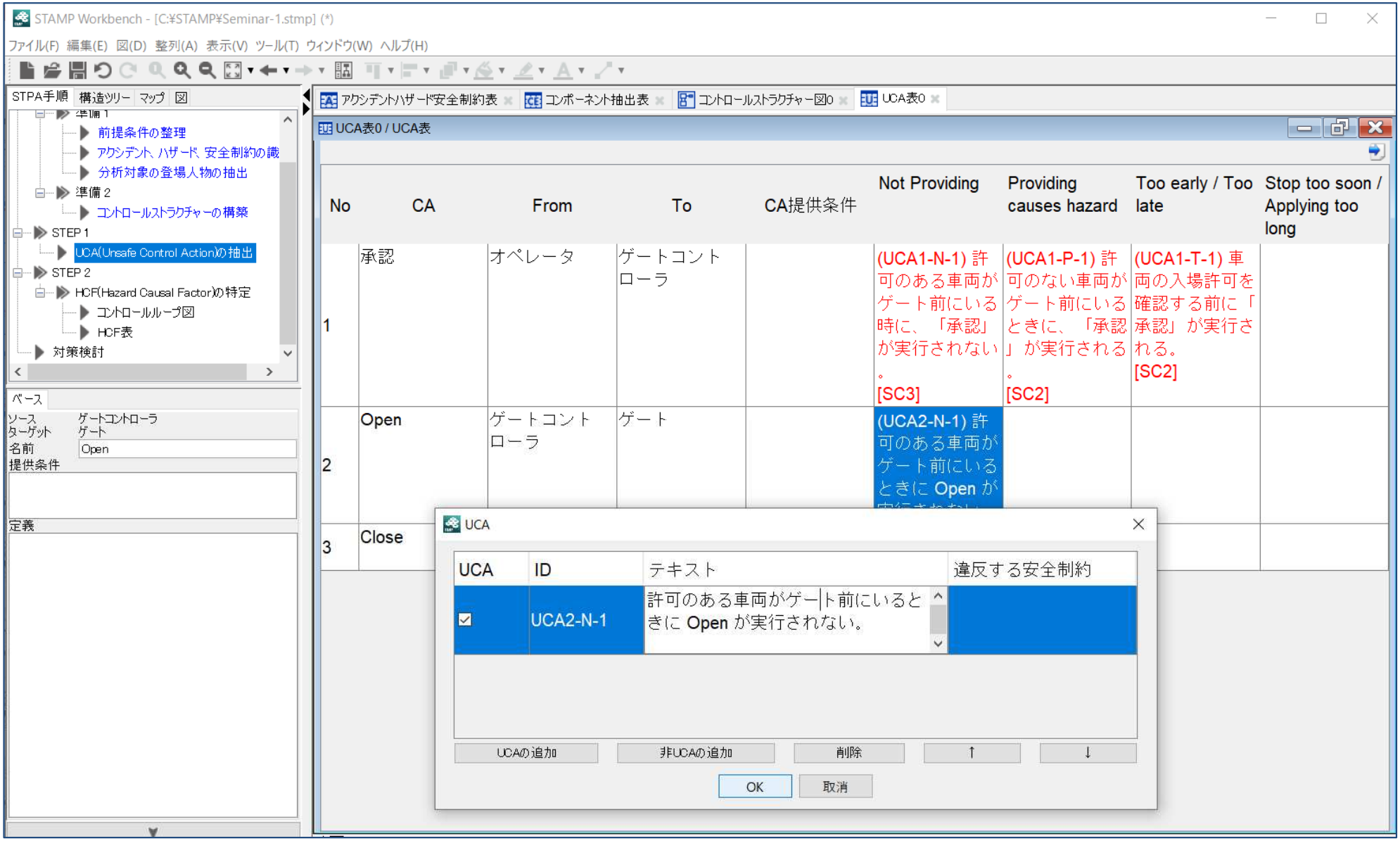
UCA9: 許可車両がゲートを通過後、「Close」が遅すぎ、ゲートが開いた状態で後続の無許可車両が来る。[H-2]

