



SWEST15 組込みシステム技術に関するサマールワークショップ2014

クルマ同士が協調する安全運転支援の技術動向

佐藤 健哉

同志社大学 モビリティ研究センター



講演内容

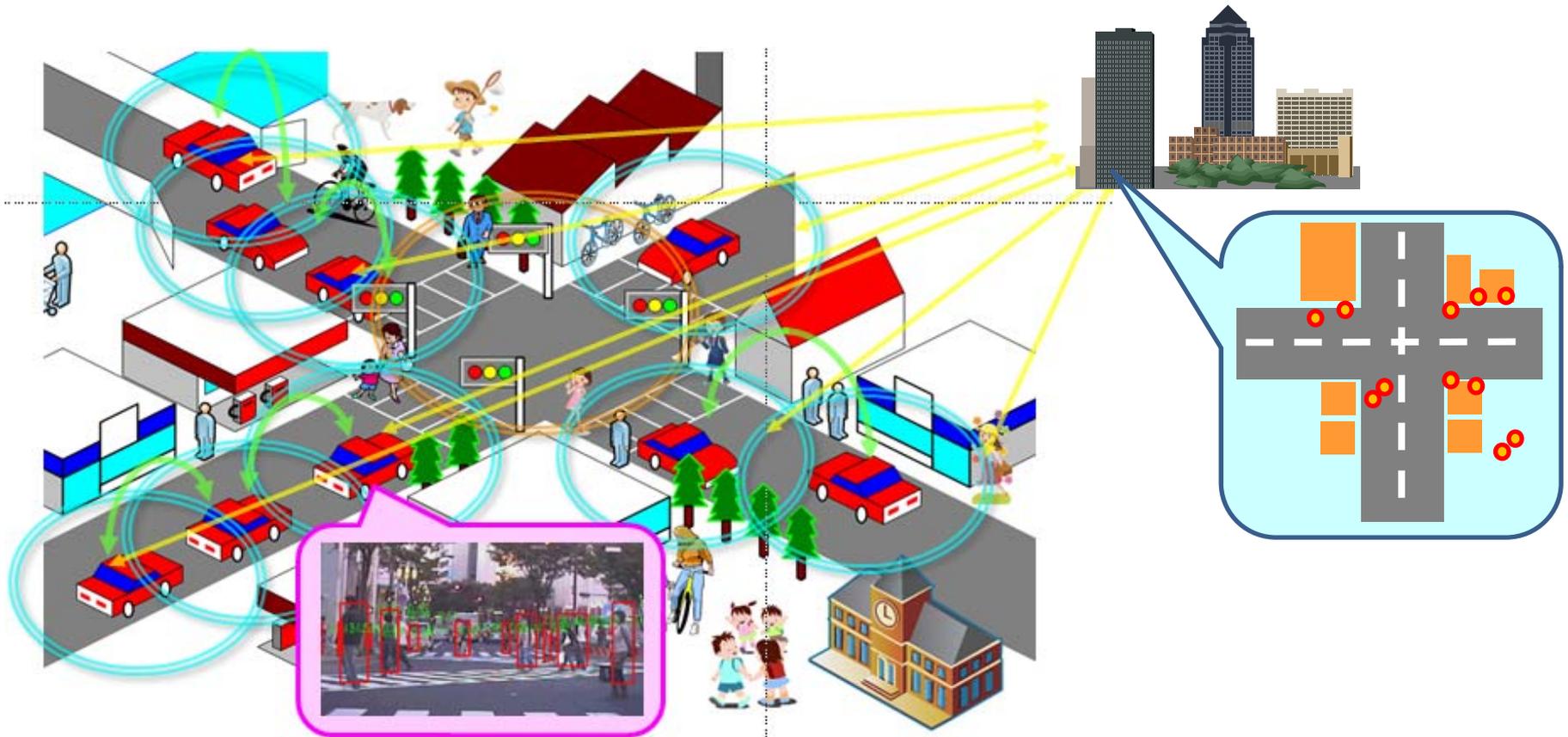
- 概要
 - 車両にカメラやレーダを搭載し周辺環境を認識することで安全運転支援や自動運転を目指したシステムが開発されている。しかし、見通しの悪い道路での飛出しなど、車載センサだけでは検知不能な状況も存在する。本講演では、現在のシステムの紹介に加え、複数車両や道路インフラ間で情報交換を行う協調型先進運転支援システムにおけるセンサデータの効率的処理の研究について、車々間・路車間通信の国際標準化動向も含めて解説する。
- 項目
 - 先進運転支援システム(ADAS)の現状と動向
 - 安全運転支援システム
 - 協調ITS
 - 自動運転
 - 研究内容
 - 車載データ共有
 - 効率的情報処理

協調型自動運転を目指して

通信を利用した先進運転支援システムにおけるデータ処理の効率化

協調型自動運転，協調ITS（C-ITS）

- カメラ，レーダなどの車載センサのみでは，死角にある対象物や遠方の状況を認識できない
- 各車両，道路側インフラが取得した情報を相互に交換し，安全運転支援や自動運転に活かすことが必要

