

# ソフトウェア開発実践演習 分散システムアーキテクチャ設計・検証[C1]

株式会社日立製作所 岡田雅江 株式会社日立製作所 木谷光博  
キヤノン株式会社 樋口史仁 株式会社インテック 佐藤駿樹

## 分散システムにおける問題点

クラウドを活用した分散システムの実現には、耐故障性のあるインタラクション設計、複製データの一貫性モデル、通信遅延によるメッセージ順序解決などの固有の難しさが存在する。  
その問題を解決するためには、システムの要求に合わせた**保証と妥協のトレードオフ**が重要となるが、適用事例が少なく、知見を蓄積することが必要である。

## 解決のためのアプローチと効果

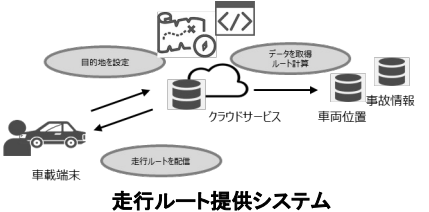
データの特性が異なる複数の分散システムに対して、クラウド参照パターンを適用した設計と、アーキテクチャ選択に対するトレードオフやリスクの検証を行った。  
これらの結果を比較することで、複数のアーキテクチャの選択肢を持つ分散システムの設計の際、**注目すべき6つの観点**と、それらに対する**アーキテクチャ選択時に考慮すべき特性**を獲得した。

## 提案プロセス

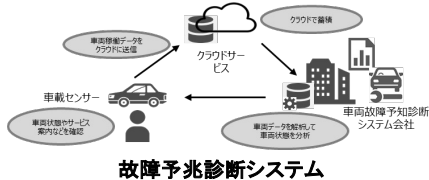
### 要件定義

ペルソナ分析, CJM

関心事:リアルタイム性

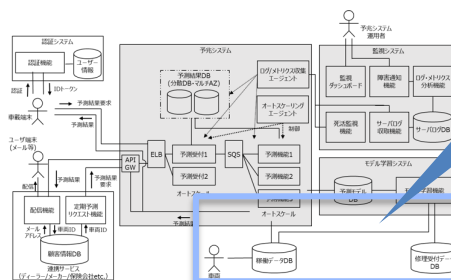
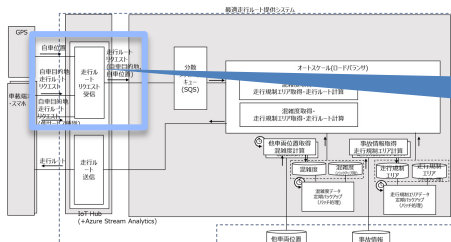


関心事:匿名性



### 設計

ADD



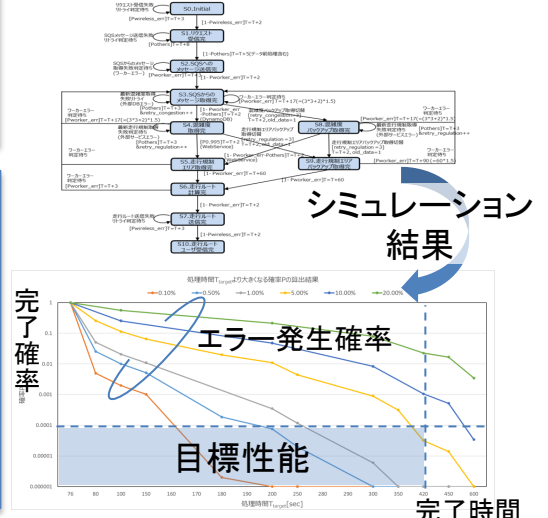
### 検証

モデル検査(PRISM)

データを直接車両から取得  
↓  
最新の車両位置情報  
データ量:小

データを外部DBから  
間接的に取得  
↓  
長期間の時系列データ  
データ量:大  
ただし、不要な情報  
(特に秘匿情報)の混入  
に注意

処理の遷移条件・確率モデル



特性の違う複数のシステム  
要件・優先度を定義

クラウド参照パターンを利用  
段階的にアーキテクチャを設計

アーキテクチャ選択を検証  
注意事項を抽出

エラー発生確率に応じた  
処理完了時間とその確率を検査

## 考察・成果

検証結果の  
比較・考察

6つの設計観点と

アーキテクチャ選択のためのシステム/アプリ/データ特性獲得  
⇒要件定義, アーキ設計・検証時の注意項目としてリスト化

<設計観点>

- ①DB選択
- ②エッジとのデータ送受信
- ③データ管理場所
- ④フロントとバックエンド結合
- ⑤サーバの拡張・縮小
- ⑥監視機能

(例)

考慮すべき特性

アーキテクチャ	エッジとのデータのやり取り			
	データの粒度	データ量	通信負荷	他情報混入の恐れ
最新エッジから取得	最新データを利用可能	小規模に向いている (データ量大の場合エッジ側に保存できるか、通信料について要検討)	高い	無
一度外部のDBに読み込んだものを取得	古いデータである可能性あり	大規模であっても可	低い	有。必要以上のデータ(特に個人と紐づく情報)が混入しないように気を付ける

アーキテクチャ選択

## 今後の課題

■扱ったシステム数が限定的であり、未検討の**特性**が存在する可能性

他ドメインのシステムに対しても  
提案プロセスを実施し  
多様な知見を蓄積していく必要有

■業務での実適用に向けて

具体化した検討が必要

例. システム性能、クラウド環境の仕様、  
データ項目や閲覧権限の詳細化