2017年度版

安全誘導型設計の特徴と試行

2018年３月

レンタコーチ代表
中村　洋
www.facebook.com/coachwokarite

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

はじめに

◆背景と経緯

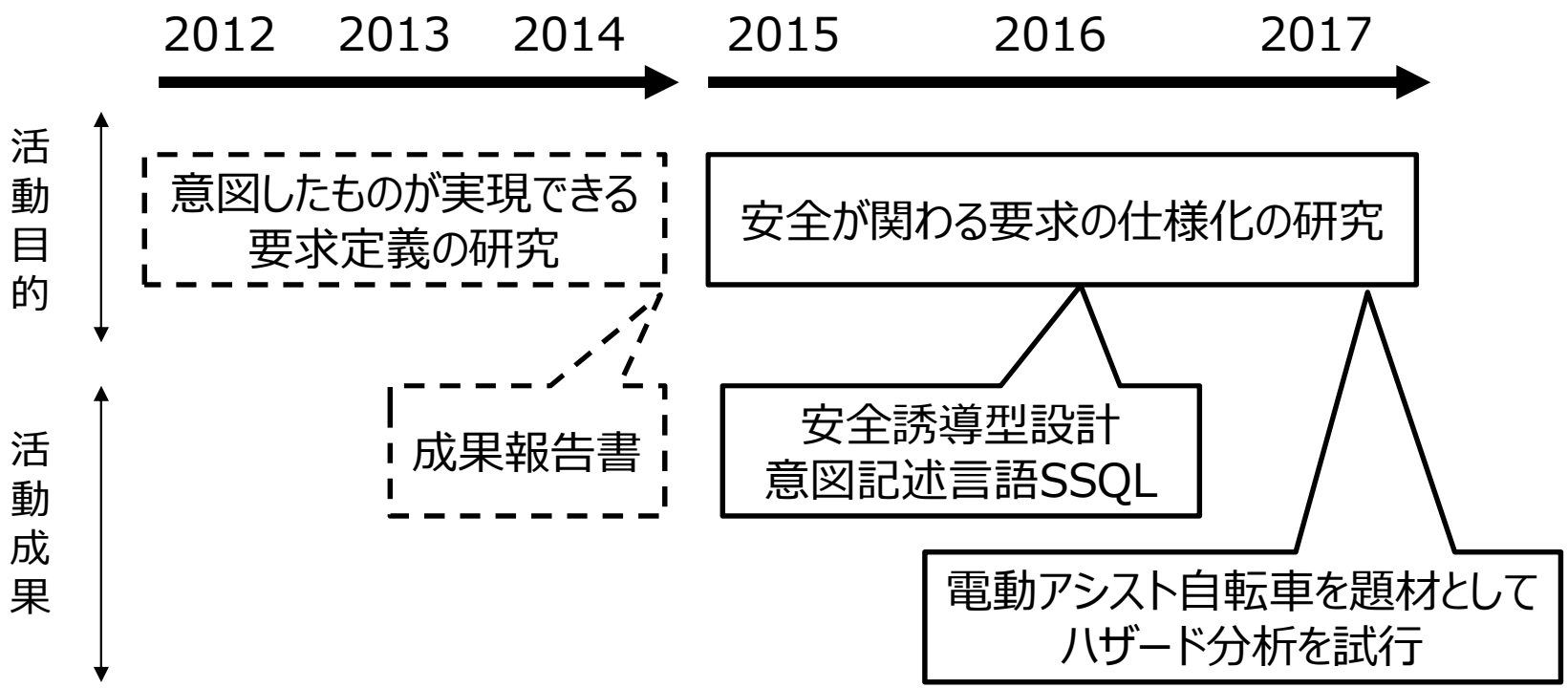
- 安全性向上委員会は2012年度から3年間、意図したものが実現できる要求定義を求めた活動に取組み、2014年度末に成果報告書を公開した。
- 2015年度からは安全仕様化WGに衣替えし、安全が関わる要求を仕様化するプロセスを研究し、それを支援するプロセスモデルと手法を提案する活動を展開している。
- 2016年度には、安全誘導型設計と呼ぶプロセスモデル、意図記述言語SSQLを提案した。
- 2017年度は、仮想的な電動アシスト自転車開発を題材として、安全誘導型設計を試行し、ハザード分析を実施している。

◆要旨

- 安全が関わる要求の仕様化に関して、要求定義に先立つ意図の記述が、要求分析段階における安全実現性の検証が、ともに求められている。
- 安全誘導型設計は、「意図を記述すれば、安全性が高まる」仮説に基づくプロセスモデルである。適用分野固有の意図体系テンプレートを使用し、意図構文にしたがって意図を記述することを特徴とする。
- 仮想的な電動アシスト自転車開発を題材として、安全誘導型設計を試行し、意図記述が、ハザード分析を容易にし、「安全性の検証と安全の作り込み」を助けることを確認した。

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

活動推移



Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

目次

1. 安全が関わる要求の仕様化
2. 安全誘導型設計の特徴
3. 意図記述プロセス
4. 意図記述言語SSQL
5. 安全誘導型設計の試行
   - STAMP/STPAハザード分析
   - FRAM分析

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

現状と課題

安全が関わる要求の仕様化

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

要求の仕様化に関する現状と課題

◆意図が示されない

- 2012年度から3年間、意図したものが実現できる要求定義を求めた活動に取組み、要求の仕様化に関する開発現場における課題をまとめた。
- 顧客との関係では、顧客の意図が示されないこともあるが、それでもソフトウェアは作成できてしまうという課題があった。仕様の間違いに気づきにくいことが問題。

◆暗黙知は要求として書かれない

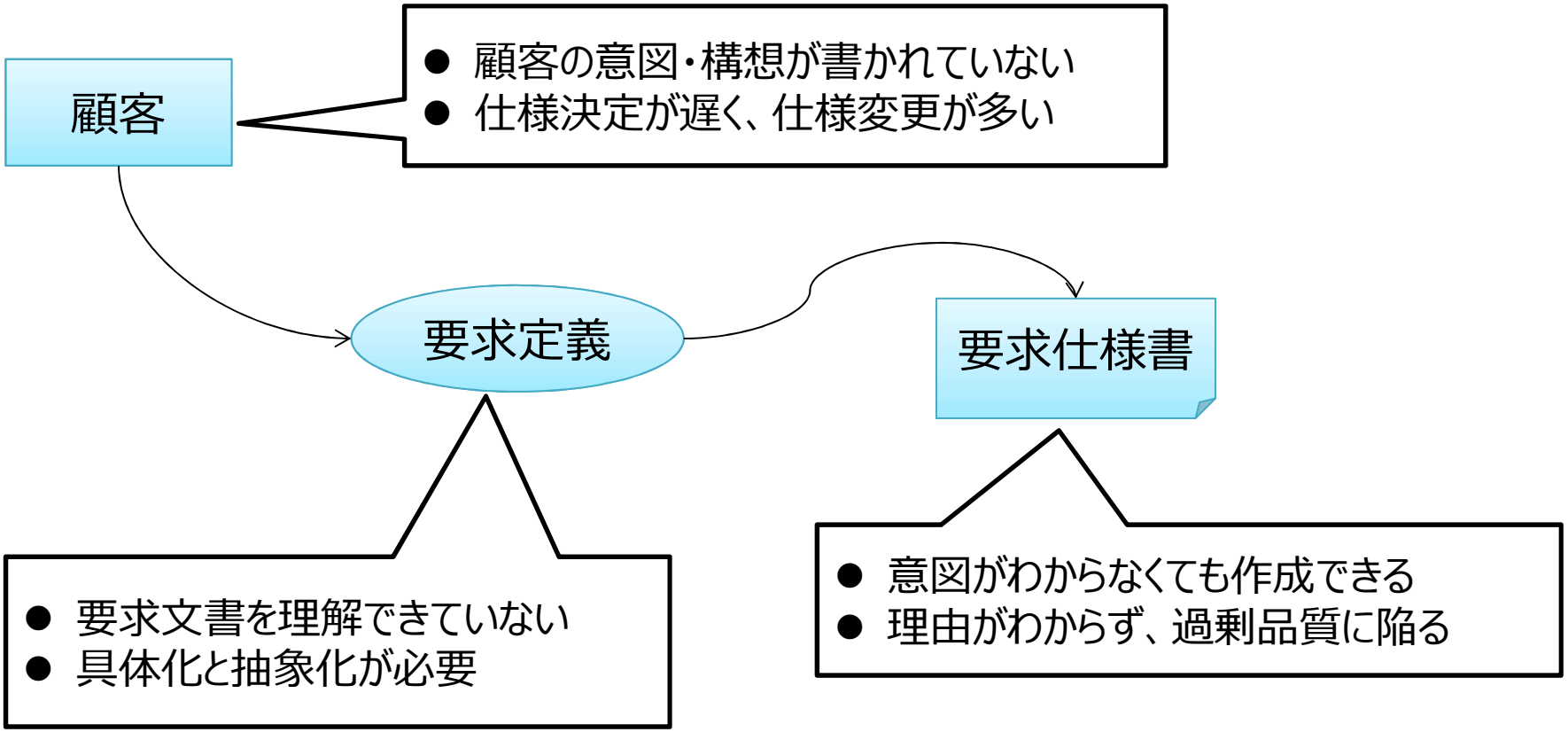
- 顧客側の暗黙の了解、技術的又はビジネス的常識、慣習などが伝わらない現状が、未解決。
- 開発側のオープン化が進むにつれて、意思疎通が一層の課題。

◆安全実現性の検証が難しい

- 技術と社会の高度化、複雑化が進展。
- 要求段階において意図する安全の実現性を検証したいが、適切な手法が定着していないのが現実。

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

参考：開発現場における課題まとめ



出典：要求の仕様化に関する、課題、プロセス及び手法、2015年3月、JASA

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

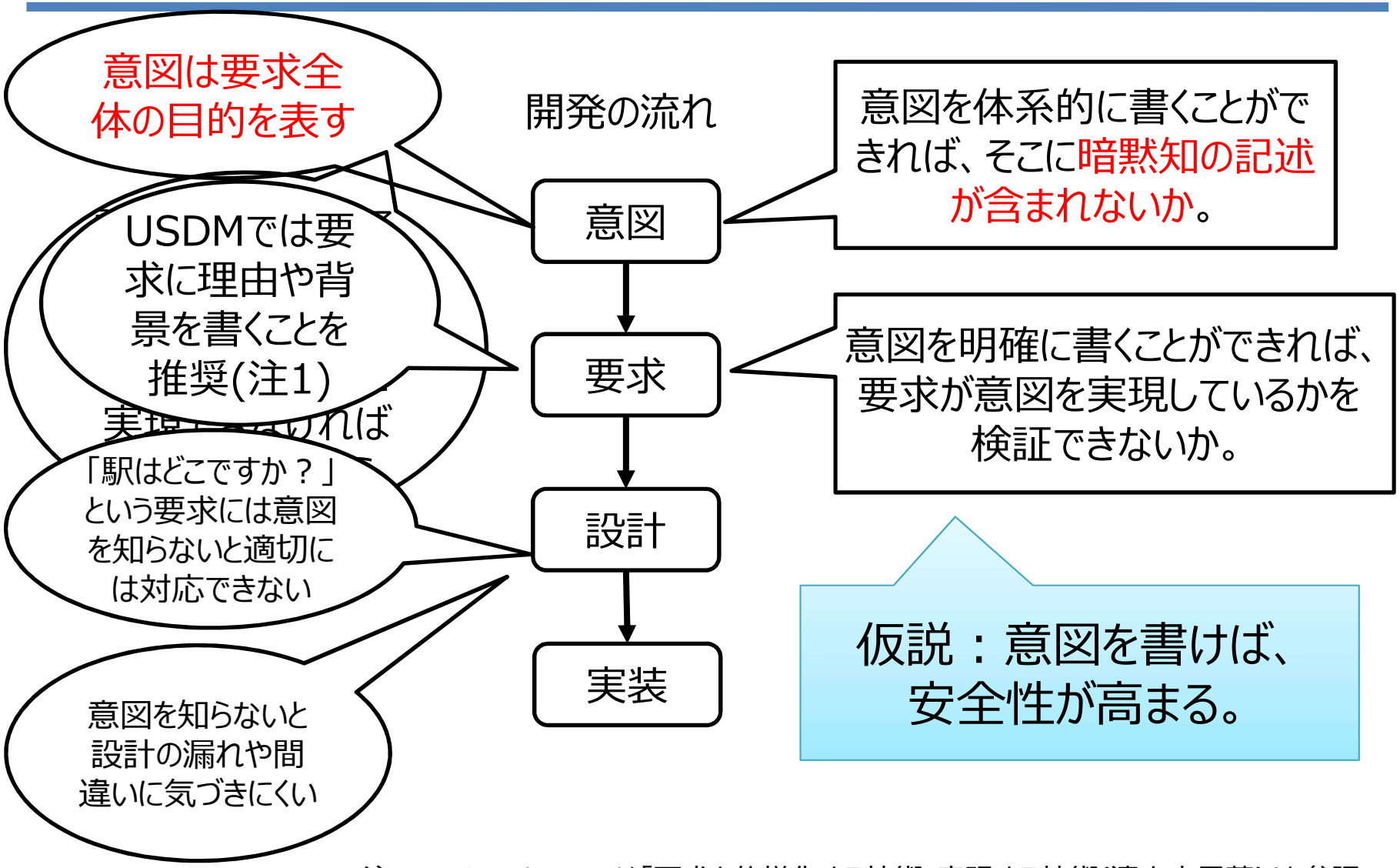
参考：伝統的な安全技術が限界に来ている

- ◆ 技術が早く変化
  - ■ 過去の経験則が陳腐化
- ◆ 経験から学ぶ時間がない
  - ■ 製品化時間の短縮
- ◆ 事故の性質が変化
  - ■ デジタル化やソフト化
- ◆ 新しいハザードが出現
  - ■ 科学や社会の進歩に伴う
- ◆ 複雑性が増大
- ◆ ひとつの事故の損失が増大
  - ■ 金融事故など
- ◆ 優先度を選び、トレードオフを決めることがより困難に
- ◆ 手動と自動との関係がより複雑に
  - ■ 人による制御の比率が増加
- ◆ 監視と規制の重視へ社会が変化
  - ■ 安全責任が個別企業から政府へ

出典：Engineering a safer world、 Nancy Leveson

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

意図の役割と期待効果



注1：USDMについては「要求を仕様化する技術・表現する技術(清水吉男著)」を参照。

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

プロセスモデル

安全誘導型設計の特徴

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

安全誘導型設計の狙い

◆適用分野

- ■ 組込み製品を対象とするシステム開発
- ■ 一般的に、安全が関わるシステム開発

◆適用プロセス

- ■ システム開発においてコンポーネント設計に先立ち、システム全体の要求を分析し、構造を設計するプロセス

◆利点

- ■ 要求分析段階において、安全性の検証と安全の作り込みを支援する。
- ■ 要求仕様が意図したことを実現しているかという、意図実現性の検証を支援する。

命名時の思い：
- ➢ 意図に照らして、安全か非安全かを判断し、
- ➢ 安全が実現する方向に進めば、適切に要求と構造を設計できる

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

安全誘導型設計の手順

1)意図記述

システム開発に関する意図を体系的に、決められた構文規則に従って記述する。

2)システム記述

意図体系を反映して、要求と構造をシステム視点から、かつ適切な仕様記述手法を用いて記述する。

3)ハザード分析

要求と構造が記述されたシステム仕様を参照して、システムレベルのハザードを分析する。ハザードが見つかれば、その対策を講じて、システム記述を繰返す。

4)仕様検証

ハザードが除去されたシステム仕様を意図に照らして検証する。必要あれば、意図記述からやり直す。

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

安全誘導型設計のプロセスモデル

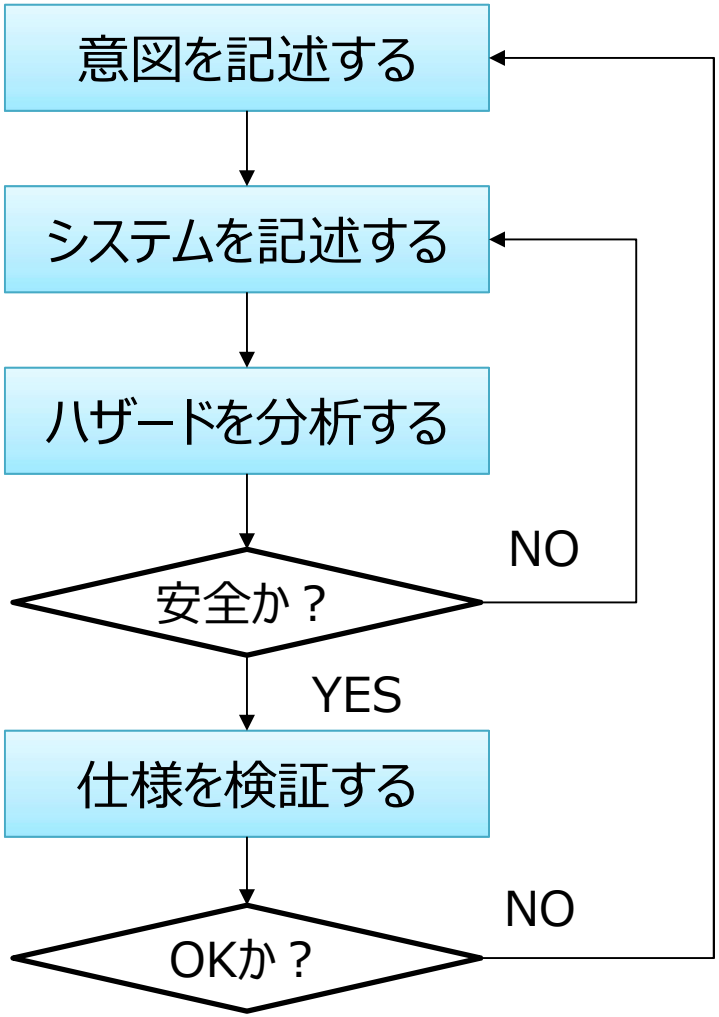
==手法==

意図体系と意図構文に基づいて
安全が関わる
システム開発の意図を記述する
意図記述手法

システム視点から
要求と構造を記述する
仕様記述手法

システム仕様をもとに
ハザードを分析する
安全解析手法

システム仕様に関して
SSQの実現を検証する
仕様検証手法



SSQ:Safety,Security and Quality

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

手法一覧

	意図記述	仕様記述	安全解析	仕様検証
SysML		○		
STAMP/STPA			○	
HAZOP			○	
SLP	○			
Mindmap	○			
VDM++		○		
FRAM			○	
SSQL	○			
Sim4stamp			○	