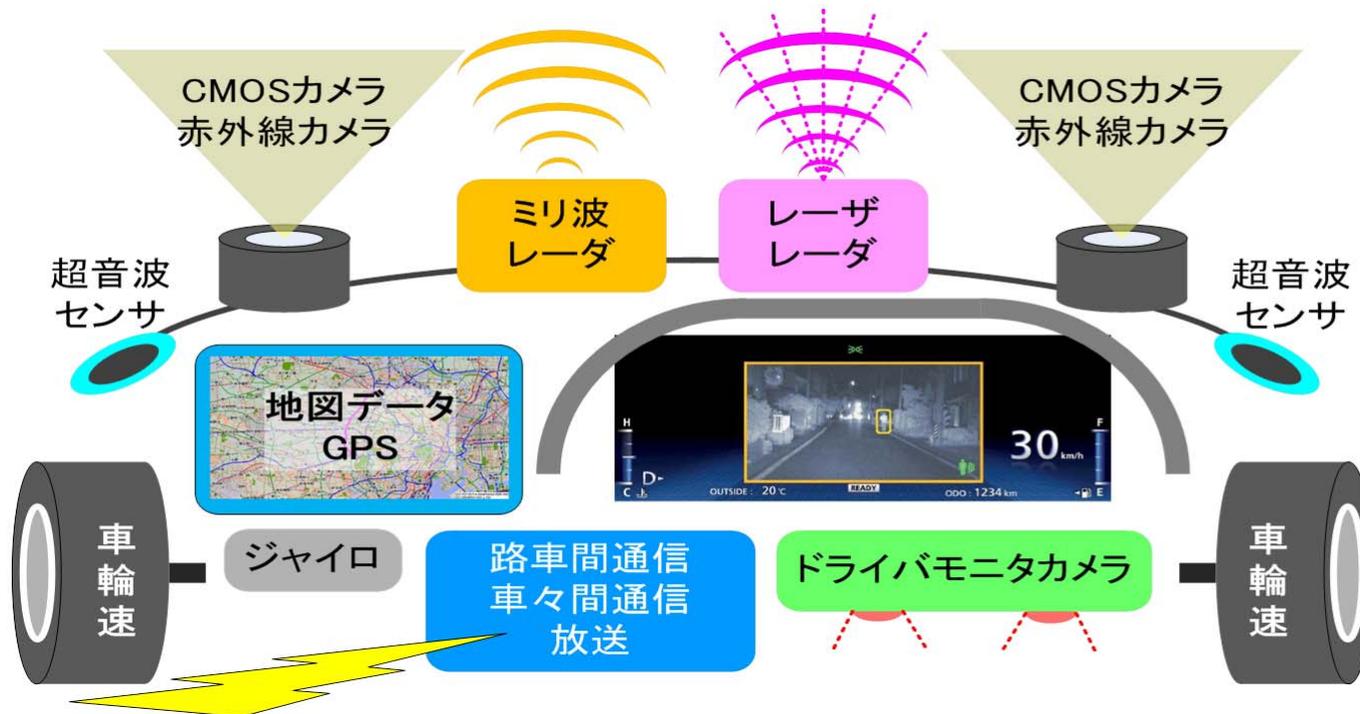


データ処理効率化

(車載データ統合アーキテクチャ)

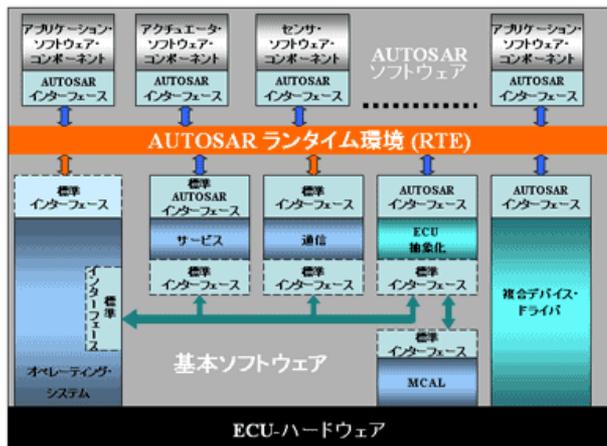
車載センサ情報

- 車両走行時の安全性向上のために様々なセンサ類が搭載
 - カメラ, レーダ, 車々間・路車間通信など
- データの種類や量の増加とともに, 管理・開発が複雑化
 - 周辺車両, 歩行者, 車線, 路面状況, 交差点信号

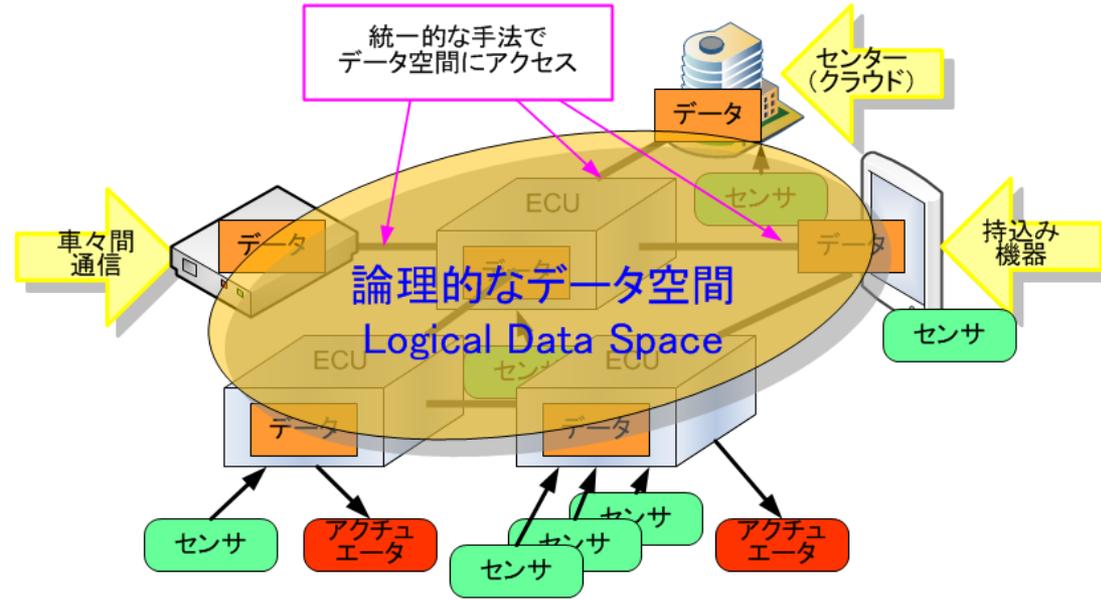


アプローチ

- 機能連携 (AUTOSAR) ⇒ データ統合 (CLOUDIA)
 - 車両の走行環境を特定の論理的なデータ空間内に表現
 - 論理的データ空間に対しての統一したアクセスで容易に情報取得