Step1：UCAの識別

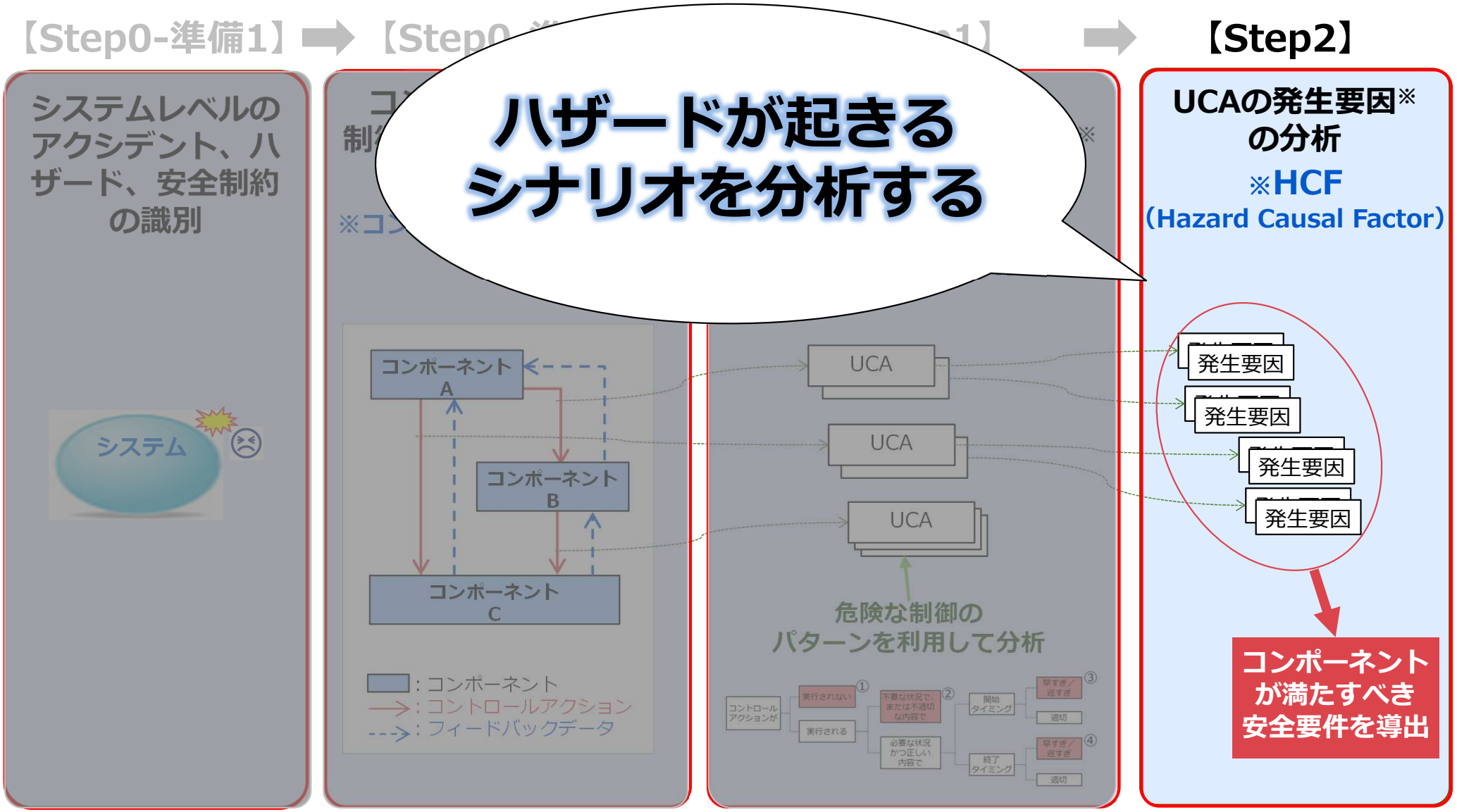
コントロールアクション	実行されない	不要/不正なものが実行される	早すぎ/遅すぎ/誤順序で実行	長すぎ/短すぎる実行
承認	**UCA1**: 許可のある車両がゲート前にいる時に、「承認」が実行されない。[H-3]	**UCA2:** 許可のない車両がゲート前にいる時に、「承認」が実行される。[H-2]	**UCA3**: 車両の入場許可を確認するより前に「承認」が実行される。	---
Open	**UCA4**: 許可のある車両がゲート前にいる時に[Open」が実行されない。[H-3]	**UCA** ゲー 「Ope [H-2]	**5**: 許可のある車両が ト前にいる時に pen」の実行が短すぎ る。[H-3]	
Close	**UCA7**: ゲート開放時、許可のない車両がゲート前にいるときに「Close」が実行されない。[H-2]	**UCA8**: 車両がゲートの上にいる時に、「Close」が実行される。[H-1]	**UCA9**: 許可車両がゲートを通過後、「Close」が遅すぎ、ゲートが開いた状態で後続の無許可車両が来る。[H-2] **UCA10**: 車両がゲートを通過し終わるより前に「Close」が実行される。[H-1]	**UCA11**: 許可車両がゲートを通過後、「Close」の実行が短すぎ、ゲートがほぼ開いた状態で、後続の無許可車両が来る。[H-2]

UCA11: 許可車両がゲートを通過後、「Close」の実行が短すぎ、ゲートがほぼ開いた状態で、後続の無許可車両が来る。[H-2]