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

参考：ハザードの定義

故障という用語を使っていない。この点が、信頼性と安全性の違い。

状態という用語を使っているが、事象を使ってもよい。その使い分けが一貫していれば。
事象とすると多くなるが、
状態とすると数が減る。

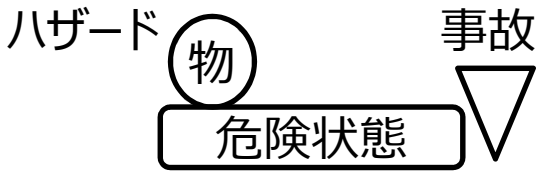
ハザード:
特定の最悪な環境条件と重なって、事故(損失)を引き起こすシステム状態又は条件

システムが制御可能であること

ハザードを回避するために

安全制約

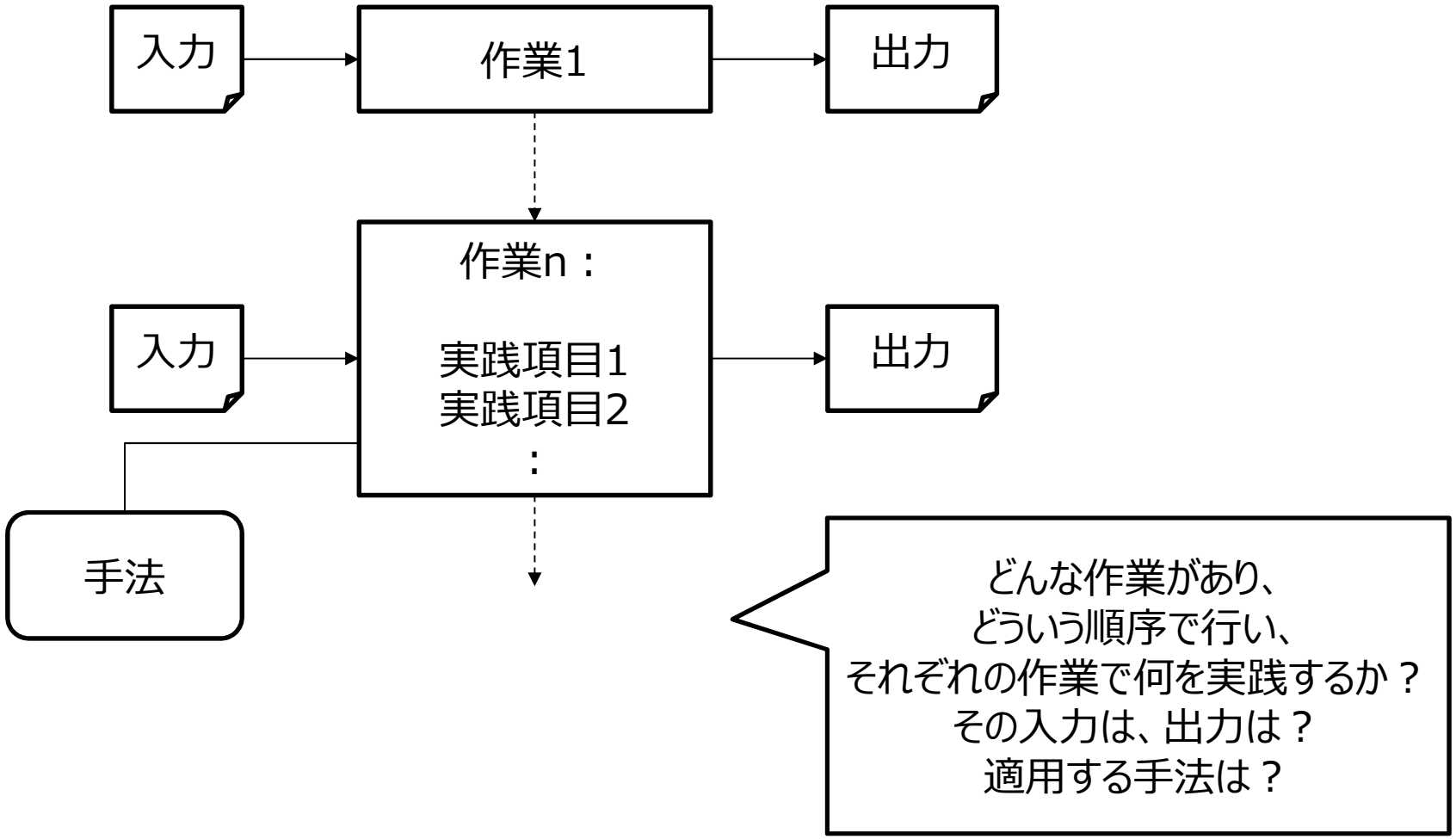
機械安全では:



condition:状態又は条件と訳す。
system state:システム状態。

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

参考:プロセスモデルと手法の関係



Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

意図記述プロセス

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

意図記述プロセスの概要

- ◆ 狙い
  - ■ システム開発の意図に関して、記述すべき視点と事項を体系的に示し、それによって記述漏れの防止を図る。
  - ■ 曖昧さを排除して記述の正確性を高め、あわせて、整合性などの検証を可能とする。
  - ■ 意図と要求の追跡性を可能とする。
- ◆ 意図体系
  - ■ 標準的な意図体系フレームワーク
  - ■ 適用分野固有な意図体系テンプレート
- ◆ 意図構文
  - ■ 意図記述言語SSQL
  - ■ 要求記述ツールSLP
  - ■ EARSパターン、テンプレート
- ◆ 意図と要求の追跡性
  - ■ ゴール構造表記法GSN
  - ■ SysML要求図

SLP: Spec Logically Perfect
EARS:Easy Approach to Requirements Syntax
GSN: Goal Structuring Notation

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

RENTACO

意図体系フレームワーク

第1階層	第2階層	視点
開発計画		開発をマネジメントするための視点
	納期	
	コスト	
	品質を検査する手段	
	安全を検査する手段	
開発目的		開発者と顧客が合意するための視点
	達成目的	
	システムに対する要求と制約	
	使用者に対する要求と制約	
	使用環境に関する仮定事項	
	事故に関する情報	
	ハザードに関する情報	
	システムの限界	

記述事項をさらに細目に分けて、個別の意図を記述。

第1階層	第2階層	視点
システム機能		システム開発者の視点
	製品に関する物理原則	
	システムレベルの設計原則	
システム構造		コンポーネント開発者へ伝達するための視点
	安全関連系	
	新規と流用の区分	
システム運用		運用へ伝達するための視点
	推奨する使用法	
	誤使用	

第1階層が記述すべき視点、第2階層が記述事項を示す。

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

意図体系フレームワークの例-1

第1階層	第2階層	視点
開発計画		マネジメントの視点
	納期	
	コスト	
	品質を検査する手段	
	安全を検査する手段	
開発目的		顧客の視点
	達成目的	
	システムに対する要求と制約	
	使用者に対する要求と制約	
	使用環境に関する仮定事項	
	事故に関する情報	
	ハザードに関する情報	
	システムの限界	

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

意図体系フレームワークの例-2

第1階層	第2階層	視点
システム機能		システム開発者の視点
	製品に関する物理原則	
	システムレベルの設計原則	
システム構造		コンポーネント開発者の視点
	安全関連系	
	新規と流用の区分	
システム運用		利用者の視点
	推奨する使用法	
	誤使用	

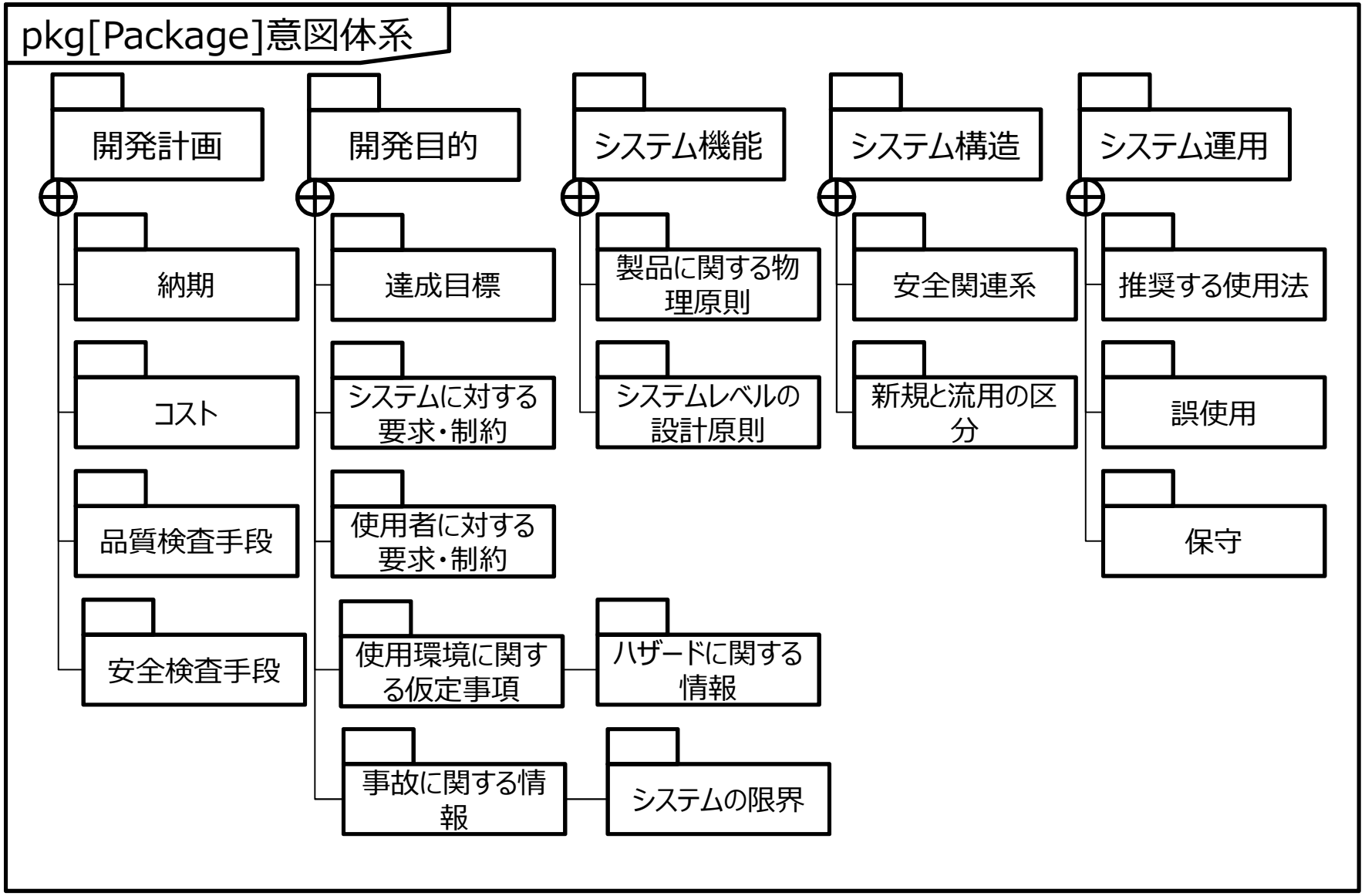
Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

意図体系テンプレートの例

第1階層	第2階層	項目
開発目的	システムに対する要求と制約	
		機能の利点
	使用者に対する要求と制約	
		使用目的
	使用環境に関する仮定事項	
		法的制約
	事故に関する情報	
		多発事故
	ハザードに関する情報	
		重量
		急加速
システム機能	製品に関する物理原則	
		変速機
		速度制限
		前照灯
システム運用	推奨する使用法	推奨する使用法
	誤使用	誤使用

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

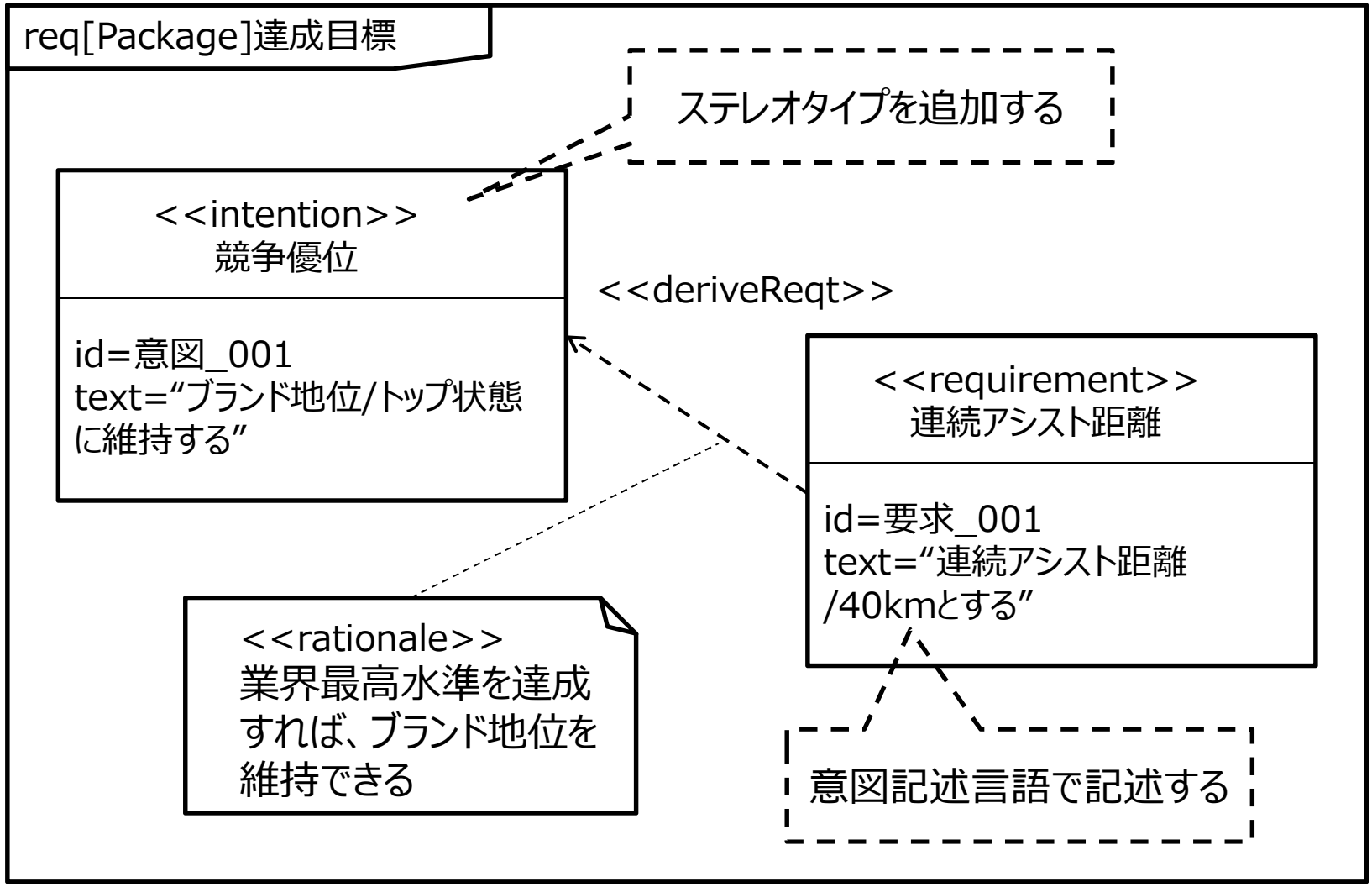
SysMLによる記述:意図体系フレームワーク



Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

RENTACO

SysMLによる記述:意図と要求の関連付け



Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

参考:意図仕様書の構造

改良

部分全体関係の抽象化
(環境、運用者、システム、サブシステム、V&V)