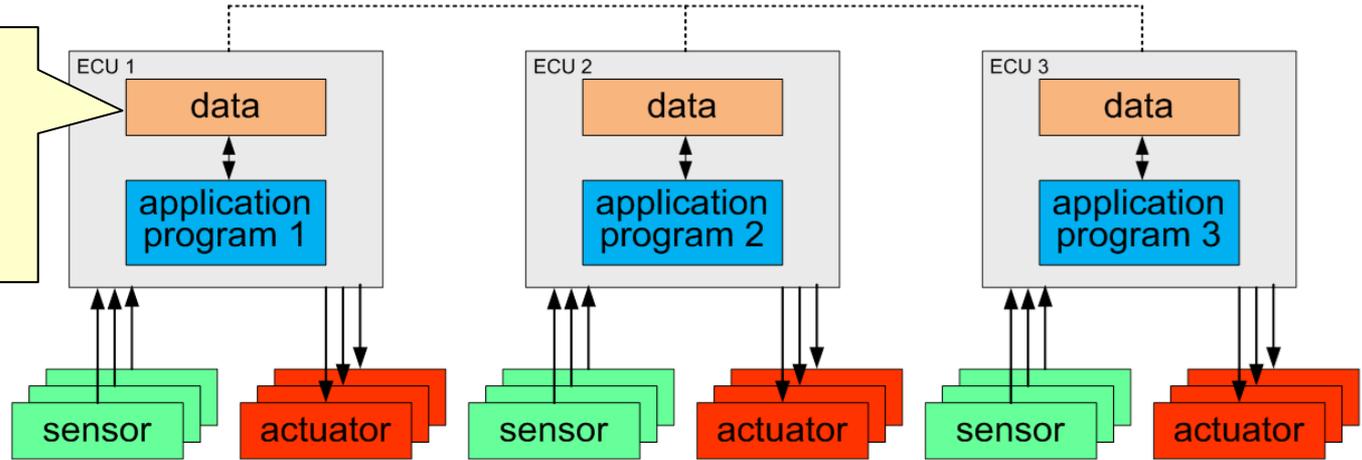
AUTOSARによる機能連携



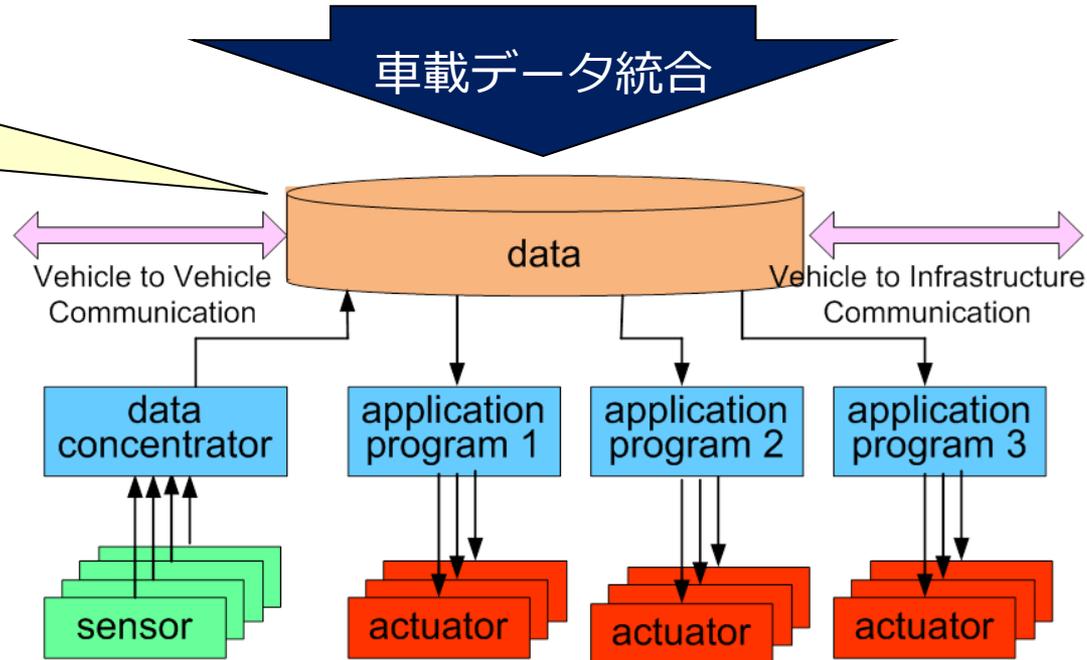
車載データ統合アーキテクチャ

車載データ統合アーキテクチャ (イメージ図)