STAMP Workbenchの利用イメージ



Step2：UCAの発生要因（HCF）の分析



Step2：UCAの発生要因（HCF）の分析

Step1までで、システムのハザードを起こす要因が
コントロールアクションの視点でブレークダウンされ、
UCAとして識別された

**Step2では、UCAの一つ一つについて
それを発生させるシナリオを
具体的に考える**

考えるきっかけとして
UCAに関わるコンポーネントの周辺でチェックすべき
ポイントが「ヒントワード」として示される。

ヒントワード

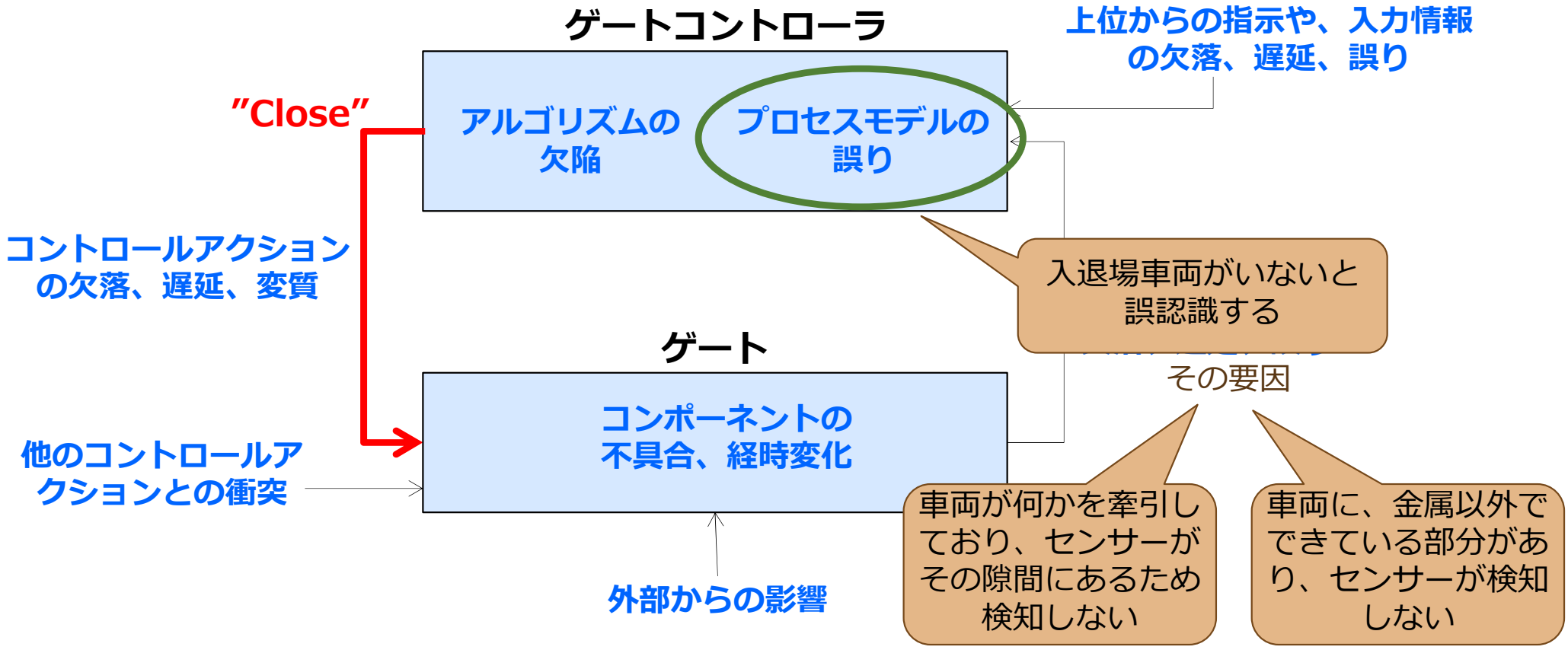
コントロールアクションを「出すコンポーネント」と「受けるコンポーネント」にフォーカスし、UCAの要因となり得る因子を示したもの

注意：ヒントワードについて、STAMPとしての規定はない。
対象システムに合うヒントワードを設定することが推奨される。
（下図、緑字で示すのは、ごく一般的なヒントワードの例）