意図の抽象化

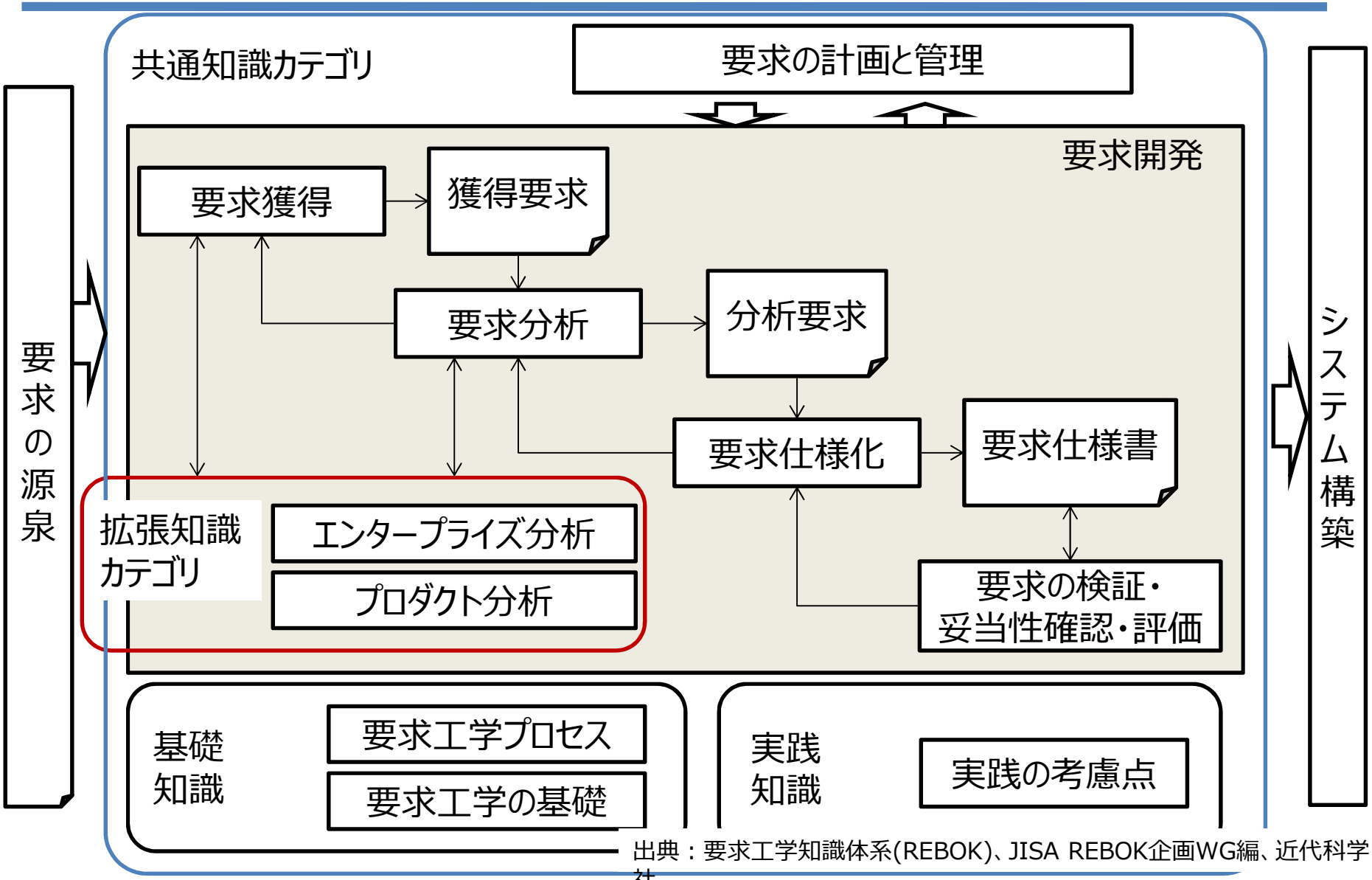
レベル0：プロジェクトマネジメント	プロジェクト計画、安全計画等
レベル1：システム目的	顧客視点で、合意形成に役立つ
レベル2：システム設計原則	システム開発者視点
レベル3：システムアーキテクチャ	システム開発者とコンポーネント開発者とのインタフェース
レベル4：設計表現	コンポーネント設計者視点
レベル5：物理表現	コンポーネント実装者視点
レベル6：システム運用	運用者視点

whatやhowではなく、whyを記述する
(設計の意図、根拠、選択の理由)

出典:Engineering a safer world, N.Leveson

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

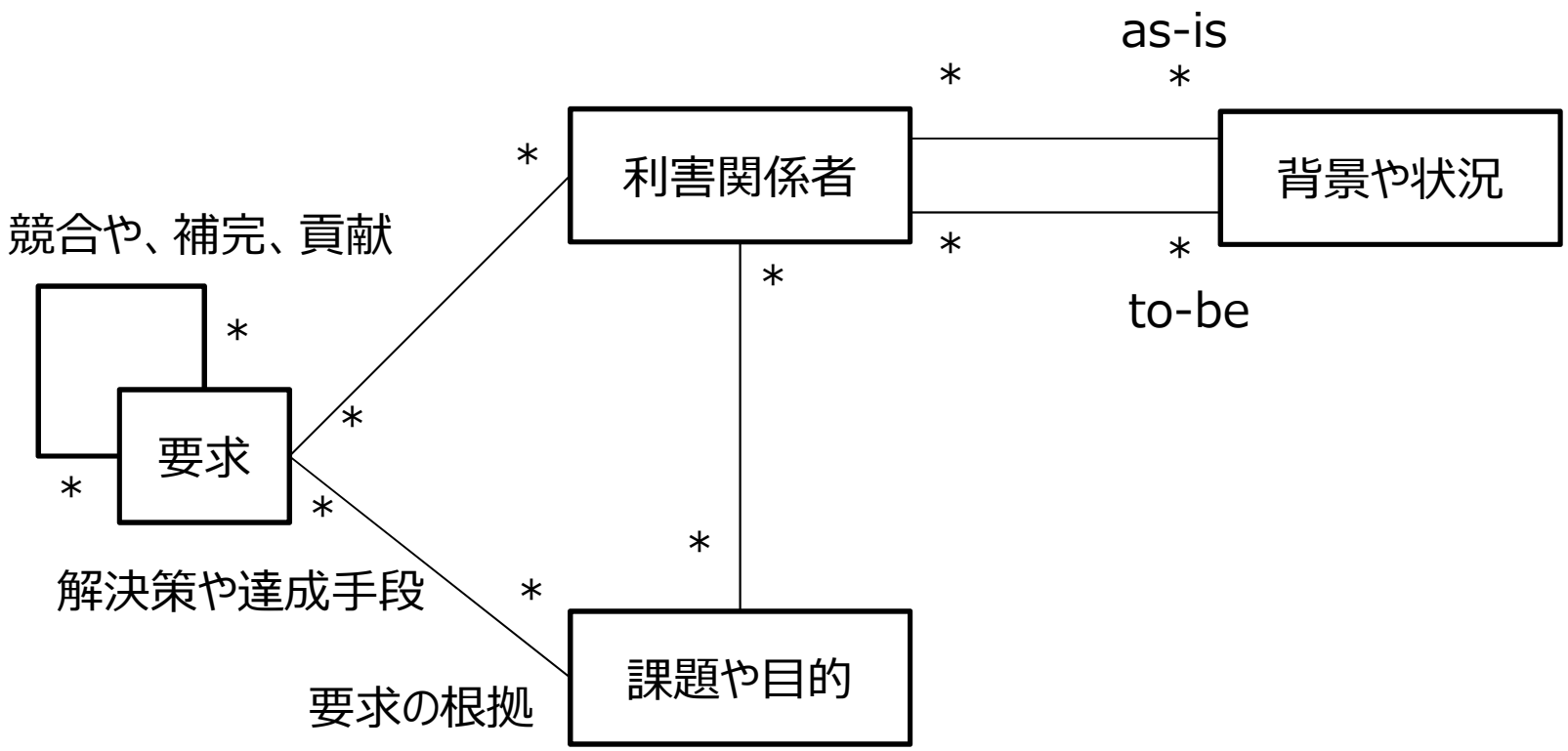
参考:REBOKのプロセスモデル



出典：要求工学知識体系(REBOK)、JISA REBOK企画WG編、近代科学社

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

参考：要求獲得で抽出される情報と要求の関係



備考:クラス図による表記。

出典：要求工学知識体系(REBOK)、JISA REBOK企画WG編、近代科学社

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

参考：要求の特性

特性	意味
単一性	要求の対象が一つであること
完全性	要求に漏れや不完全な記述がないこと
一貫性	要求に矛盾がないこと
法令遵守	法律や規制に準拠していること
独立性	要求がそれ自体で完結し、他の要求に依存していないこと
追跡可能性	要求の源泉や設計など、前後の工程の成果物と関連付けることが可能であること
最新性	要求が最新の条件に基づいていること
実現可能性	要求がプロジェクトや環境などの特別な制約なしに実現可能であること
無曖昧性	要求の中に曖昧さがないこと(たとえば、二つ以上の解釈がないこと)
必須性	要求が必須の内容(それが欠けると、要求の意味がなくなったり、不具合となるもの)を含むこと
検証可能性	要求が検証可能であること

出典：要求工学知識体系(REBOK)、JISA REBOK企画WG編、近代科学社

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

意図記述言語
SSQL

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

SSQLの概要

◆目的
- ■ システム開発に関する意図を記述するための構文を提供する。
- ■ 意図体系テンプレートの記述を可能にする。
- ■ 既存の処理系の活用を可能にする。

◆特徴
- ■ テンプレートに沿って意図とそれに関連する要求を記述できる。
- ■ EARSパターン相当の要求記述構文が使用できる。
- ■ SLPテキストへの変換によってSLP処理系が活用できる。

◆意図・要求構文
- ■ 一般型2種：意図と要求の記述構文
- ■ EARS型4種：EARSパターン準拠の要求記述構文

◆コメント構文
- ■ SSQLコメント：意図体系テンプレートを記述するため
- ■ SLPコメント：SLP言語に同じ

SSQL : SSQ's intentions describing Language

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

意図をSSQL構文で記述できるか？

要求する、規定する、又は選択する意図	if(主語、述語) 目的語、述語
新年度セールスの目玉として販売する。	新年度セールスの目玉/本製品にする
主婦が自分の判断で購入できる。	購入可否/主婦が一人で判断できる
トップブランドの地位を維持する。	ブランド地位/トップ状態に維持する
自然で滑らかな乗り心地を実現する。	
坂道でもパワフルで滑らかな乗り心地を実現する。	
目的外使用に関わる安全にコストをかけない。	if(使用目的/規定範囲以外である)　安全コスト/配慮しない
利用拡大のために法的制約を避ける。	
発生件数が多い事故を避ける。	多発する事故/避ける
アシスト力が急に大きくなることに慣れない。	if(アシスト力/急に大きくなる)　アシスト機能/利用者が慣れないと感じる
高速度での衝突による衝撃を回避する。	if(走行速度/高速である)　衝突による衝撃/回避する

目的語は何か？

条件はあるか？

表現は自由だが、曖昧さが残る

構文化され、不自由だが曖昧さはない

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

意図体系テンプレートを記述できるか

第1階層、第2階層はコメントで

意図項目はコメントで

意図に関連する要求も記述する

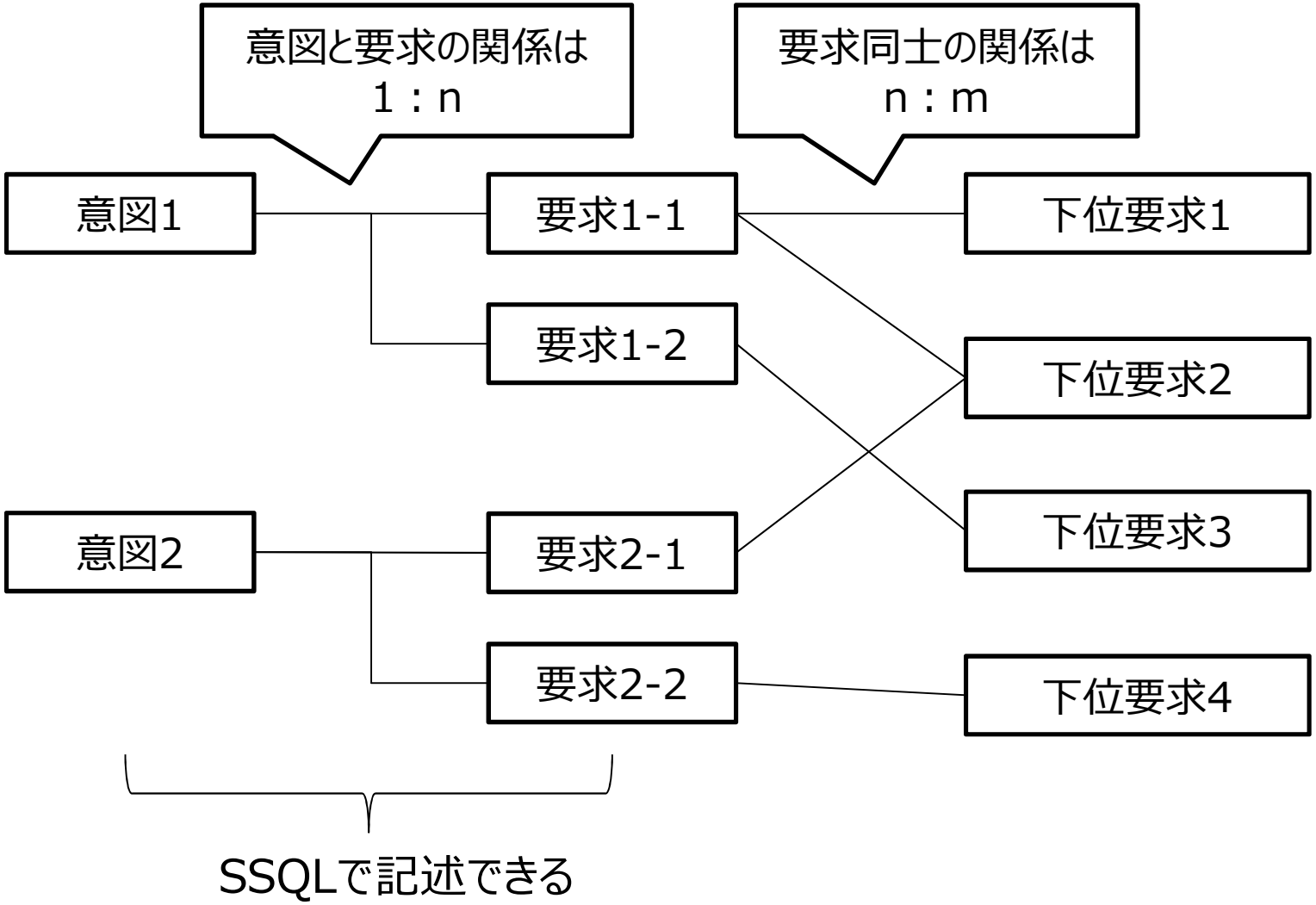
意図体系テンプレート
1.開発計画
2.開発目的
2.1　達成目標
2.1.1　競争優位
　ブランド地位/トップ状態に維持する
(:　　要求
　連続アシスト距離/40kmとする
(:　　業界最高水準を達成すれば、ブランド地位を維持できる

2.2　システムに対する要求・制約
2.3　使用者に対する要求・制約

3.システム機能
4.システム構造
5.システム運用

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

意図と要求の関連付け



Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

SLPテキストへ変換できるか

SSQLコメントをSLPコメントに変換する

意図構文をSLP構文に変換する

```
(:  意図体系テンプレート
(:  1.開発計画
(:  2.開発目的
(:  2.1  達成目標
(:  2.1.1  競争優位
 Do  <ブランド地位>を{トップ状態に維持する}とせよ
(:    要求-1
 Do <連続アシスト距離>を{40kmとする}とせよ
(:    業界最高水準を達成すれば、ブランド地位を維持できる

(:  2.2  システムに対する要求・制約
(:  2.3  使用者に対する要求・制約

(:  3.システム機能
(:  4.システム構造
(:  5.システム運用
```

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

文

	構文規則	SLP変換
記述文	(空白)意図・要求構文	対応するSLP表現
SSQLコメント文	文字列	(:文字列
SLPコメント文	(:コメント	同左

先頭文字で文を区別する:
空白 -> 記述文
(　　-> SLPコメント文
それ以外 -> SSQLコメント文

SLPインポートツール:
SSQLテキストを読み込み、
SLP変換を施し、その結果を
SLP処理系に渡す

文の区切りは改行

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

意図・要求構文

		構文規則	SLP表現
一般型		x/P	Do <x>を{P}とせよ
		if(y/Q) x/P	if <y>が{Q}ならば Do <x>を{P}とせよ else Do nothing endif
EARS型	事象型	when(y/Q) 一般型	if <y>が{Q}という事象が発生したならば 一般型表現 else Do nothing endif
	状態型	while(y/Q) 一般型	if <y>が{Q}という状態にあるならば 一般型表現 else Do nothing endif
	環境型	where(y/Q) 一般型	if <y>が{Q}という環境にあるならば 一般型表現 else Do nothing endif

備考：xとyは「意図対象」。Pはxを目的語とする述語部、Qはyを主語とする述語部。
ここで、「述語部」とは、一つの文の中で主語又は1つの目的語を除く残りの部分とする。

例： if(アシスト力/急に大きくなる) アシスト機能/利用者が慣れないと感じる

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

参考：EARSパターン

	邦訳	要求の内容	時間関連
Ubiquitous	常時実行型	いつでも、どこでも、無条件に行うべき要求	
WHEN	事象応答型	特定の事象が発生したときにする応答	○
IF-THEN	問題対処型	特定の問題が存在するときにする対処	
WHILE	状態駆動型	特定の状態にある間にする処理	○
WHERE	環境依存型	特定の機能が使えるときに行うべき要求	
Complex	複合型	複合的な要求	