ECUごとにセンサ・アクチュエータを配備し、アプリケーションごとにデータ独立して管理した設計



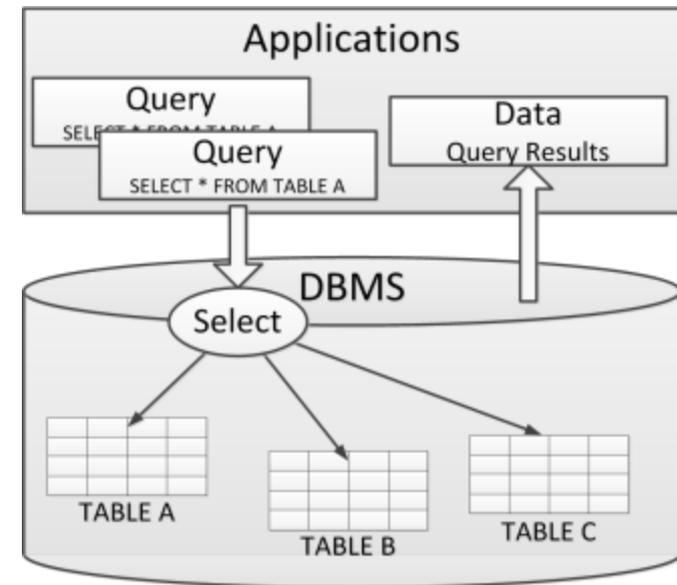
各ECUに分散されているデータを統合管理し、共通アクセスするアプリケーション設計



- センサ追加・高性能化
 - 複数センサデータ融合
 - 車外データ利用
 - 処理の統合/分散
 - 新規アプリ独立開発
- これらが容易に実施可能

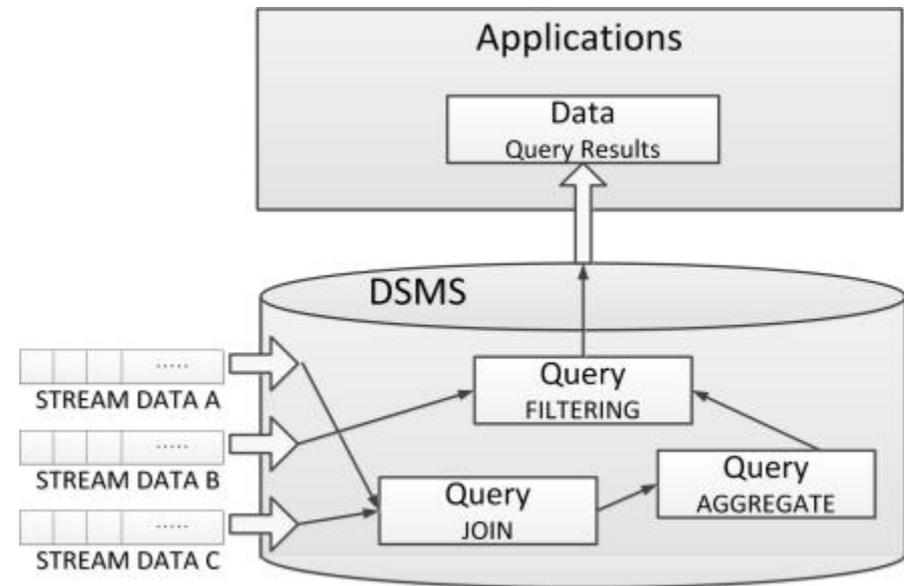
リレーショナルデータベース

- 通常のデータベース管理システムDBMS (Data Base Management System)を利用した場合
 - SQLを利用したデータベース
- 問題点
 - データベース管理システム (DBMS) では, 予約情報などデータの一貫した管理が重要
 - 蓄積したデータに対してクエリを発行し, 一貫性を確保した応答を行うため, データアクセスが遅い