青字：ヒントワード

Step2：UCAの発生要因（HCF）の分析

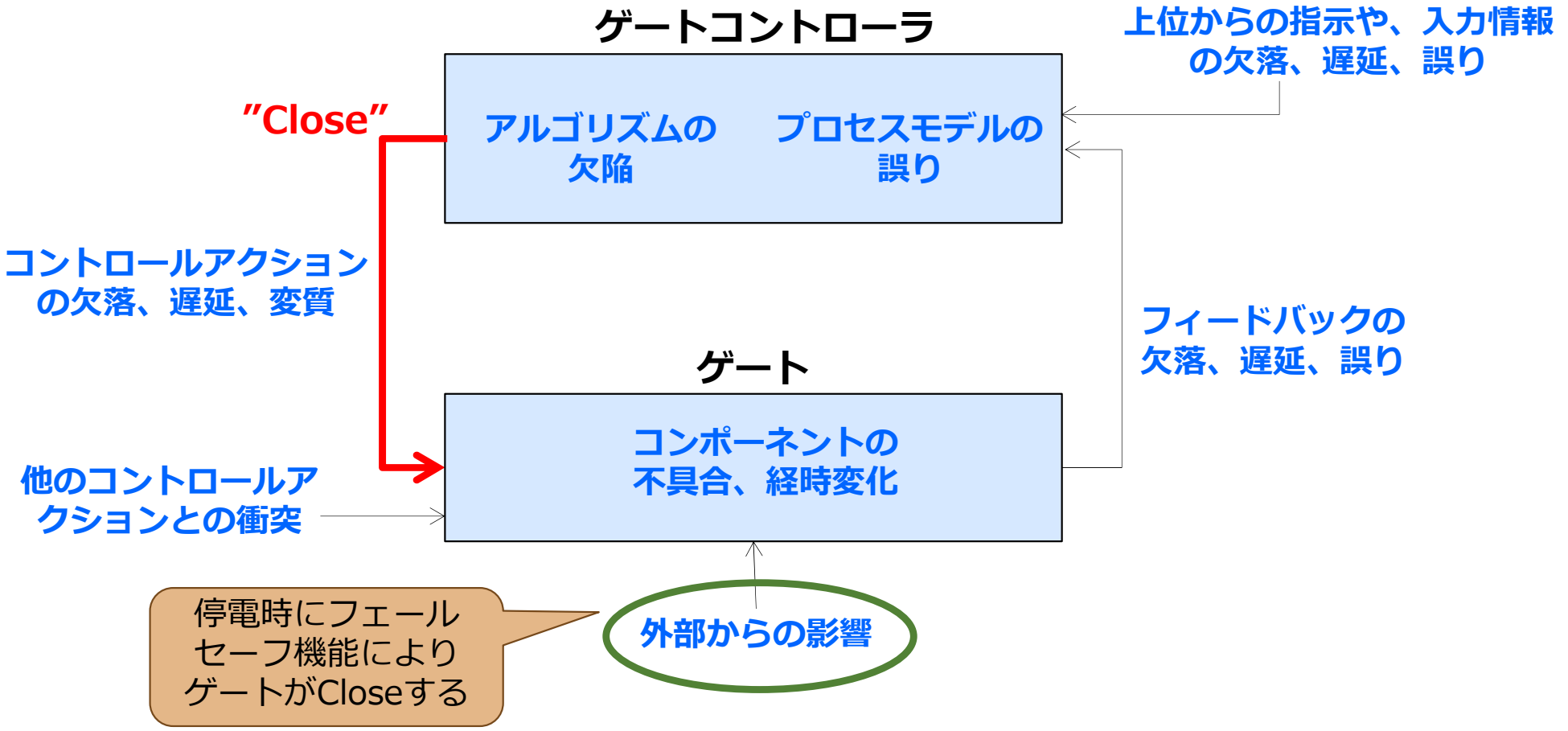
例 **UCA8**: 車両がゲートの上にいる時に、「Close」が実行される。[H-1]