EARS:Easy Approach to Requirements Syntax

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

参考：EARS日本語テンプレート案

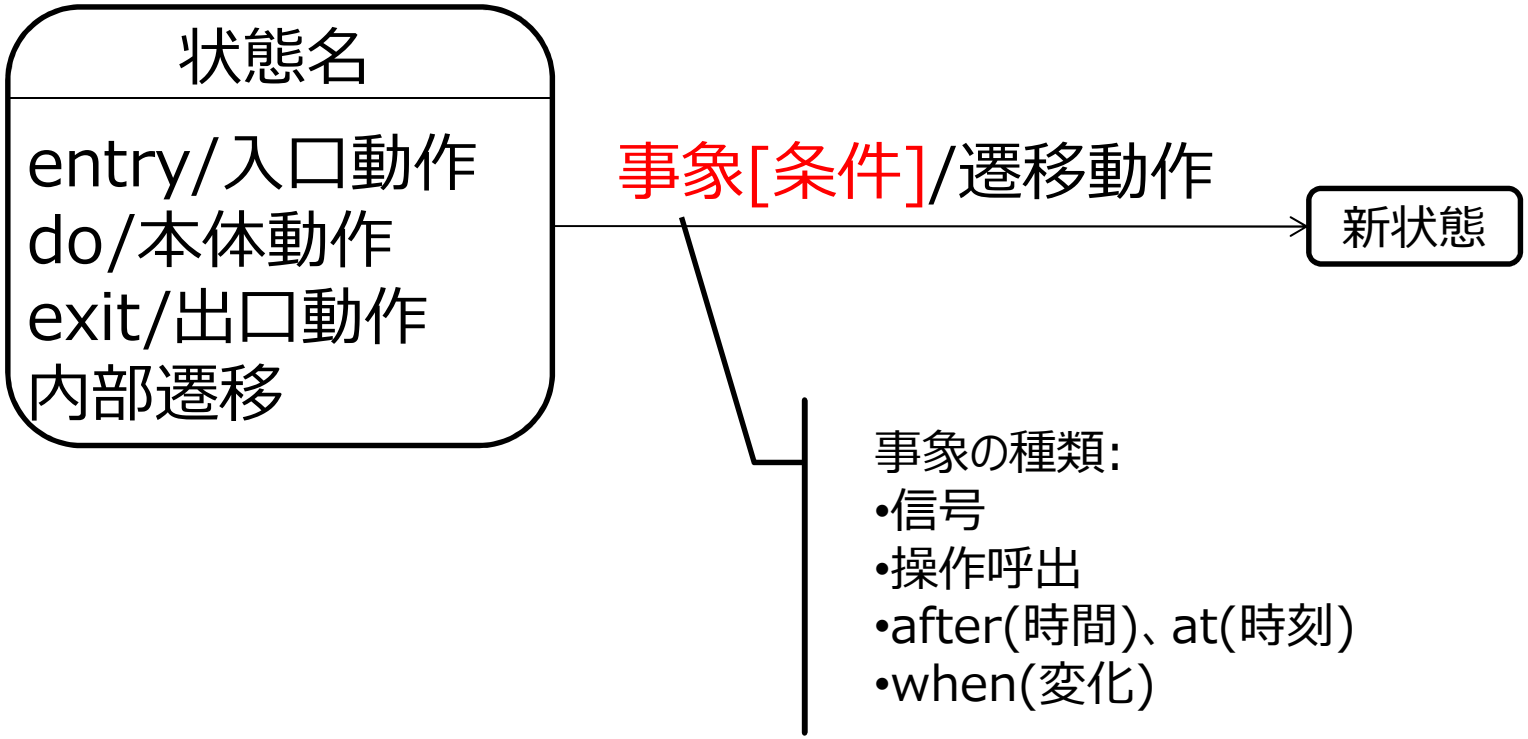
	テンプレート
	<システム行動>=<要素名>は、<システム応答>せよ。
常時実行型	常に、<システム行動>
事象応答型	<事象>という事象が発生したとき、<条件>という条件が成立すれば、<システム行動>
	<事象>という事象が発生したとき、<システム行動>
問題対処型	<問題状況>という問題があれば、<システム行動>
状態駆動型	<特定状態>という状態にある間には、<システム行動>
環境依存型	<特定環境>という環境にあれば、<システム行動>
複合型	<システム行動>が他のテンプレートに変わる

<>は文又は名詞。

時間関連を明示するために、「とき」、「間には」を使用する。

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

参考:条件付きの状態遷移



備考:操作呼出の場合、状態機械に対応する構成要素の操作が呼出されることによって状態が遷移する。

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

安全誘導型設計の試行

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

試行の概要

◆対象とするシステム開発
- ■ 仮想的な電動アシスト自転車の開発

◆対象機能
- ■ バッテリーと自転車本来の機能を除き、電動アシスト機能に限定

◆安全誘導型設計の適用範囲
- ■ 意図記述
  - ●意図体系テンプレートを作成
  - ●SSQLを用いて意図・要求を記述
- ■ ハザード分析
  - ●STAMP/STPA手法を適用

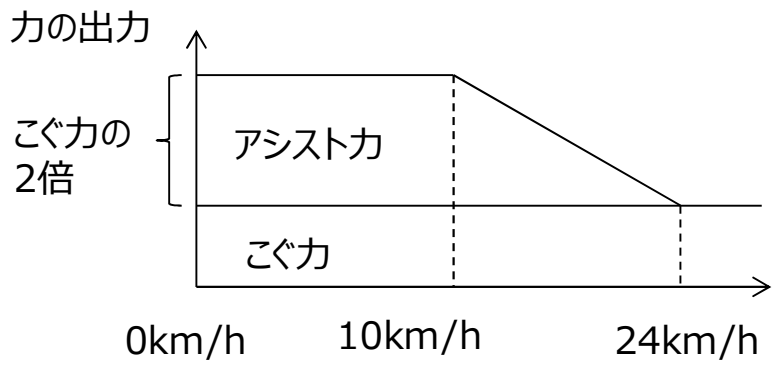
◆参照資料
- ■ ヤマハ製電動アシスト自転車　PASナチュラL取扱説明書

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

電動アシスト機能の主要な特徴

◆最適なアシスト

■ペダルを踏む力や走行速度、変速位置などに応じて基準範囲内でアシストする



◆2種類の走行モード

強モード	急な坂
標準モード	平坦な道

◆3種類のセンサー

■走行速度

■トルク

■クランク回転

◆独自の制御機構

■走行速度とモーター回転速度から変速位置を検出

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

参考:電動アシスト自転車の事故が相次ぐ

経済産業省によると、2014年の出荷台数は約47万台。11年にはオートバイを超え、10年で約２倍になった。鹿児島市など多くの自治体が、高齢者の車の事故防止や環境への配慮のため、電動自転車の購入に補助金を出しているほか、「レンタサイクル」での導入も進んでいる。

普及にともなって事故も増加した。警察庁によると、電動自転車による交通事故件数は15年に1394件。この10年で500件増えた。

公益財団法人「交通事故総合分析センター」（東京・千代田）の調査では、自転車事故の死者数は全体では11年に578人で01年から４割減ったが、電動自転車に限ってみると11年に50人となり、10年間で３倍と急増している。

特に、片足だけペダルに乗せ、もう片足で地面を蹴って勢いをつけて乗り出す「ケンケン乗り」での事故が目立つという。体全体が自転車に乗りきっていない不安定な姿勢のままでモーターが作動し、急加速してしまうことがあるためだ。

アシスト機能が優れていることで、車体の重さに気付きにくくなる点も危険という。車体は25～30キロの重量があり、自転車の専門家らでつくる「自転車の安全利用促進委員会」（東京・渋谷）の一員で自転車ジャーナリストの遠藤まさ子さんは「子供２人と親子で乗れば、全体で100キロを超える。アシスト機能によりスピードも出るので、ぶつかった時の衝撃が大きい」と指摘する。

出典:5/10/2016付けの日経電子版

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

電動アシスト自転車開発の意図体系テンプレート

1.開発計画
1,1　納期
1.1.1　発売時期
1.2　コスト
1.2.1　販売価格
1.3　品質を検査する手段
1.4　安全を検査する手段
2.開発目的
2.1　達成目標
2.1.1　競争優位
2.2　システムに対する要求・制約
2.2.1　機能の利点
2.3　使用者に対する要求・制約
2.3.1　使用目的
2.4　使用環境に関する仮定事項
2.4.1　法的制約
2.5　事故に関する情報
2.5.1　多発事故

3階層目が意図項目

2.6　ハザードに関する情報
2.6.1　急加速
2.6.2　重量
3.システム機能
3.1　製品に関する物理原則
3.1.1　変速機
3.1.2　前照灯
3.1.3　トリプルセンサー
3.1.4　バッテリー
3.1.5　自己診断
4.システム構造
4.1　安全関連系
4.2　新規と流用の区分
4.2.1　外部調達
5.システム運用
5.1　推奨する使用法
5.1.1　不慣れ事故
5.2　誤使用
5.2.1　不用意な使用
5.3　保守
5.3.1　定期検査

Copyright© 2016-2018 Hiroshi NAKAMURA. All Ri

SSQLによる意図・要求の記述

意図は、実現可能性などの要求特性を満たす必要がない。
それでいて、つい、要求を書いてしまう。
しかし、意図は、要求を縛る。

2.開発目的
2.1　達成目標
2.1.1　競争優位
　ブランド地位/トップ状態に維持する
(:　要求
　連続アシスト距離/40kmとする

要求は、実現可能性、検証可能性などの要求特性を満たさなければならない。

(:　業界最高水準を達成すれば、ブランド地位を維持できる

意図を実現できるという論証(理由)も必要になる。

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

意図体系テンプレートに沿った意図・要求の記述

1.開発計画
1,1　納期
1.1.1　発売時期
　新年度セールスの目玉/本製品にする
(:　要求
　販売開始時期/3月後半とする
1.2　コスト
1.2.1　販売価格
　購入可否/主婦が一人で判断できる
(:　要求
　販売価格/10万円未満とする
1.3　品質を検査する手段
1.4　安全を検査する手段
2.開発目的
2.1　達成目標
2.1.1　競争優位
　ブランド地位/トップ状態に維持する
(:　要求
　連続アシスト距離/40kmとする
(:　業界最高水準を達成すれば、ブランド地位を維持できる

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

2.2 システムに対する要求・制約

2.2.1 機能の利点

乗り心地/自然で滑らかにする
乗り心地/坂道でもパワフルで滑らかにする

(: アシスト基準

while(走行速度/時速10km以下) アシスト力/ペダルを踏む力1に対して最大で2まで
while(走行速度/時速10km超で24km未満) アシスト力/順次弱める
while(走行速度/時速24km以上) アシスト力/ゼロ

(: SPEC3制御

while(変速位置/1速) アシスト力/時速10kmの手前で弱め始め、24kmの手前で止める
while(変速位置/2速) アシスト力/時速24kmの手前で止める
while(変速位置/3速) アシスト力/基準どおり

(: 走行モード

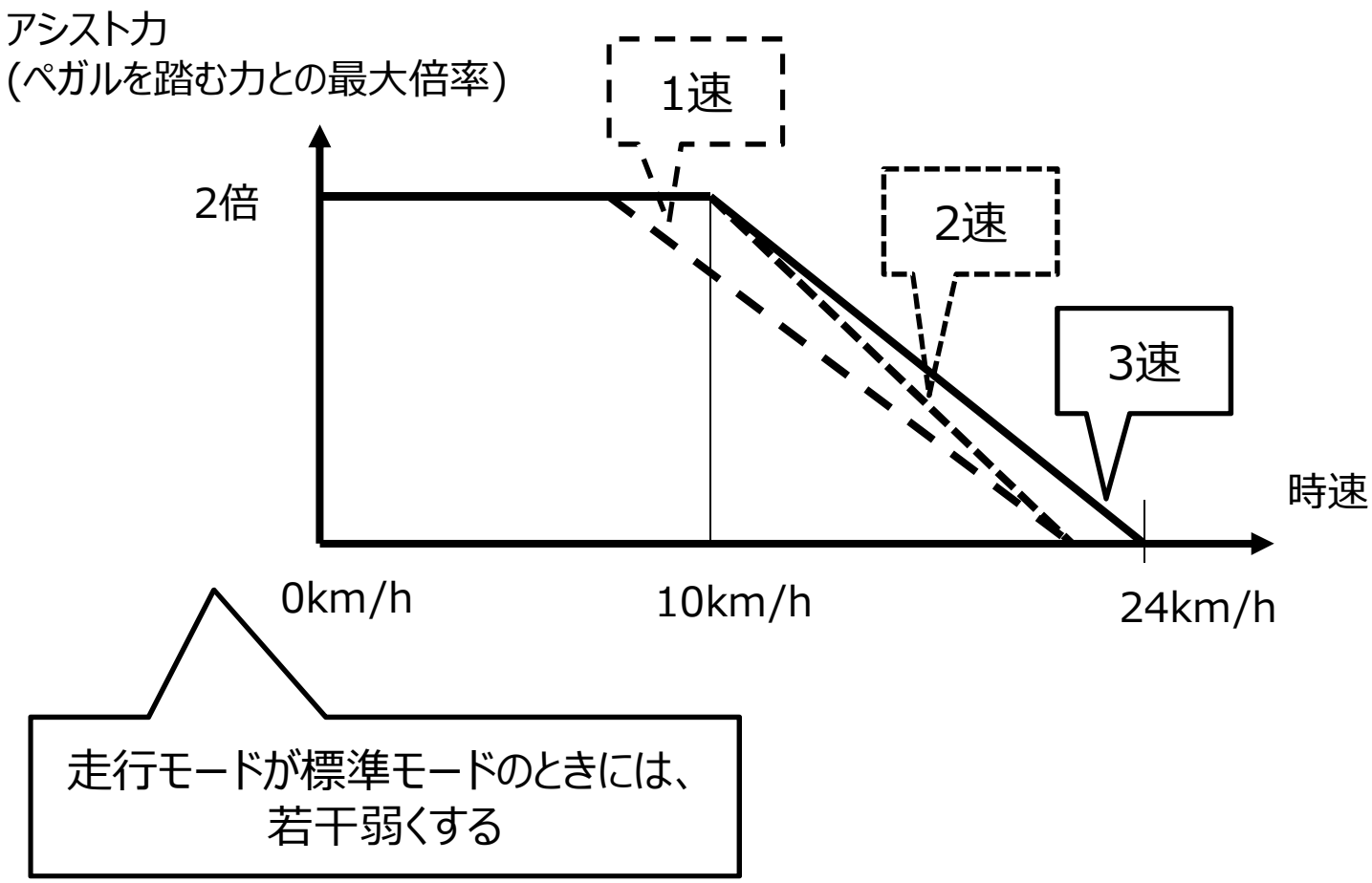
while(走行モード/強モードにある) アシスト力/標準モードより強い
when(電源/オンになる) 走行モード/標準モード

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

3.システム機能
3.1　製品に関する物理原則
3.1.1　変速機
　変速位置/電気的には検出できない
(:　要求
　変速機/内装3段方式とする
　変速位置/走行速度とモーター回転数から算出する
3.1.2　前照灯
　無灯火運転/確実に防止する
(:　要求
　バッテリー/前照灯のために使用する
　前照灯/照度センサーで周囲の明かりを検出し、自動的にオンオフする
3.1.3　トリプルセンサー
(:　要求
　ペダルを踏む力/トルクセンサーで検出する
　走行速度/スピードセンサーで検出する
　ペダルを回す速さ/トランク回転センサーで検出する
3.1.4　バッテリー
(:　要求
　when(ペダルを踏む力/5分以上かかっていない)　電源/オフにする
　while(バッテリー/切れている)　本製品/普通の自転車として使える

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

アシスト基準



Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

STAMP/STPAハザード分析

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

アクシデント、ハザード、安全制約の識別

アクシデント	ハザード	安全制約
運転中の転倒又は衝突によるけが	意図しない急加速	意図しない急加速を防止する

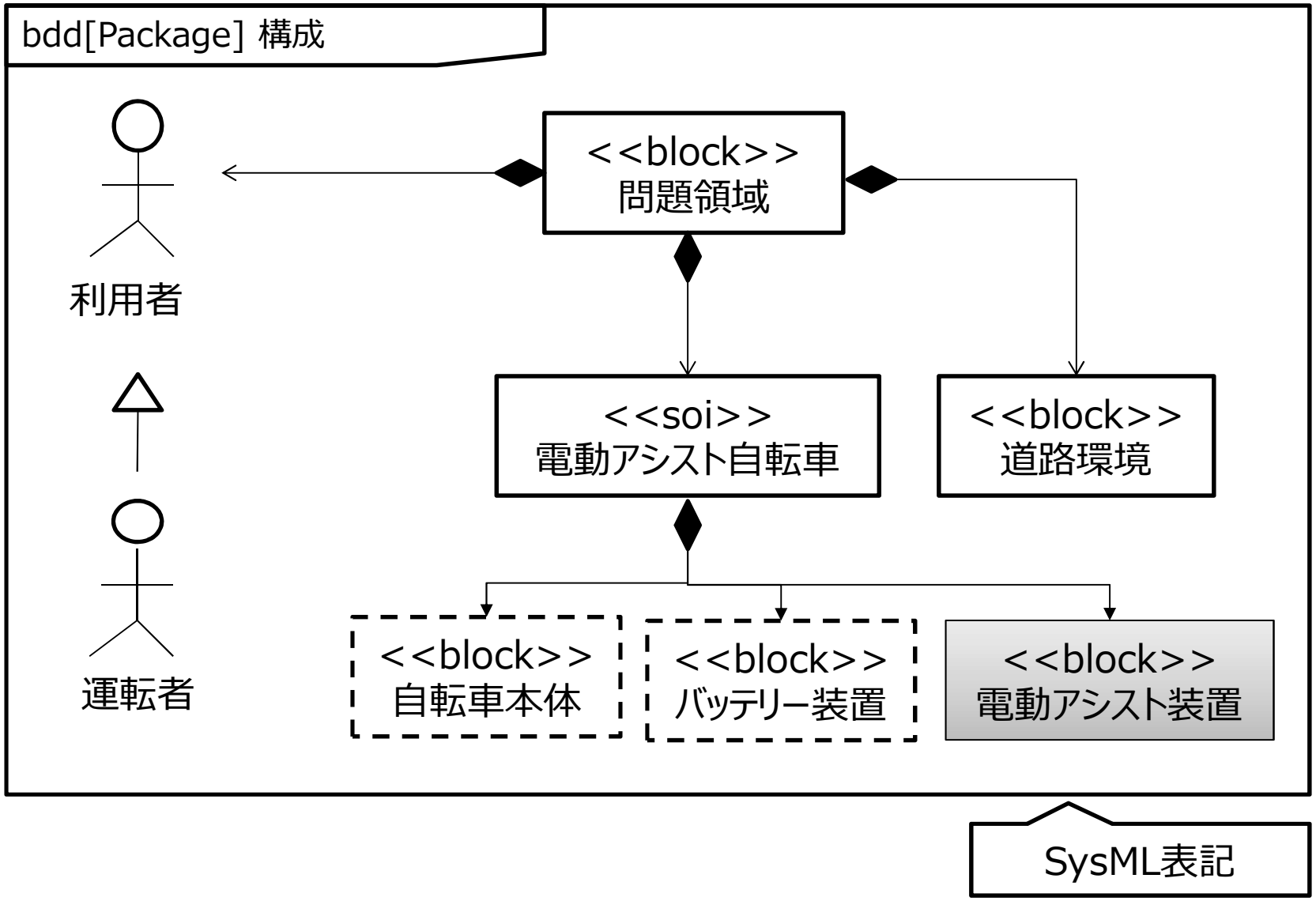
次の意図記述から識別:

2.5　事故に関する情報
2.5.1　多発事故
　転倒によるけが/避ける
　衝突によるけが/避ける
(:　要求
　予期しないアシスト/急に作動させない
　速度の出し過ぎ/防止する

前提事項:
バッテリ関係、本来の自転車機能は分析の対象外とする

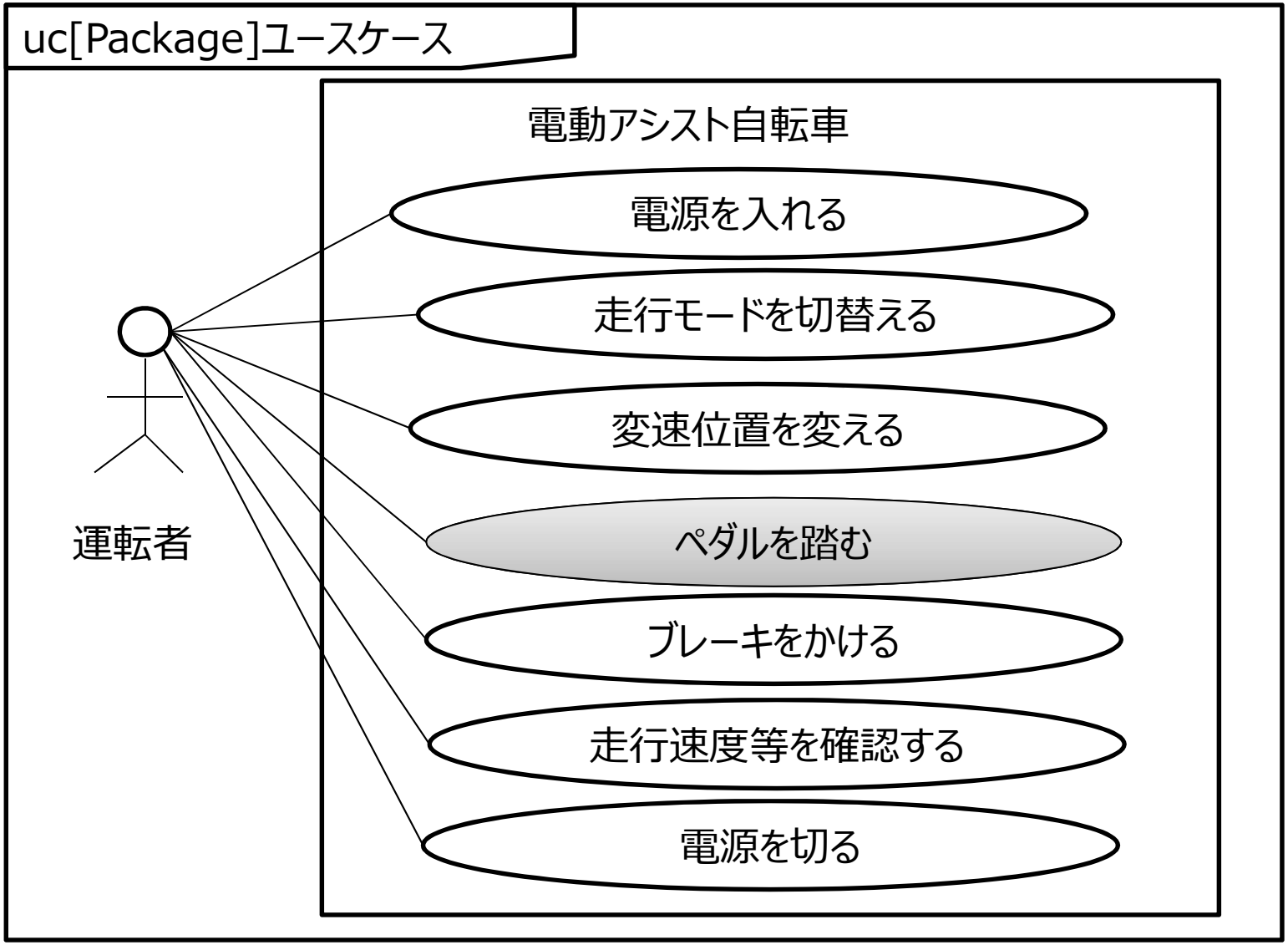
Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

システムコンテキスト図：分析の範囲



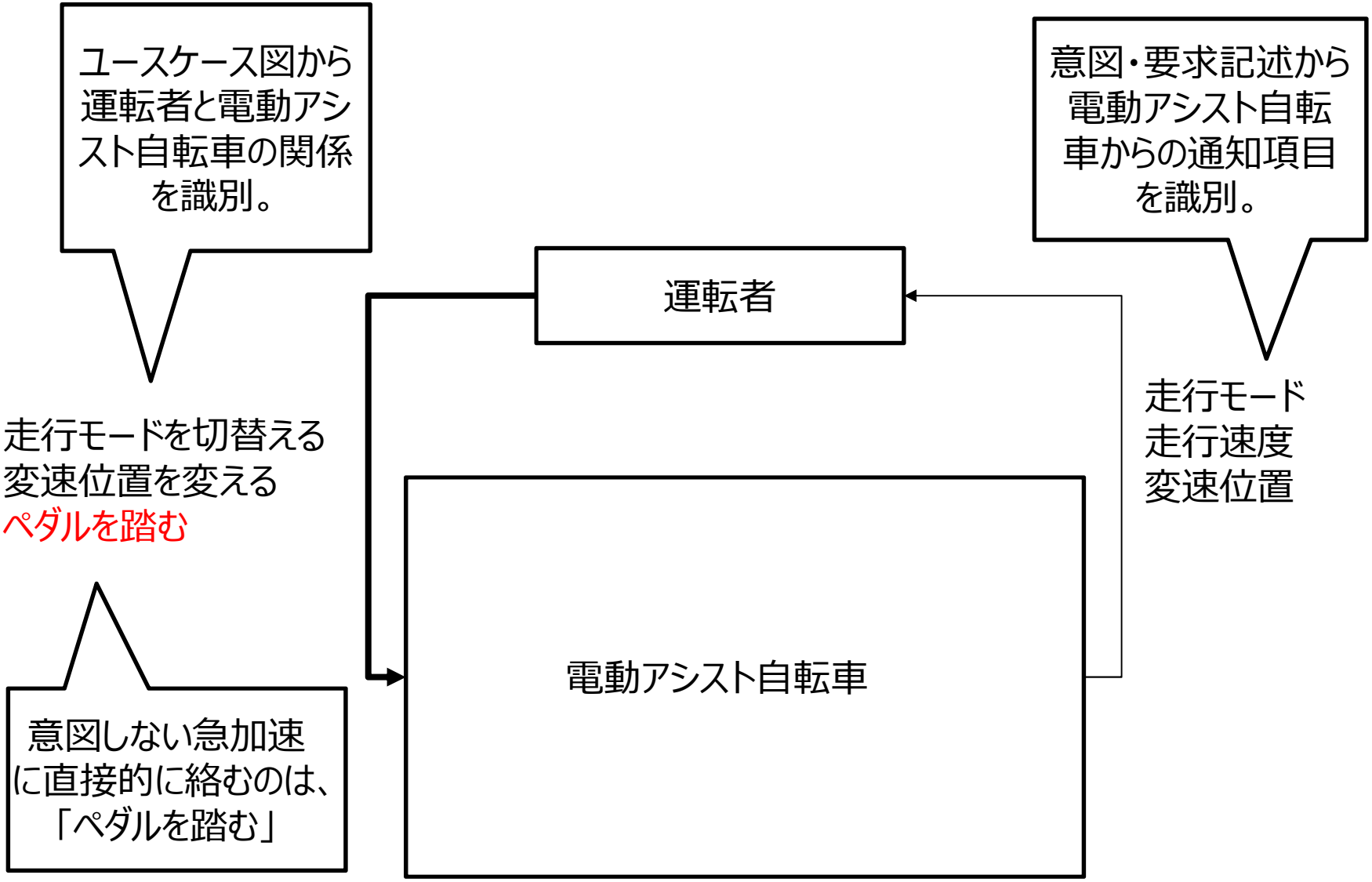
Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

ユースケース図：運転者は何ができるか



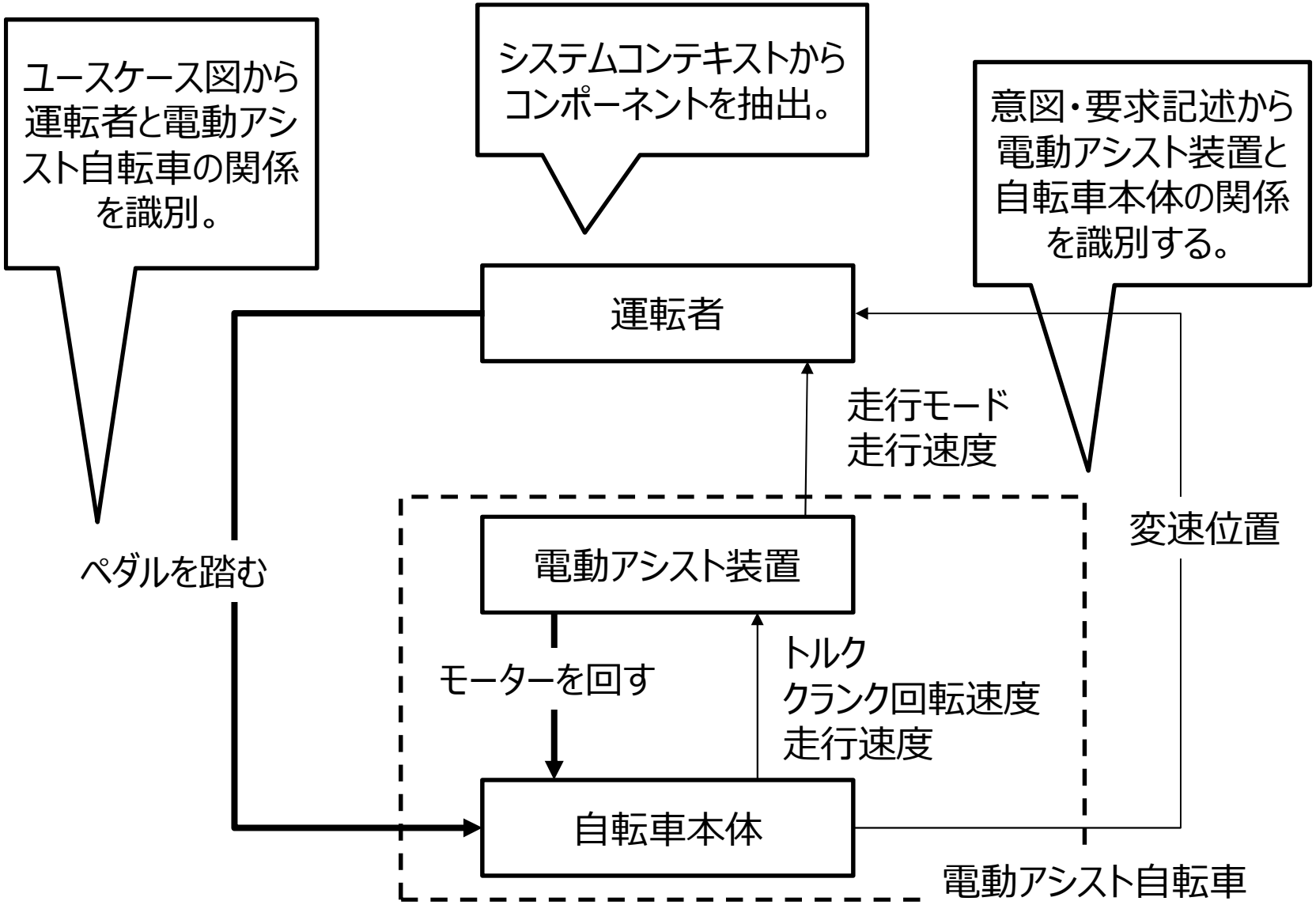
Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

運転者と電動アシスト自転車の相互作用



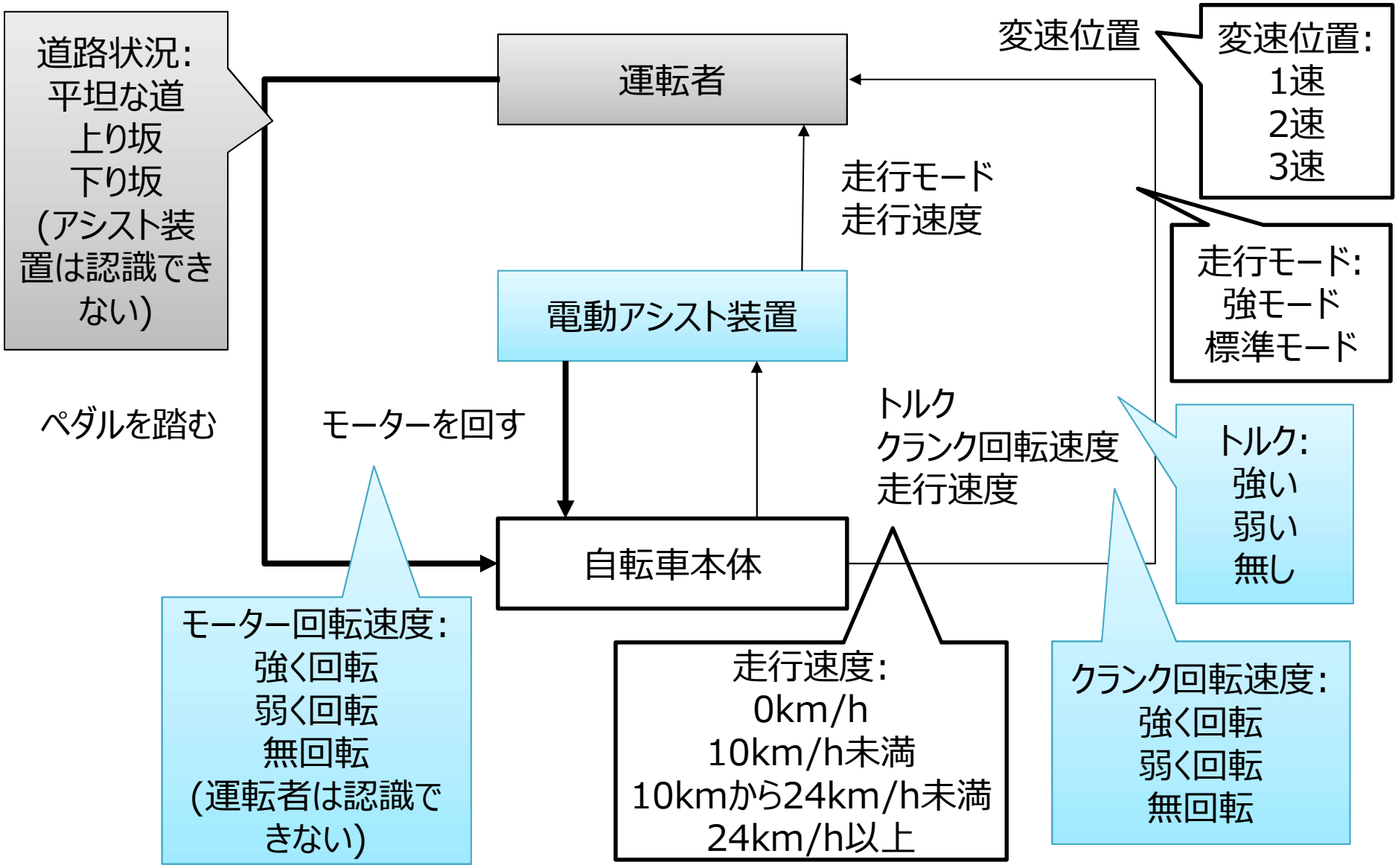
Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

制御構造図



Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

制御行動に影響を及ぼす運転状況



Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

状況変数の抽出

状況変数	制御行動を反映		フィードバックに対応	運転者認識の可否	同値分割数
	ペダルを踏む	モーターを回す			
トルク	✓		✓	X	3
クランク回転数	✓	✓	✓	X	3
走行速度	✓	✓	✓	○	3
モーター回転速度		✓		X	3
走行モード			✓	○	2
変速位置			✓	○	3
道路状況				○	3

「ペダルを踏む」に関しては：

- トルクとクランク回転数はモーター回転速度を変化させるので、モーター回転速度で代用できる。
- 走行モード、変速位置、道路状況は、ペダルの踏み方に影響されない。

状況数は1458とおり

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

RENTACO

UCAを識別する：最大アシスト基準の場合

変速位置:3速、走行モード:強モード、道路：平坦な道

与えられないとハザードは安全側

状況		運転者：ペダルを踏む
走行速度	モーター速度	与えられるとハザード(P)
0km/h	無回転	乗り出し時にペダルを強く踏むと、急加速が行われてハザードになる(UCA1)
10km/h未満	強く回転	強く踏んでも、加速されない
	弱く回転	ペダルを回しながら急に強く踏むと、加速されるが、急加速にはならない
	無回転	ペダルを回していても、いなくても、急に強く踏むと、急加速が行われてハザードになる(UCA2)
10km～24km/h	強く回転	N/A(強く回転することはない)
	弱く回転	急加速にはならない
	無回転	急加速にはならない

ペダルが回っていても、モーターが回っていないときもあると仮定

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

UCAを識別する：上り坂の場合

変速位置:1速、走行モード:強モード、道路：上り坂

与えられないとハザードは安全側

状況		運転者：ペダルを踏む
走行速度	モーター速度	与えられるとハザード(P)
0km/h	無回転	乗り出し時にペダルを強く踏むと、急加速が行われてハザードになる(UCA1)
(10-α)km/h 未満	強く回転	強く踏んでも、加速されない
	弱く回転	ペダルを回しながら急に強く踏むと、加速されるが、急加速にはならない
	無回転	N/A(常にペダルを踏んでいるから、モーターは回転している)
(10-α)km～(24-β)km/h	強く回転	N/A(強く回転することはない)
	弱く回転	急加速にはならない
	無回転	N/A(常にペダルを踏んでいるから、モーターは回転している)