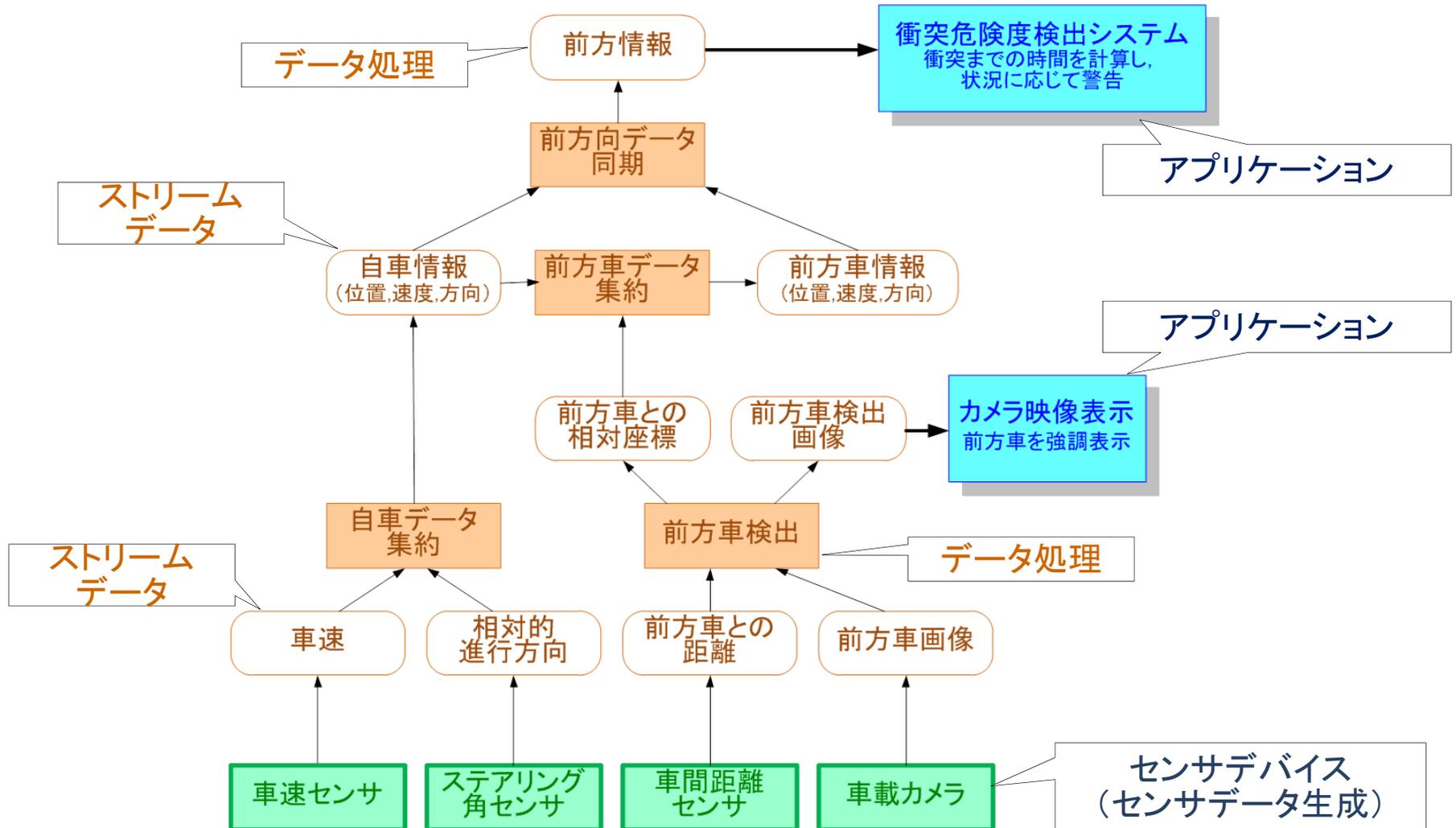
ストリームデータ処理

- ストリーム処理技術DSMS (Data Stream Management System)の登場
 - 株価や外国為替レートの情報, ネットワークのパケットなど, 時系列に生成されるストリームデータを処理
 - スタンフォード大学やMITなどがDSMSの研究用プラットフォームを開発
- データの抽出条件や集約関数などを事前にアプリケーションが登録しておき, ストリームデータの中から出力すべきデータ出現時の結果を取得
- ストリームデータに対して演算やクエリを提供し, 一貫性を確保しないため高速処理が可能



DSMSによるシステム構築 (1)

- データストリーム管理システム (DSMS) を利用したデータ統合の応用システム



DSMSによるシステム構築 (2)

- DSMS (MIT開発のBorealis) を利用し, Linux上で実装
 - 同スキーマを持つ時系列データとして処理
 - データ処理ボックスによるデータ変換の機能の実現
 - アプリケーションからのクエリによるデータ提供

DSMSによるストリーム演算プロセス

The screenshot displays a complex stream processing environment. At the top, several terminal windows show logs for 'Borealis@127.0.0.1' and 'precrashsafety.cc', detailing data flow and processing steps. A central window shows a camera feed with a green bounding box around a vehicle. At the bottom, more terminal windows show successful deployment of components like 'speed.cc', 'interfering.cc', 'elder.cc', and 'camera.cc'. The overall layout is organized into three main functional areas: stream processing, application processing, and sensor data generation.

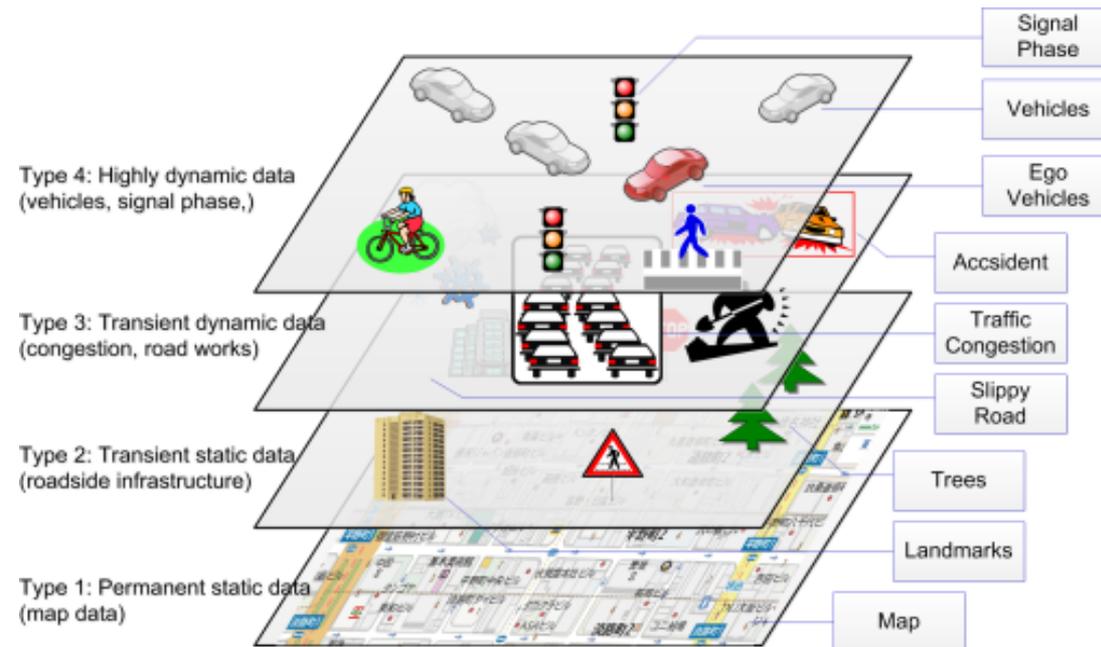
アプリケーションプロセス

センサデータ生成プロセス

地図データとセンサデータの融合 (ストリームLDM パート1)