Step2：UCAの発生要因（HCF）の分析

例

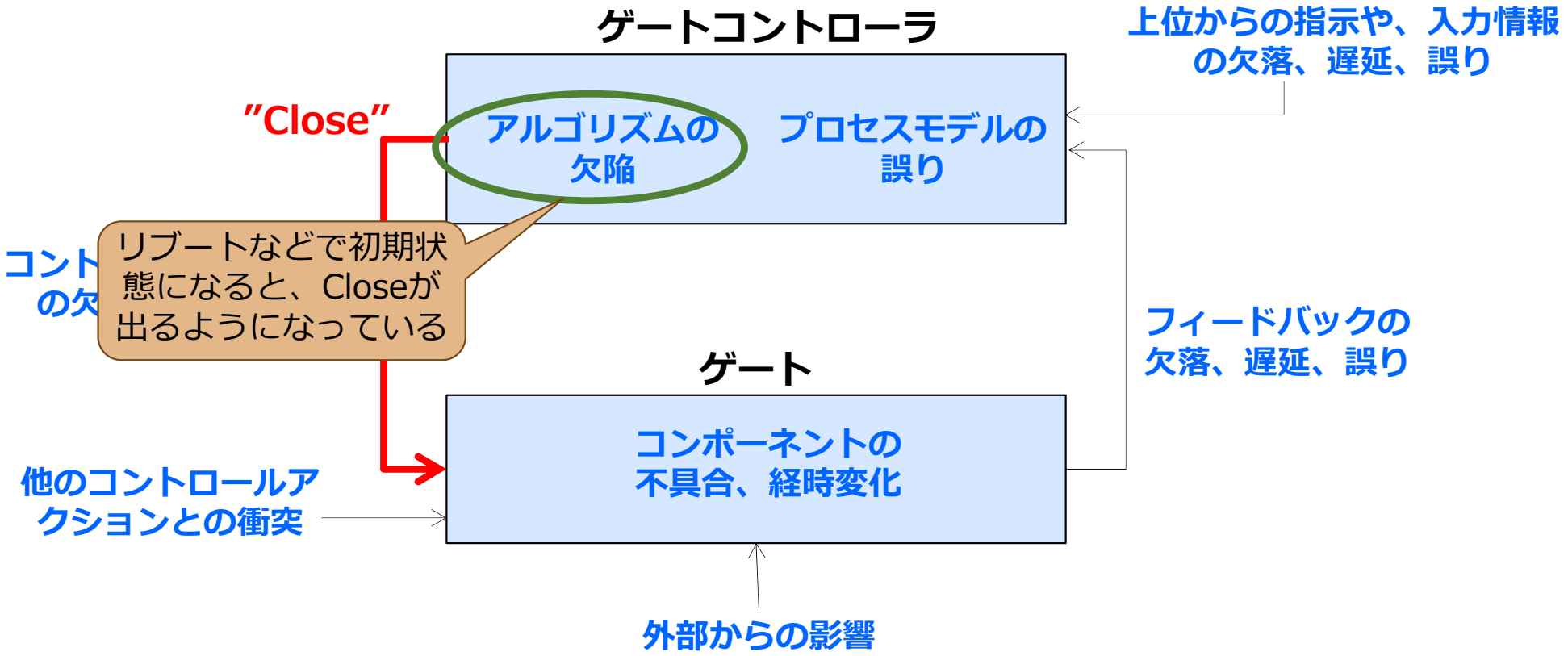
UCA8: 車両がゲートの上にいる時に、「Close」が実行される。[H-1]



Step2：UCAの発生要因（HCF）の分析

例

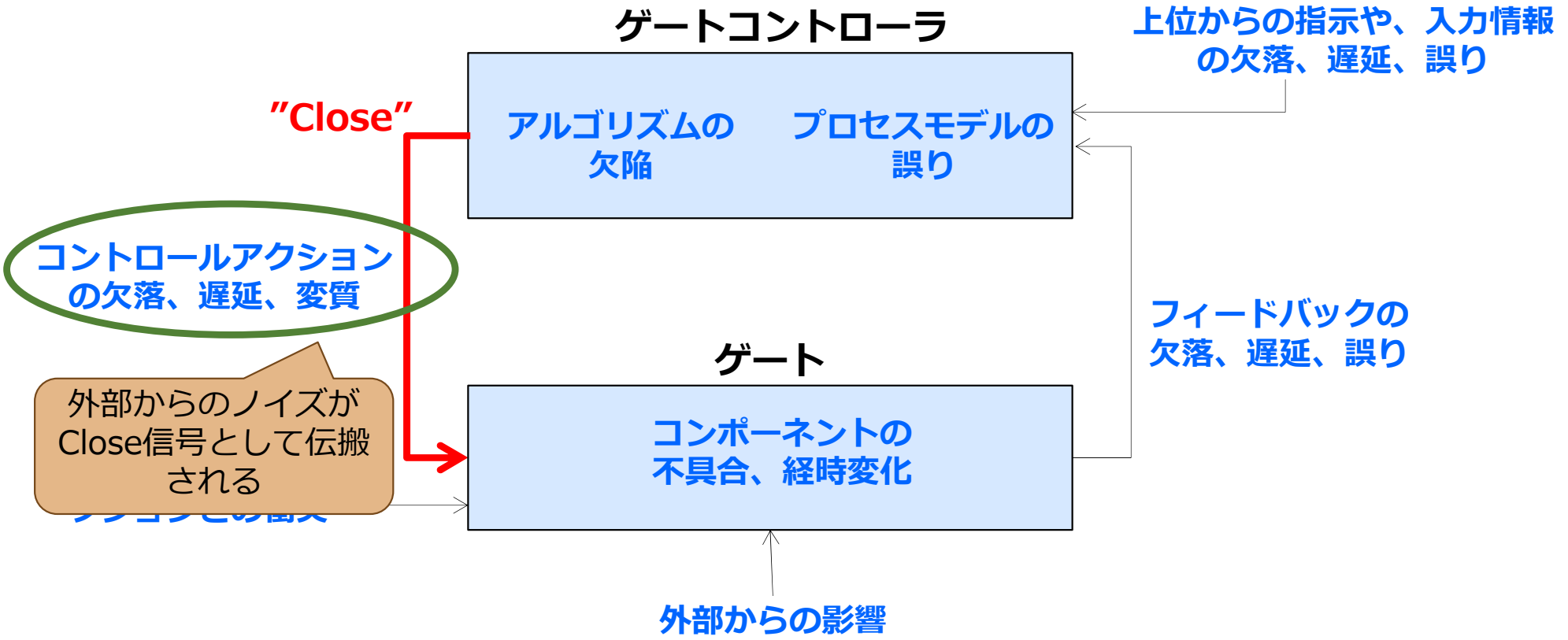
UCA8: 車両がゲートの上にいる時に、「Close」が実行される。[H-1]