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

UCAを識別する：下り坂の場合

変速位置:2速、走行モード:標準モード、道路：下り坂

与えられないとハザードは安全側

状況		運転者：ペダルを踏む
走行速度	モーター速度	与えられるとハザード(P)
0km/h	無回転	乗り出し時にペダルを強く踏むと、急加速が行われてハザードになる(UCA1)
10km/h未満	強く回転	強く踏んでも、加速されない
	弱く回転	ペダルを回しながら急に強く踏むと、加速されるが、急加速にはならない
	無回転	ペダルを回していても、いなくても、急に強く踏むと、急加速が行われてハザードになる(UCA2)
10km～(24-β)km/h	強く回転	N/A(強く回転することはない)
	弱く回転	急加速にはならない
	無回転	急加速にはならない

最大アシスト基準の場合に同じ

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

ハザードシナリオを識別し、対策を考える

UCA	ハザードシナリオ	対策
UCA1: 走行速度0km/h モーター無回転 ペダルを強く踏む	1. 乗り出しのときに、運転者が慌ててペダルを強く踏んでしまう。(スリップ) 2. トルクが急に大きくなる。 3. それを検知して、アシスト装置がモーターを急に強く回す。 4. 自転車が急に加速され、ハザードを引き起こす。	モーターが無回転のときには、トルクが急に大きくなっても、モーターを強く回さない。
UCA2: 走行速度10km/h未満 モーター無回転 ペダルを強く踏む		

人に関する制御行動を対象に

1. オミッションエラー：適切な指令を行わない
2. ミステイク：適切でない指令を行う
3. スリップ：指令は適切だが、間違えて実施する

ヒューマンエラーの分類に基づくガイドワード

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

ハザードシナリオの識別と対策

UCA	ハザードシナリオ	対策
UCA1: 走行速度0km/h モーター無回転 ペダルを強く踏む	1. 乗り出しのときに、運転者が慌ててペダルを強く踏んでしまう。(スリップ) 2. これ以降はUCA2に同じ。	対策1に同じ
UCA2: 走行速度10km/h未満 モーター無回転 ペダルを強く踏む	1. ゆっくりと走っているときに、加速しようとして、運転者がペダルを強く踏んでしまう。(ミステイク) 2. トルクが急に大きくなる。 3. それを検知して、アシスト装置がモーターを急に強く回す。 4. 自転車が急に加速され、ハザードを引き起こす。	対策1： モーターが無回転のときには、トルクが急に大きくなっても、モーターを強く回さない。 対策2： 運転者がモーター回転速度を認識できるようにし、無回転時には強くペダルを踏まないように注意喚起する。

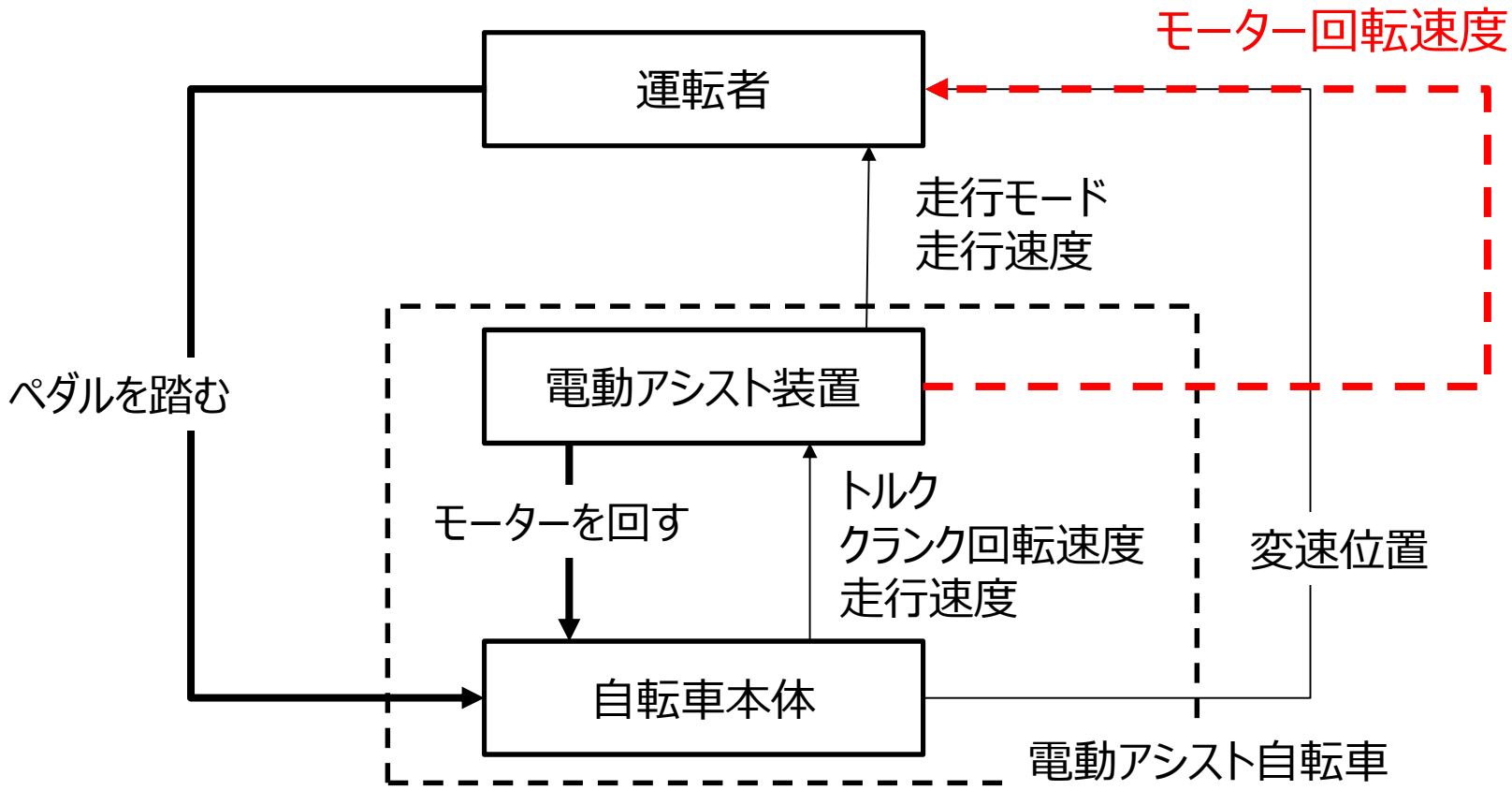
両ケースとも、
モーターが回転せず、
アシストが働いていない

「乗り心地を自然で滑らかにする」という意図の実現には、対策2が推奨策。

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

対策2：フィードバックの追加

電動アシスト装置の動作状態を反映するフィードバックがなかった
(制御構造の観察から発見可能か)



Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

参考：アクセル踏み間違え、急発進させない

対策1に相当する

　自動車用品大手オートバックスセブンは、運転中の高齢者らがアクセルとブレーキのペダルを踏み間違えた際に、車が急発進するのを防ぐ装置「ペダルの見張り番」を、来月５日に売り出す。停止時や徐行時にアクセルを強く踏んでも、電子的に制御して加速を抑える仕組み。
　アクセルとブレーキを両方踏んだ場合にブレーキを優先する機能も備え、無料の会員登録をすれば１年間の交通事故傷害保険もつく。国産車１００車種以上に対応し、オートバックスの全国の店舗で取り付け作業を受け付ける。工賃込みで税抜き３万９９９９円。
　高齢者らのペダル踏み間違いが原因とみられる事故が相次いでおり、自動車メーカー各社も、周囲の障害物を検知して加速を抑える機能などを備えた車を売り出している。

出典:11/30/2016付けの朝日デジタル

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

考察

- ◆要求分析段階で安全性を検証し、安全を作り込むためには、意図の体系的な記述が役に立つと考える。
- ◆意図を体系的に記述するときには、意図体系テンプレートに沿った記述が便利であり、その元になる標準的なフレームワークの整備が望まれる。
- ◆意図は、要求とは異なり、要求特性に制約されないが、要求の目的や根拠などの役割を担い、その記述には訓練、演習が必要だ。
- ◆意図とそれを反映する要求を体系的に記述し、その結果をもとにしてハザード分析を行うと、適切な安全制約を導き出すことができた。
- ◆「意図を記述すれば、安全性が高まる」と主張できるのではないか。事例研究の継続が必要だ。

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

改訂:2/13/2018

FRAM分析

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

ステップ1：機能を識別し、記述する

task:
work-as-imagined
activity:
work-as-done
function=activity

機能の種類:
- ◆ 技術の機能
- ◆ 人の機能
- ◆ 組織の機能

機能の重要性:
- ◆ フォアグラウンド機能
- ◆ バックグラウンド機能

ステップ1:
機能を識別し、記述する

機能とは目標を達成するための手段
- ◆ 技術システムだけで実現
- ◆ 人と協調して実現

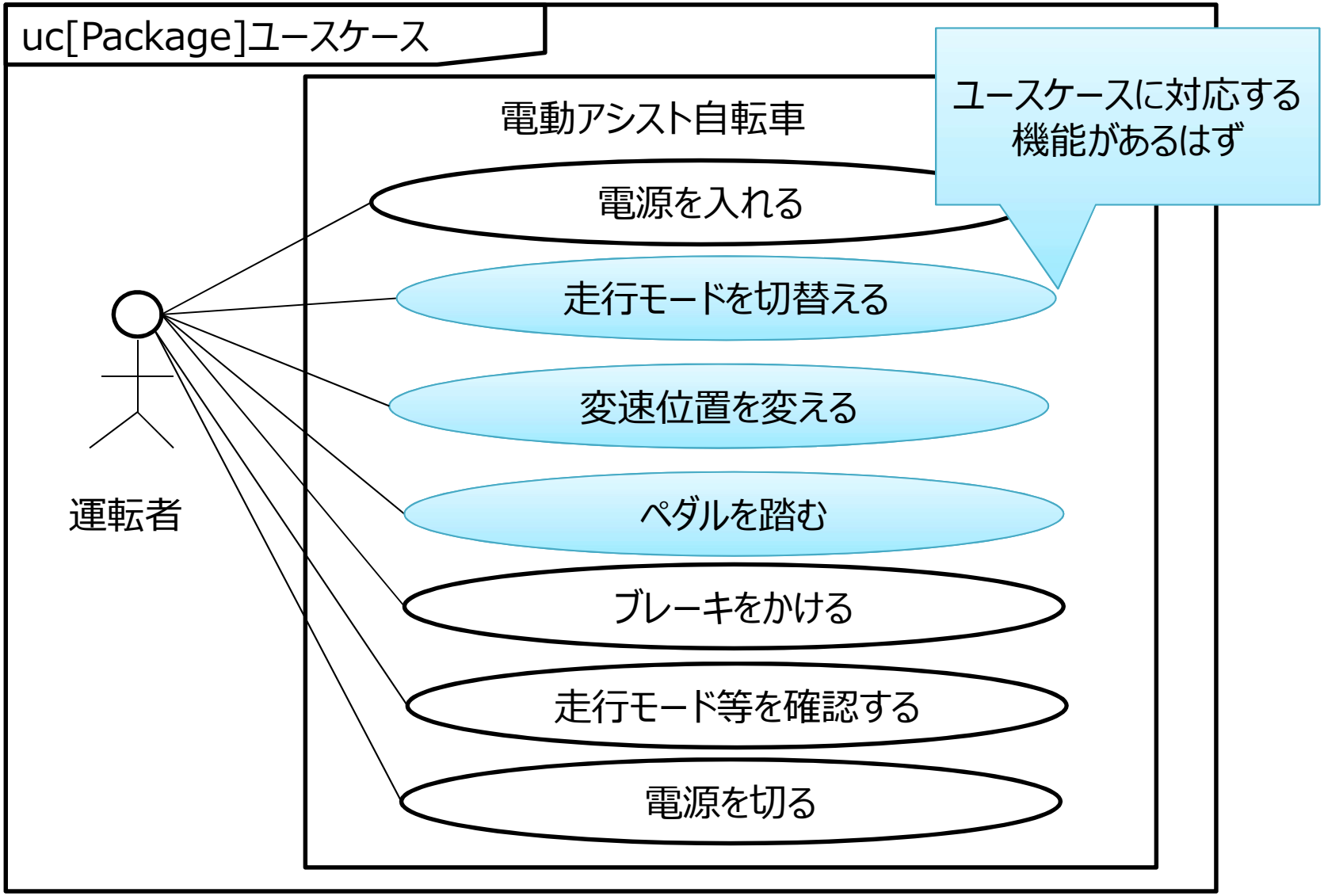
ステップ2:
想定される変動性の識別

ステップ3:
実際の変動性の集積

ステップ4:
推奨策の提案

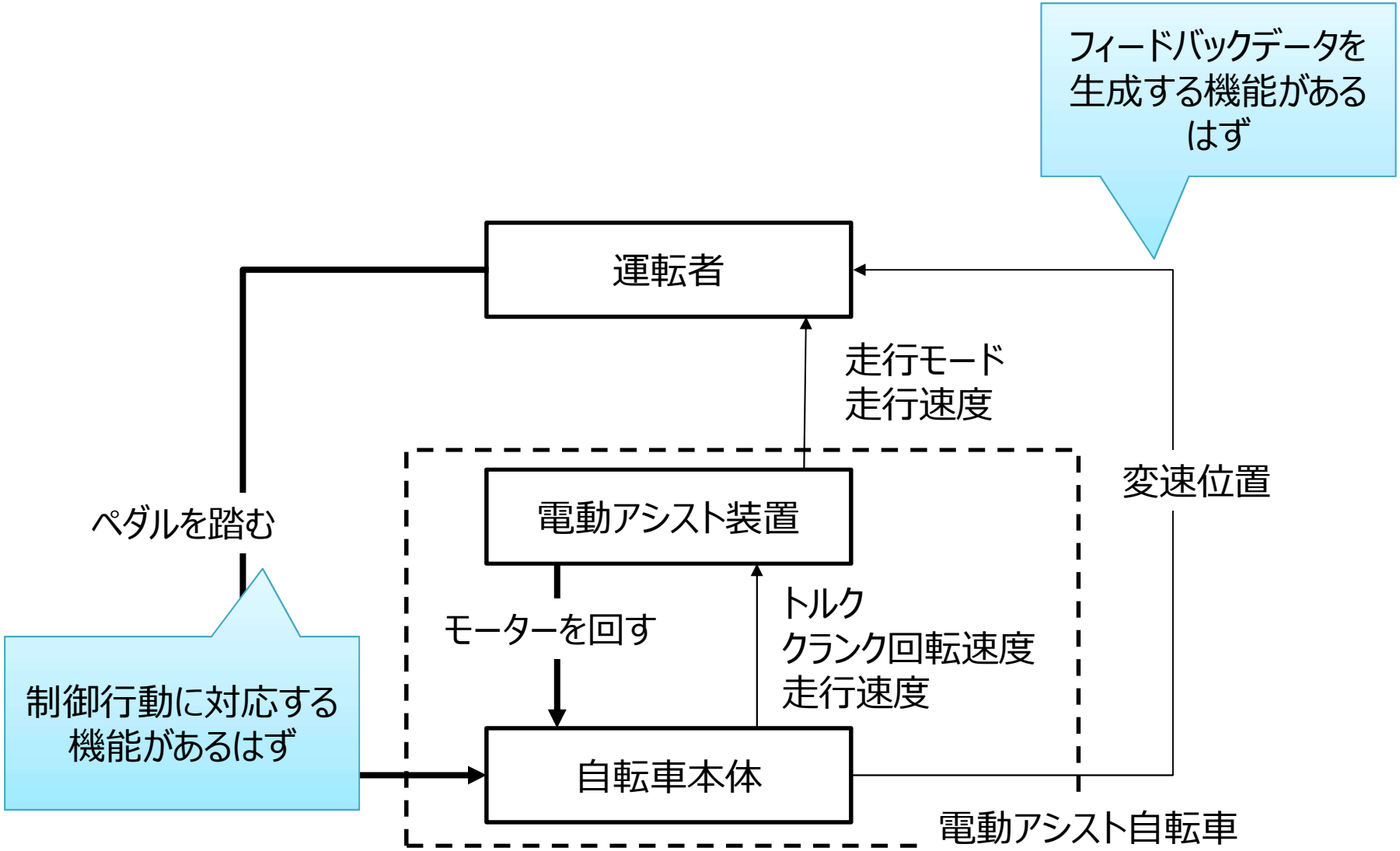
Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