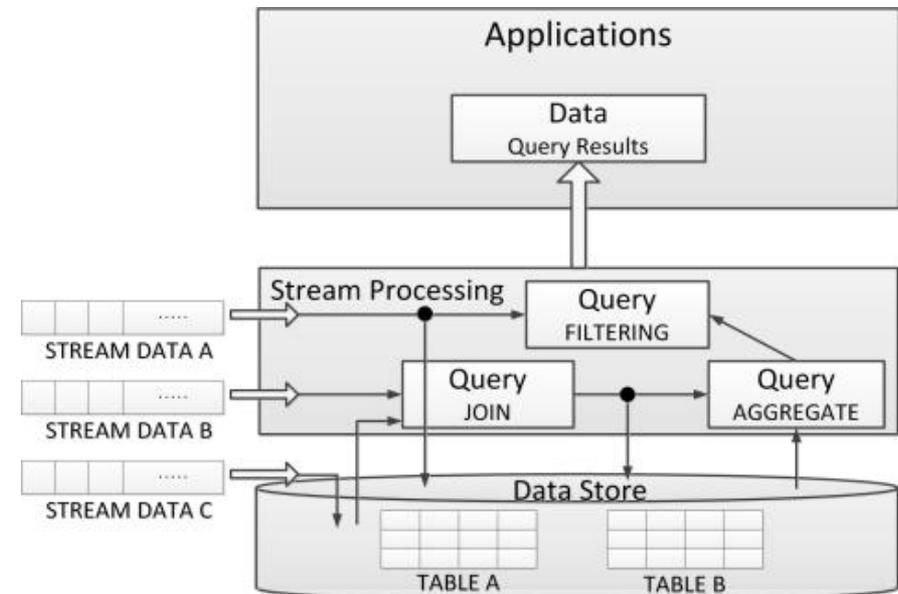
Local Dynamic Map

- 地図データ標準 LDM (Local Dynamic Map)
 - 地理的情報, 周辺車両・道路状態・交通状況・天気など
 - 更新頻度に応じて4階層で管理・保持している概念的なデータ集合体
 - アプリケーションからの問合せ(クエリ)で, 追突防止支援, 緊急車両接近警告, 信号情報提供, 歩行者・自転車存在情報を提供
- 車々間・路車間通信を利用した安全運転支援システム(協調ITS)で利用
- 欧米が協調して国際標準化を目指しているキーテクノロジー



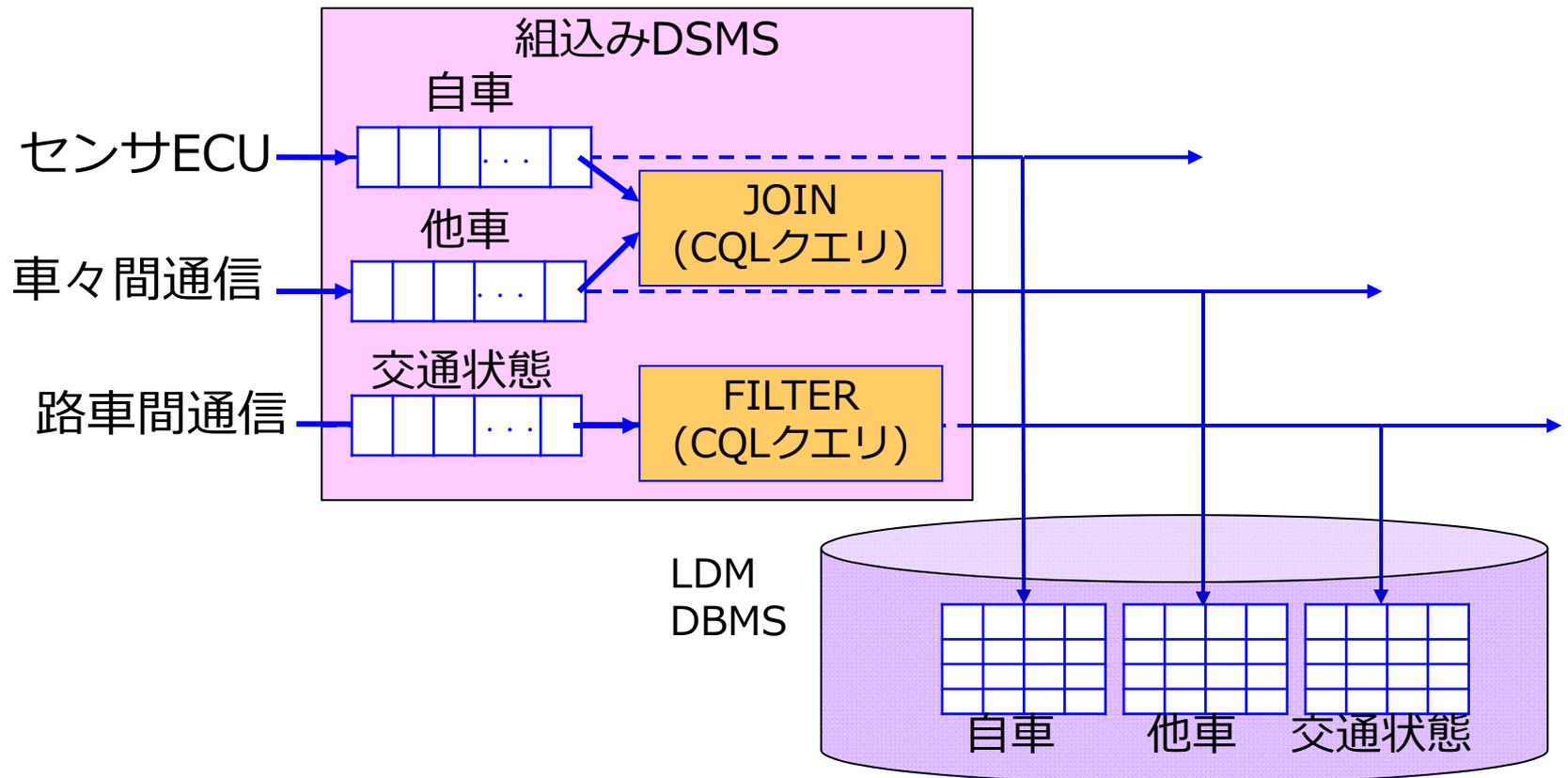
ストリームLDM (DSMSとDBMSの融合)