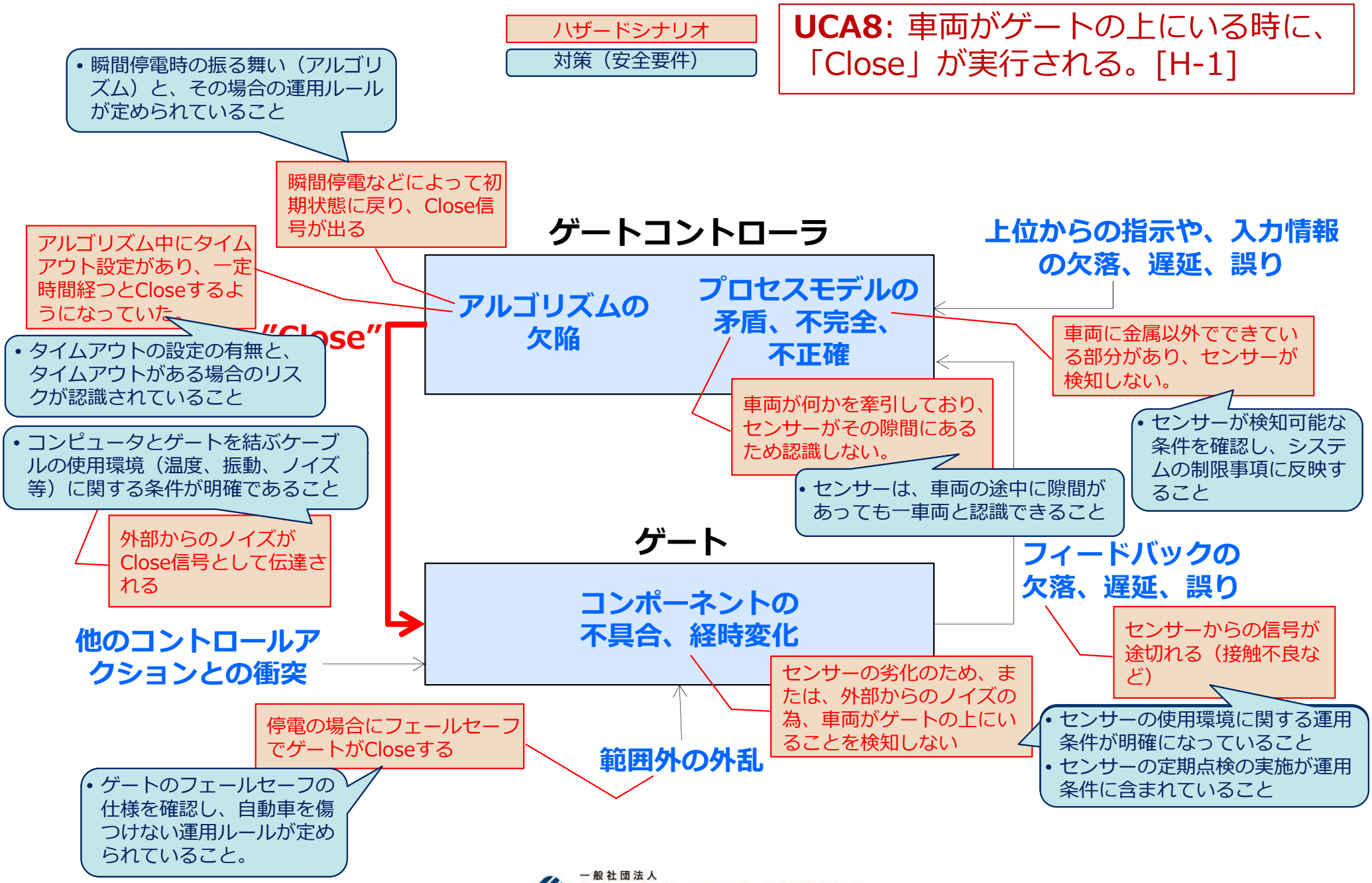
Step2：UCAの発生要因（HCF）の分析

例

UCA8: 車両がゲートの上にいる時に、「Close」が実行される。[H-1]