ユースケース図から機能を識別する



Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

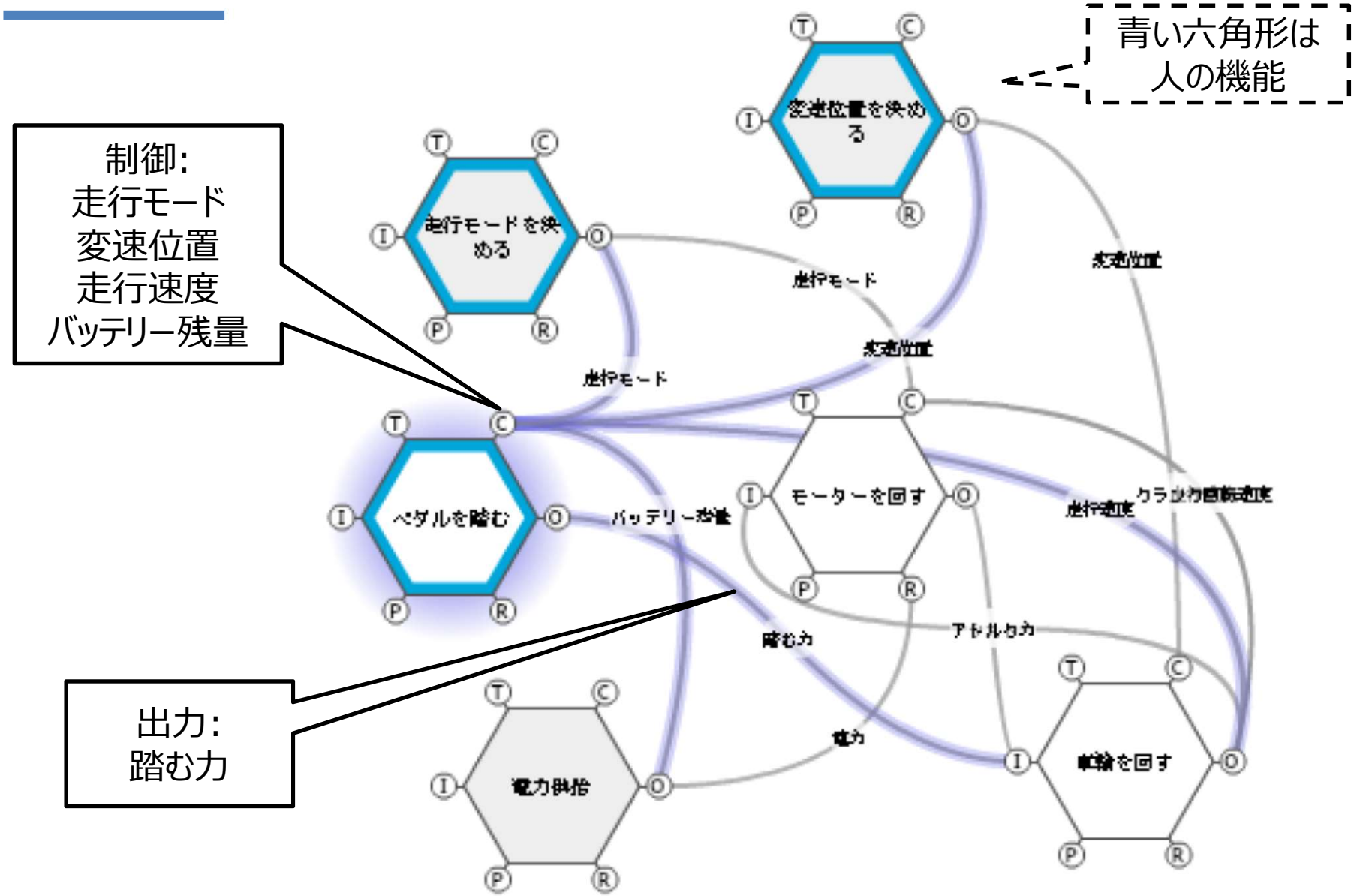
制御構造図から機能を識別する



Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

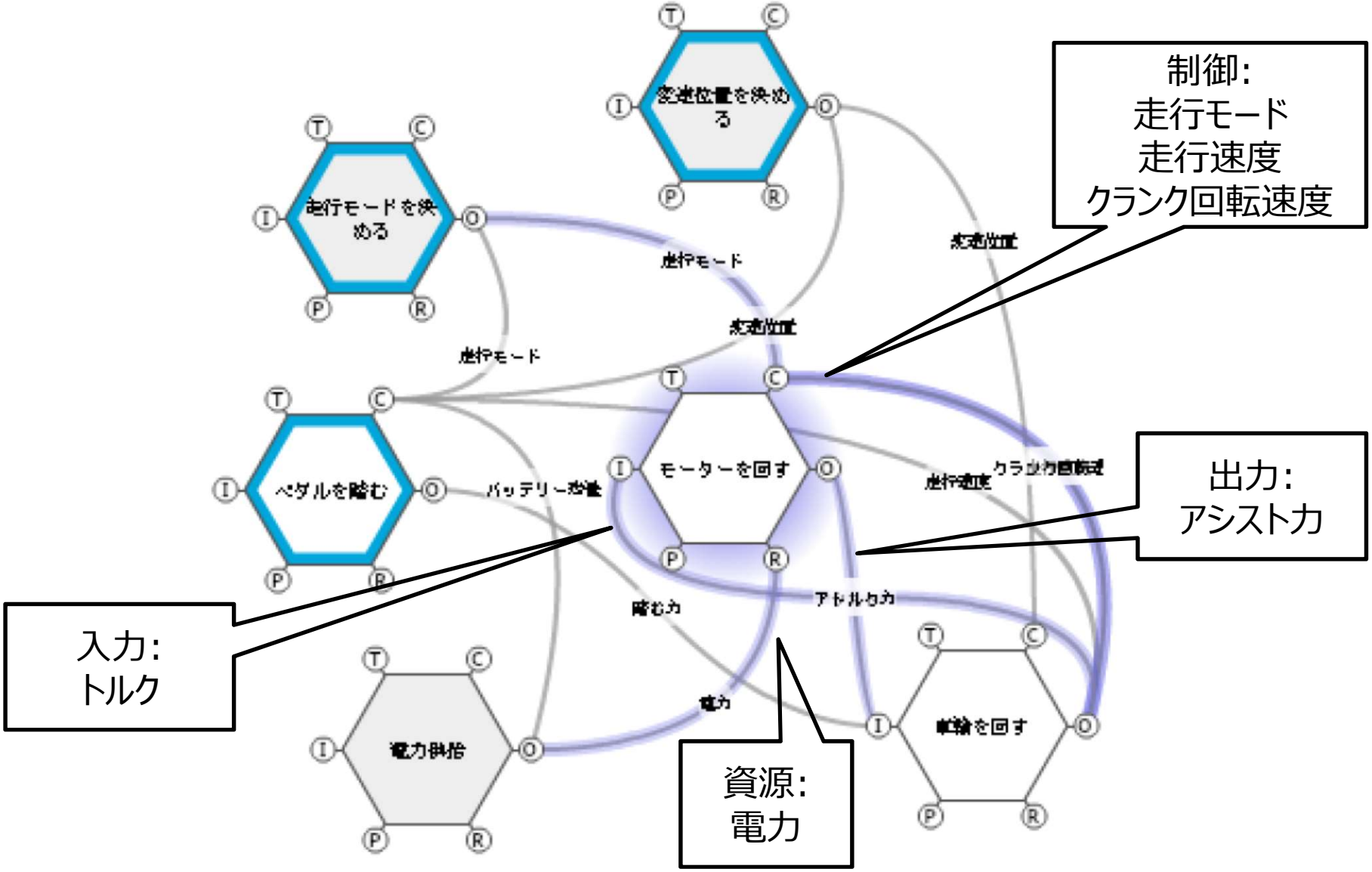
RENTACO

制御行動:ペダルを踏む



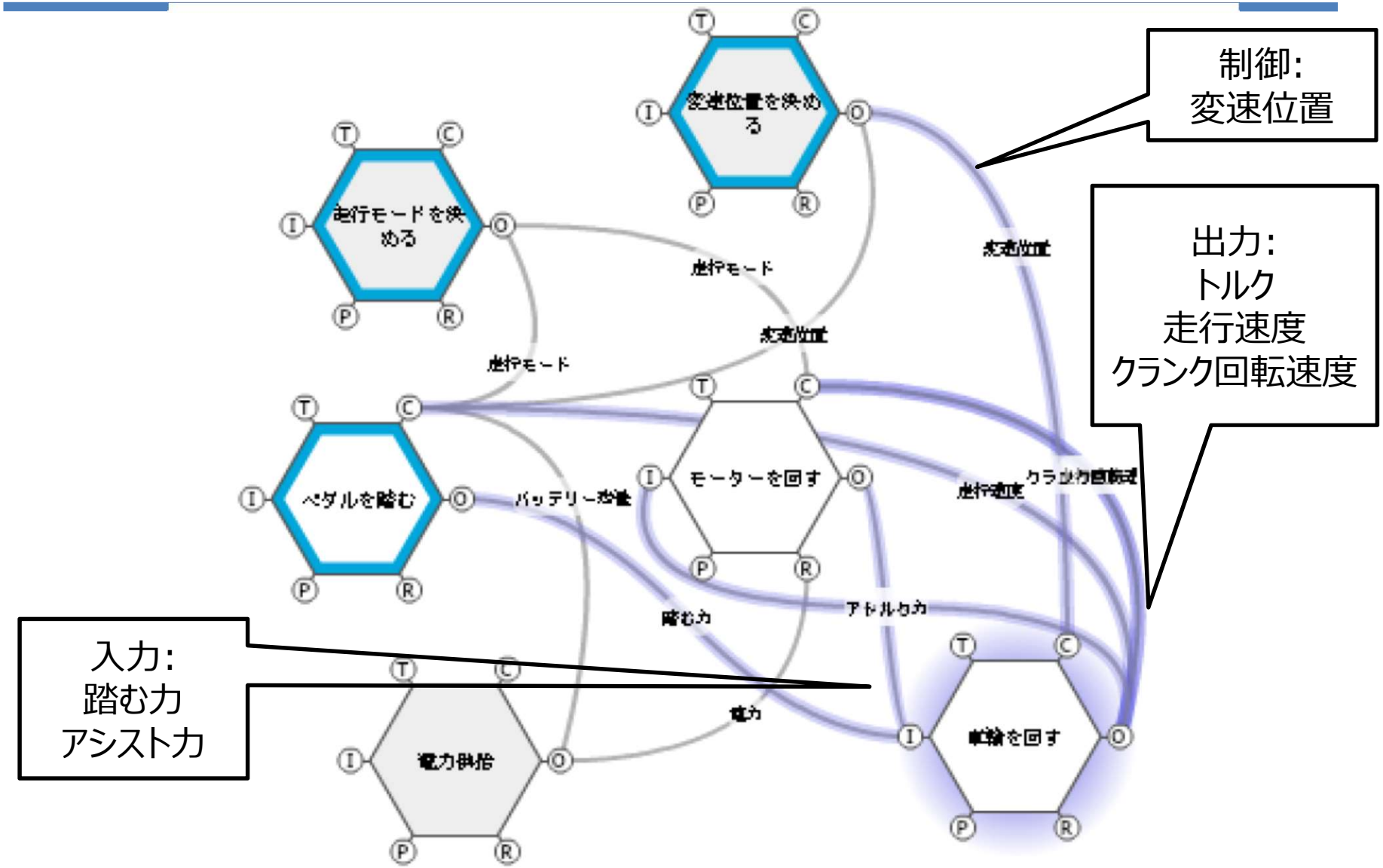
Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

制御行動:モーターを回す



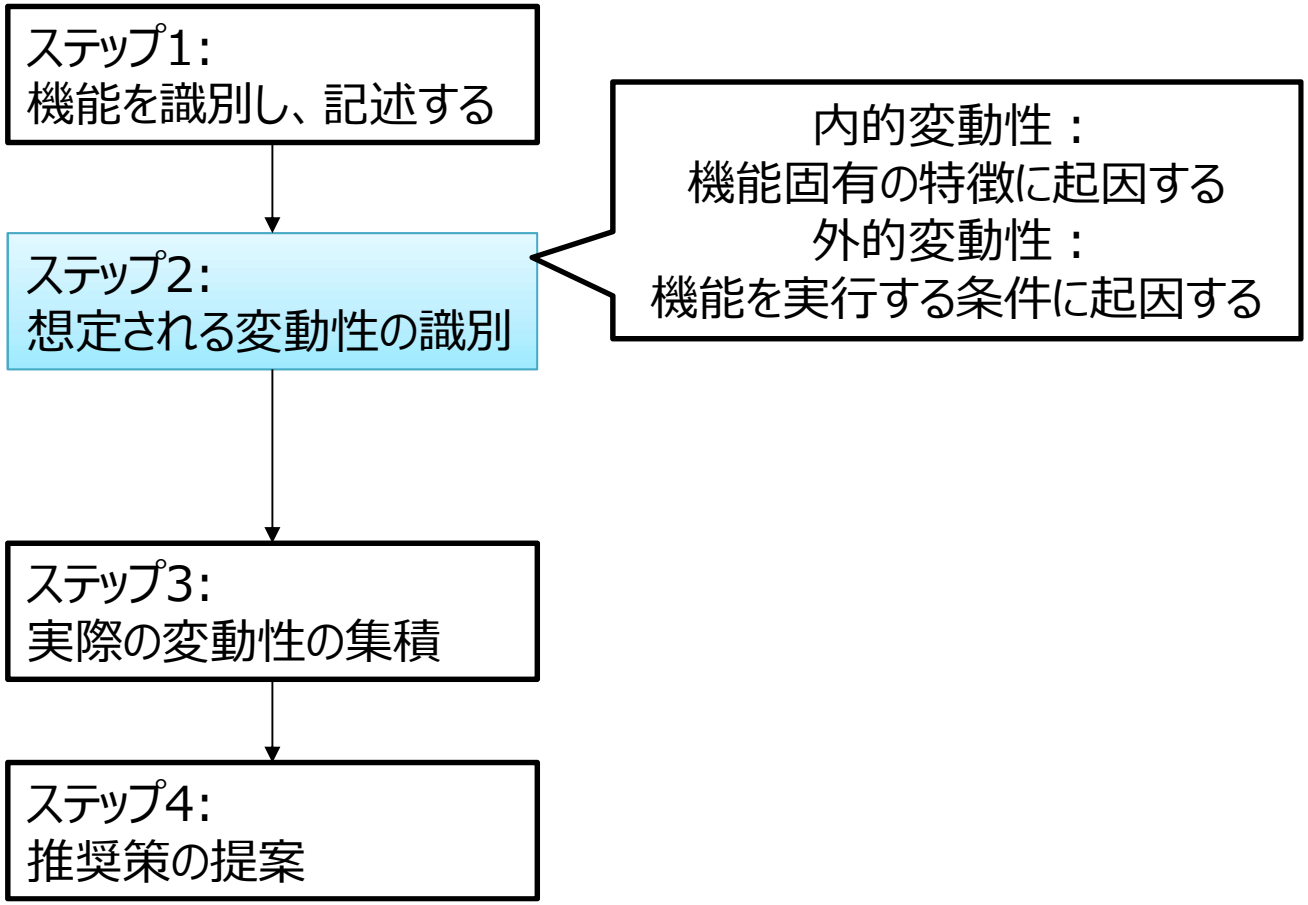
Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

フィードバック生成：車輪を回す



Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

ステップ2：想定される変動性の識別



Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

「ペダルを踏む」の出力の変動性を識別する

変動性	説明
タイミング/持続期間	早すぎる
	遅すぎる
	省略
強度/距離/方向	弱すぎる
	強すぎる
	短すぎる、近すぎる、間違った方向に
	長すぎる、遠すぎる、
間違ったもの	間違ったもの
順序	一連のものや動き、変化が、次のように変動する: ● 省略:一部が消える ● ジャンプ:一部が飛ばされる ● 繰り返し: ● 反転:前後が逆転する ● 間違った部分:間違ったものが入る