- データベース管理システムとデータストリーム管理システムの融合
- カメラや車載センサのデータを高速に処理可能な組込み向けストリーム処理管理システム
 - 高速ストリーム処理実現ため、地図データなどDBMSの静的情報のストリーム化機能を追加
 - 組込みシステム向け静的生成によるストリーム処理システムのランタイムを開発
- 周辺車両増加に伴う処理負荷増大に対応するため、優先度制御機構を導入
- 単一ノードだけでなく、センタ併用などのクラウドにも対応可能



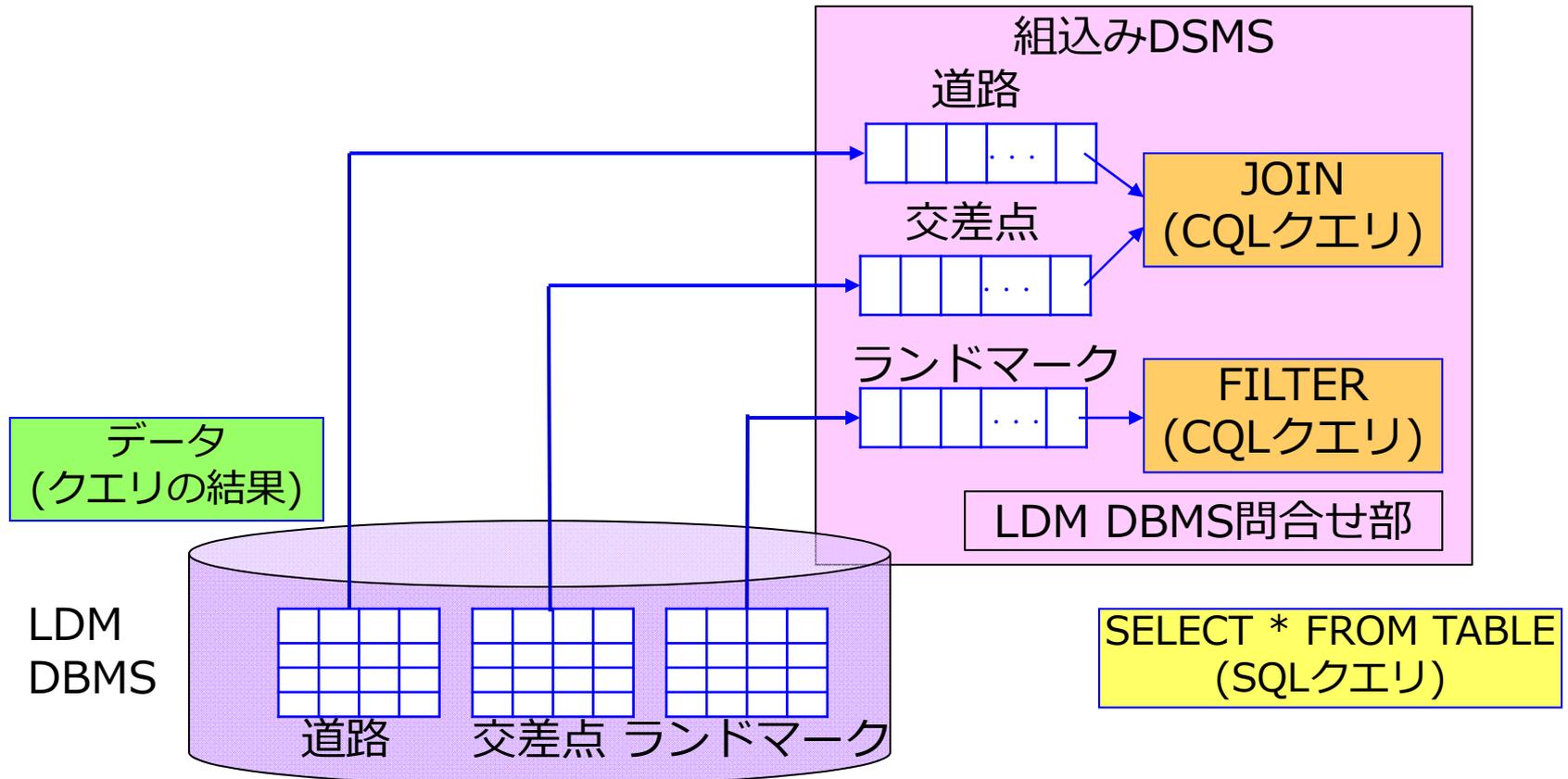
動的情報のストリーム化

- 自転車, 他車, 交通状態などの動的な情報のストリーム化
 - センサや通信からデータをDBMSのテーブルへの挿入前に, ストリームに入力
 - テーブルに挿入するデータはデータレートによってフィルタリング



静的情報のストリーム化

- 道路, 交差点, ランドマークなどの静的な情報のストリーム化
 - DBMSのテーブルに定期的に問合せ, 自転車周辺の情報を取得しストリームに入力
 - 周辺情報の取得にあたって, 自転車の走行方向, 速度などを考慮