Step2：UCAの発生要因（HCF）の分析



今回の
STAMP/STPA分析事例の
おさらい

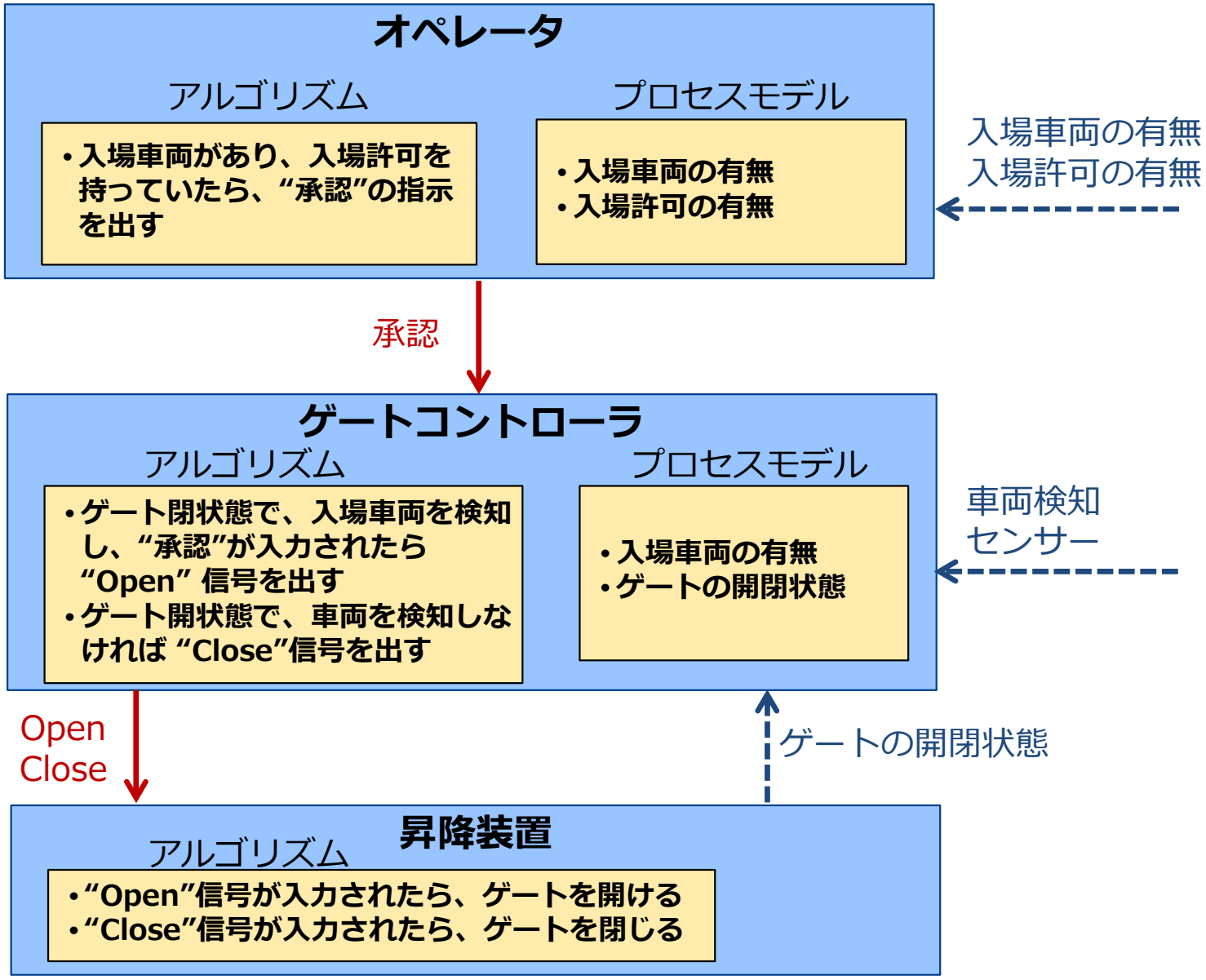
Step0 – 準備１：
アクシデント、ハザード、安全制約の識別

ID	アクシデント
A1	入場する車両が損傷する
A2	許可のない車両が入場してしまう
A3	許可のある車両が入場できない

ID	ハザード
H1	車両がゲートの上にいるときにゲートが上昇する
H2	許可のない車両がゲート前にいるときにゲートが開いている
H3	許可を持つ車両がゲート前にいるときにゲートが開かない

ID	安全制約
SC1	車両がゲートの上にいるときにゲートが上昇してはならない
SC2	許可のない車両がゲート前にいるときにゲートが開いていてはならない
SC3	許可を持つ車両がゲート前にいるときにゲートが開かなければならない