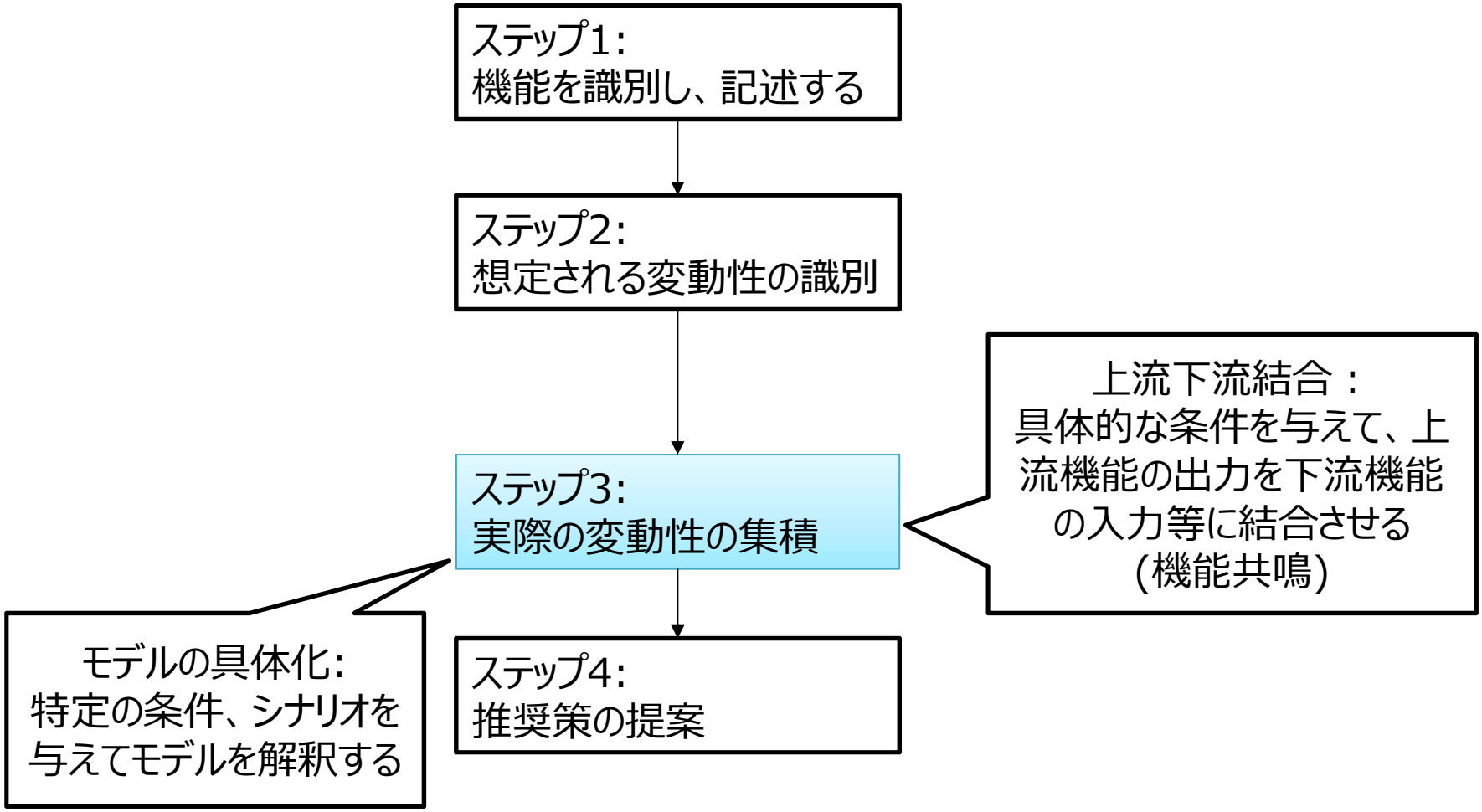
早すぎる:
バランスが取れる
前にペダルを踏む
かもしれない

遅すぎる:
自転車が走り出
してからペダルを
踏むかもしれない

強すぎる：
慌ててペダルを強
く踏む

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

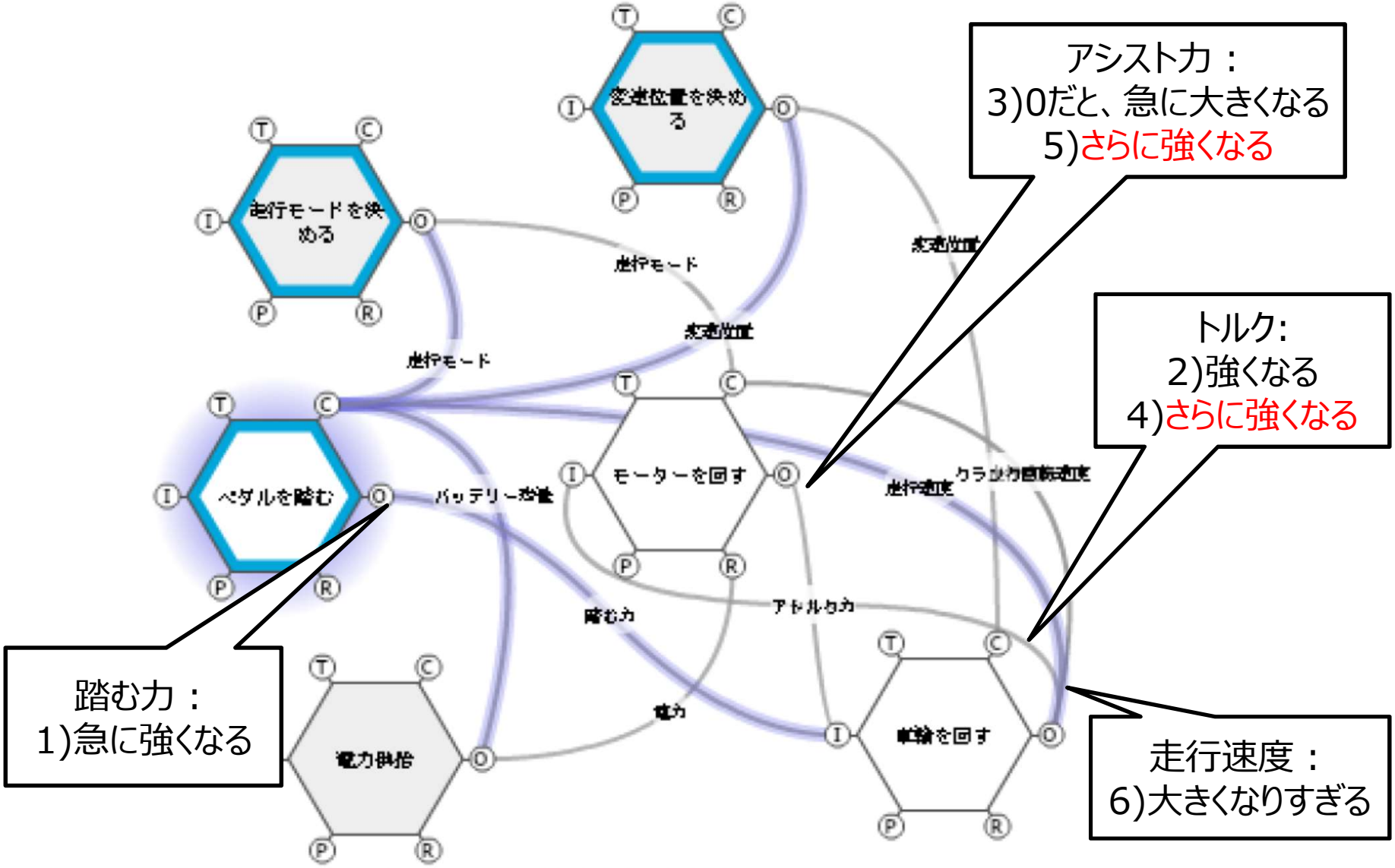
ステップ3：実際の変動性の集積



Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

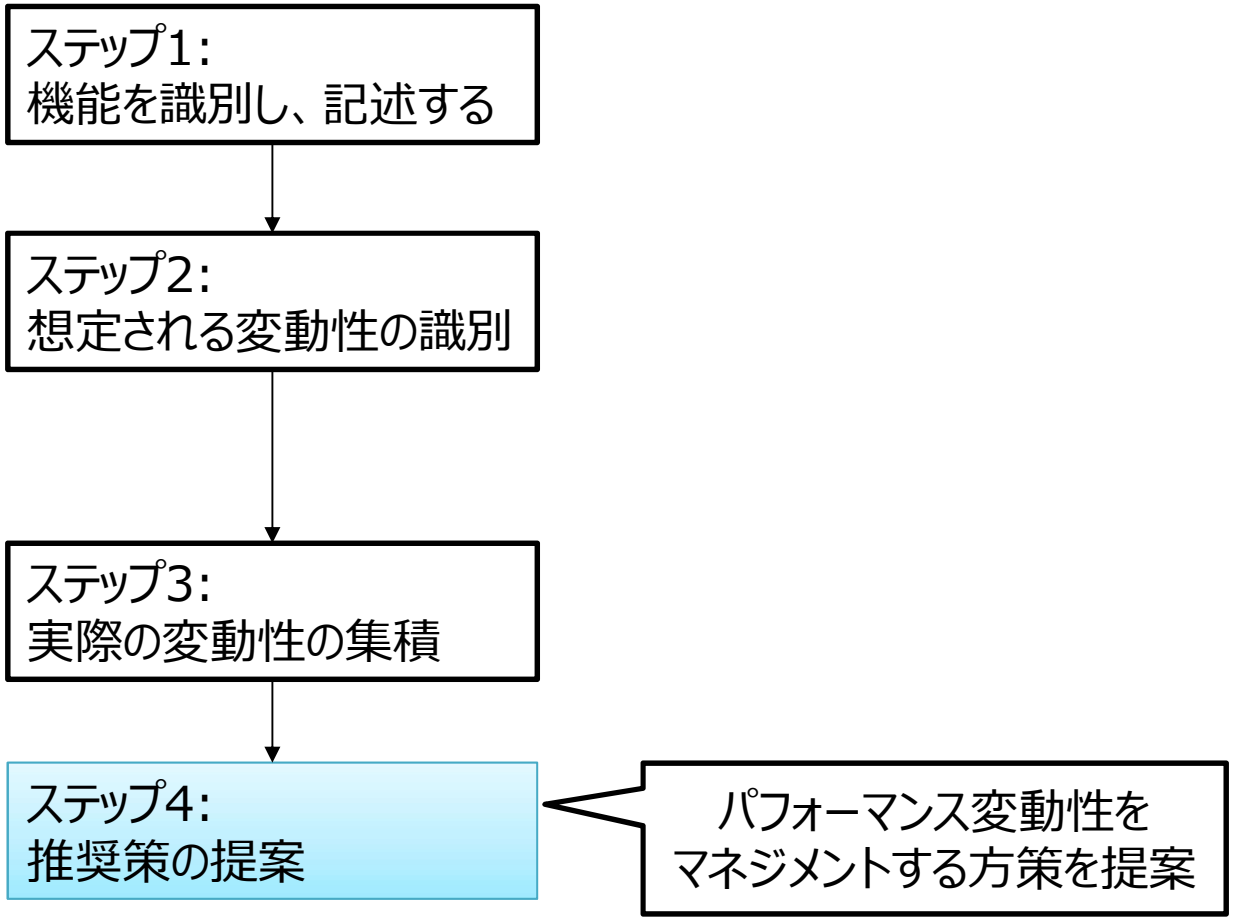
RENTACO

変動性の集積：1つのシナリオ



Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

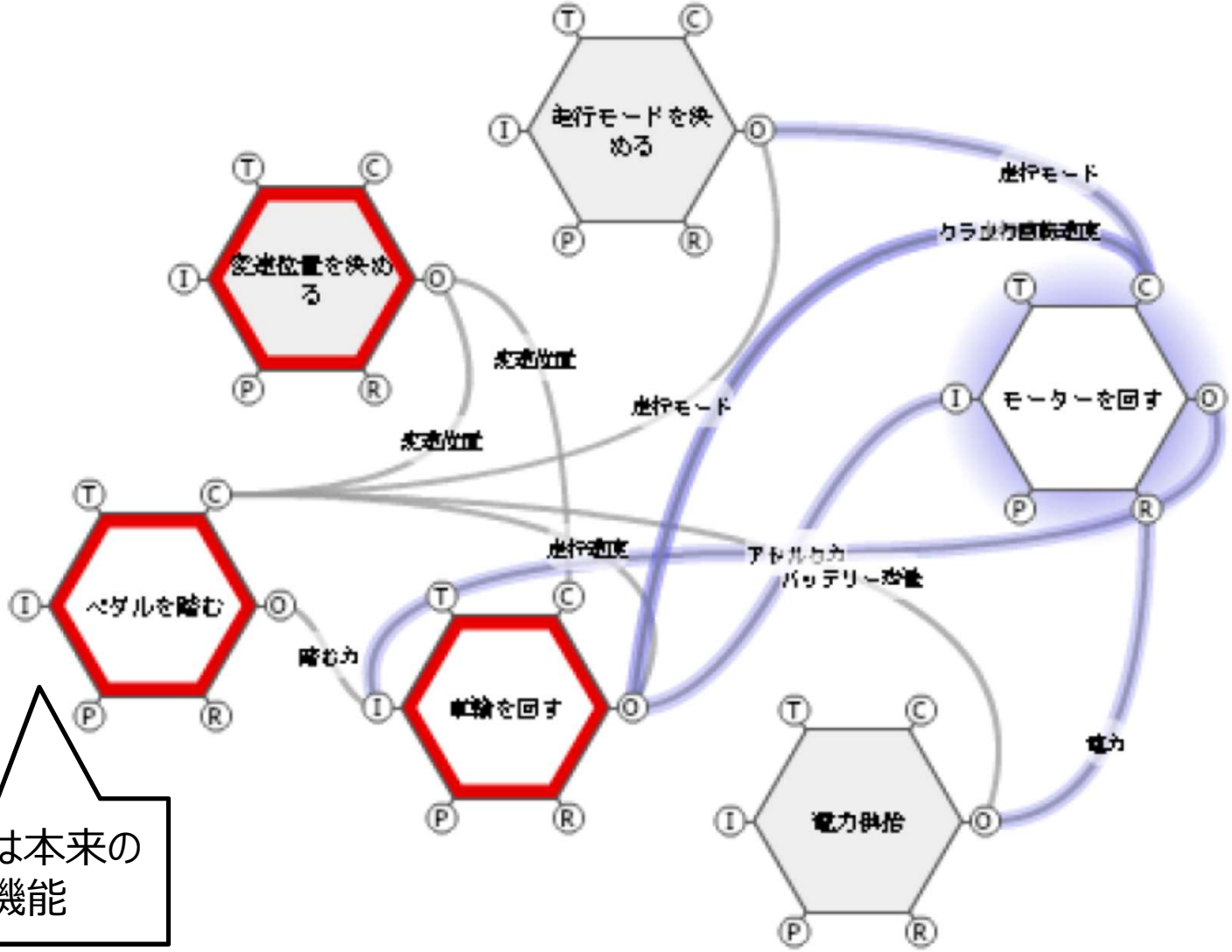
ステップ4：推奨策の提案



Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

FRAMモデルを変形させる

特徴：
ペダルを踏むと
モーターを回す
との間にリンク
がない



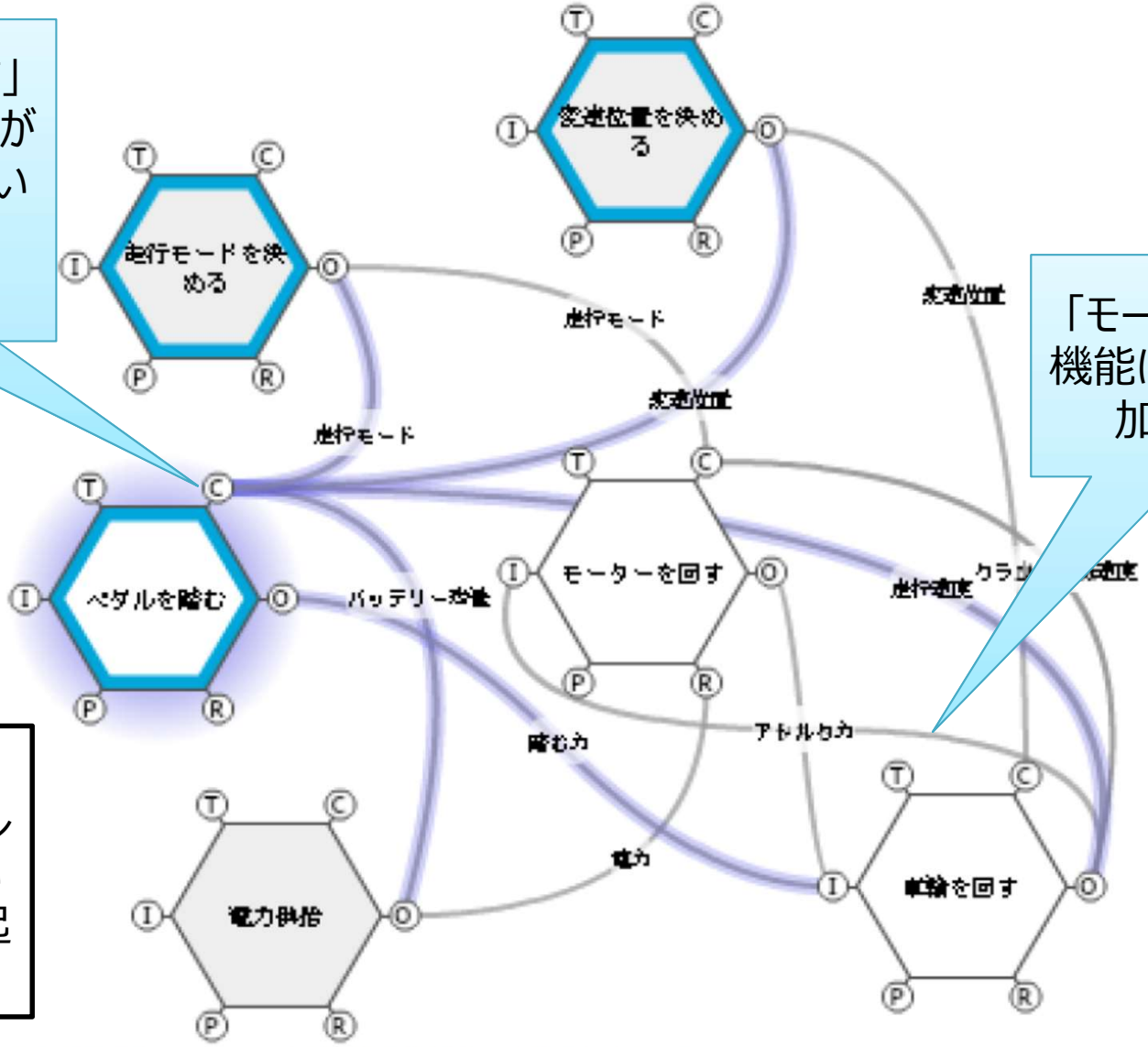
赤い六角形は本来の
自転車機能

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

RENTACO

出力の変動性をマネジメントするために

「モーターを回す」機能の出力だけがリンクされていない
=>
リンクを追加



「モーターを回す」機能によるトルクは加算しない

アシスト力が0のときにはペダルを強く踏まないように注意喚起する

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

考察

- ◆ 機能の識別
  - ■ 今回は、制御構造図を参照して機能を抽出した。
  - ■ 制御行動とフィードバックに対応する機能があるはず。
- ◆ 機能を記述する手順
  - ■ 制御行動を与える機能から始める。
  - ■ フィードバックデータを制御に対応させる。
  - ■ 機能の出力を決め、入力、事前条件、時間制約は何かを考える。
  - ■ 次に、入力等を生成する機能の記述を続ける。
- ◆ 変動性
  - ■ 出力の変動性を識別する。
  - ■ 特定のシナリオに沿って、変動性の集積を試みる。
  - ■ 今回は、FRAMモデルを観察することによって、その構造的特質に気づいて、変動性をマネジメントする方策にたどり着いた。
- ◆ STPA手法との相違
  - ■ 両手法ともほぼ同じ推奨策を導出できた。
  - ■ STPAでは制御-フィードバックの2側面だが、FRAMでは6側面から機能を記述するので、機能間の依存関係が見やすくなる。

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

参考資料

1. 2014年度成果報告書「要求の仕様化に関する課題、プロセス及び手法」、JASAホームページ
2. 「意図を記述すれば、安全性が高まる」、日経テクノロジーオンライン(2016年11月)
3. 「安全誘導型設計の特徴と試行」、ET2016　JASA技術本部セミナー
4. 「意図記述言語SSQLの狙いと特徴」、ET-WEST2017 JASA技術本部セミナー
5. 「意図・要求記述レベルのSTAMP/STPA手法」、第2回STAMPワークショップ(2017年11月)
6. 「STAMP/STPA適用事例の解説と演習」、安全性解析手法STAMP/STPAセミナー@大阪（2017年12月）

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.

問合せ先：
中村　洋
hiroshi19.nakamura@nifty.com

Copyright© 2016-2018 Hiroshi NAKAMURA. All Rights Reserved.