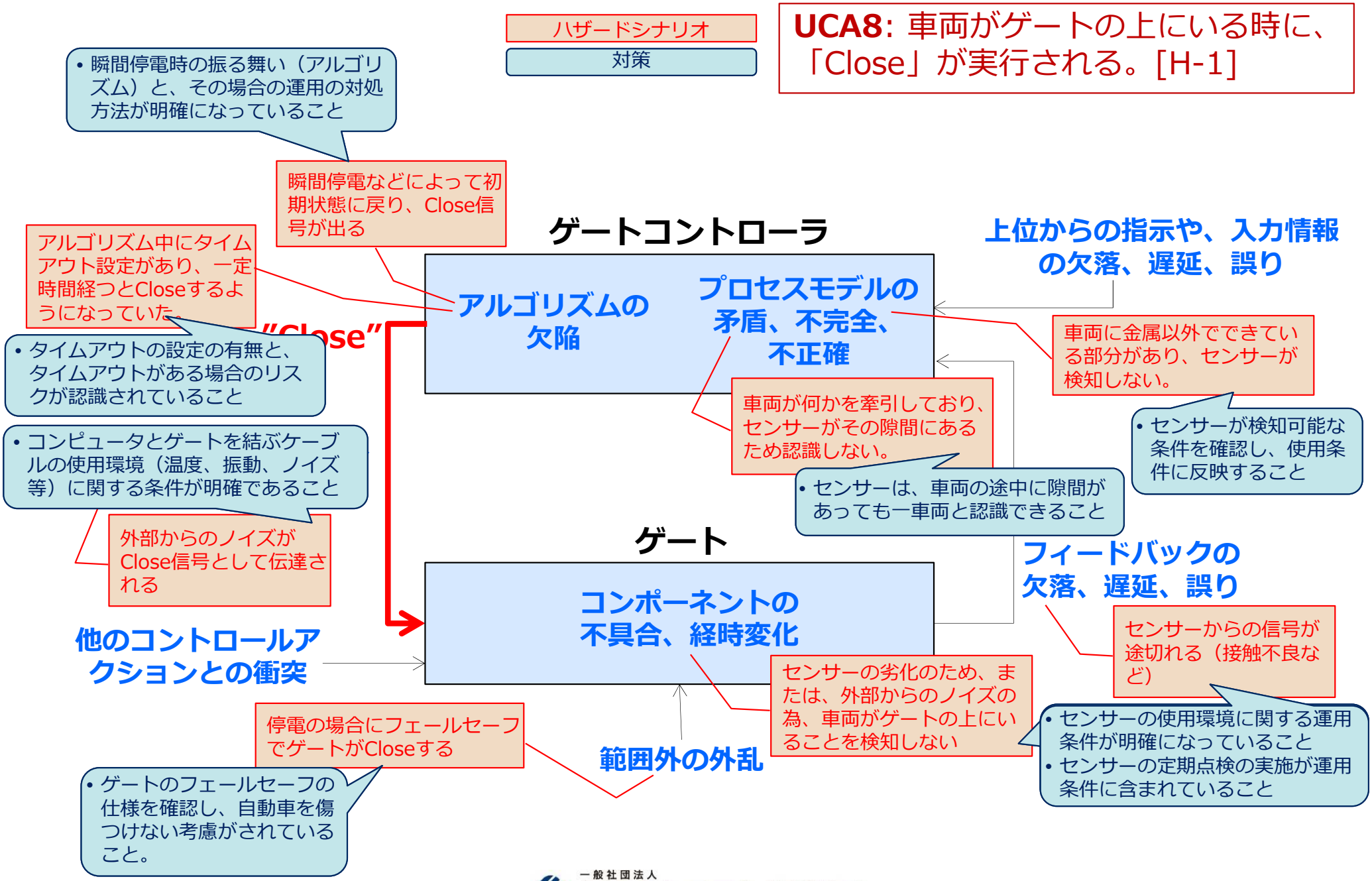
Step0 – 準備２：
コントロールストラクチャー図の作成



Step1：UCAの識別

コントロールアクション	実行されない	不要/不正なものが実行される	早すぎ/遅すぎ/誤順序で実行	長すぎ/短すぎる実行
承認	**UCA1**: 許可のある車両がゲート前にいる時に、「承認」が実行されない。[H-3]	**UCA2:** 許可のない車両がゲート前にいる時に、「承認」が実行される。[H-2]	**UCA3**: 車両の入場許可を確認するより前に「承認」が実行される。[H-2]	---
Open	**UCA4**: 許可のある車両がゲート前にいる時に[Open」が実行されない。[H-3]	**UCA5**: 許可のない車両がゲート前にいる時に、「Open」が実行される。[H-2]	---	**UCA6**: 許可のある車両がゲート前にいる時に「Open」の実行が短すぎる。[H-3]
Close	**UCA7**: ゲート開放時、許可のない車両がゲート前にいるときに「Close」が実行されない。[H-2]	**UCA8**: 車両がゲートの上にいる時に、「Close」が実行される。[H-1]	**UCA9**: 許可車両がゲートを通過後、「Close」の実行が遅すぎるため、後続の無許可車両も通過する。[H-2] **UCA10**: 車両がゲートを通過し終わるより前に「Close」が実行される。[H-1]	**UCA11**: 許可車両がゲートを通過後、「Close」の実行が短すぎるため、ゲートが十分に上がらず、後続の無許可車両も通過する。[H-2]

Step2：UCAの発生要因（HCF）の分析



Q&A