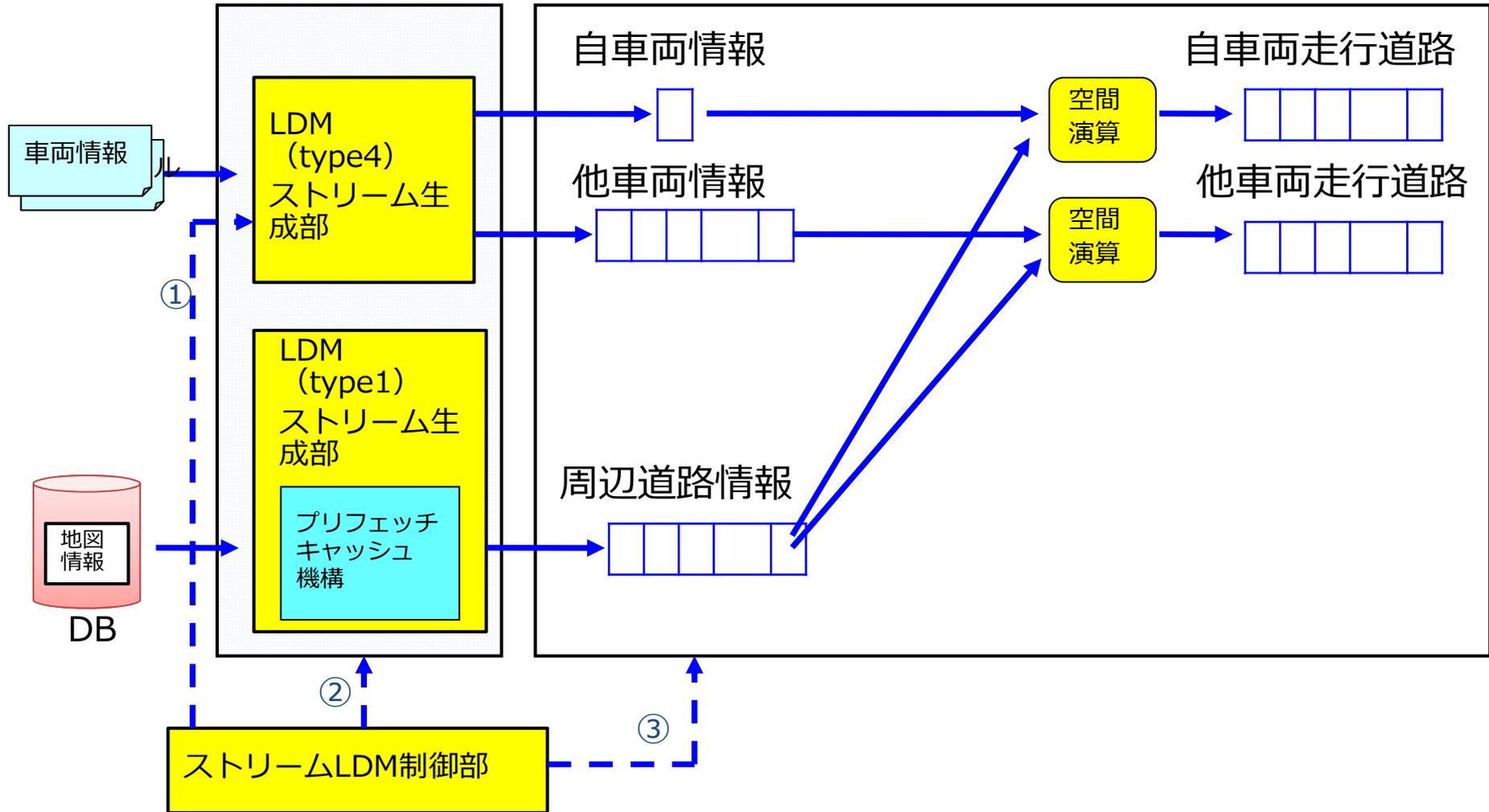
地図データストリーム化

- 目的
 - 地図情報キャッシュの更新コスト
 - 空間演算の実行コスト
 - 従来方式（RDB方式）による性能との比較
- 評価クエリ
 - 車両情報と周辺道路情報を入力し空間演算（位置マッチング）により車両走行情報を生成するクエリ
- 評価データ（既存データを使用）
 - 地図データ（RDB）、走行データ（CSV）
- 評価項目
 - レイテンシ、スループット、メモリ使用量
- 評価環境
 - PostgreSQL 8.4.8（CPU: Core2 Duo 3.0GHz / OS: Linux 2.6）

ストリームLDM構成アウトライン

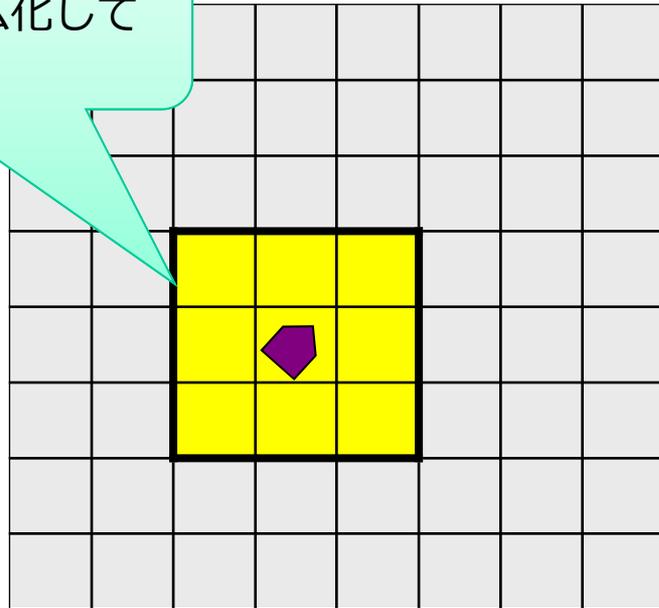
LDMストリーム化機構

クエリ実行部



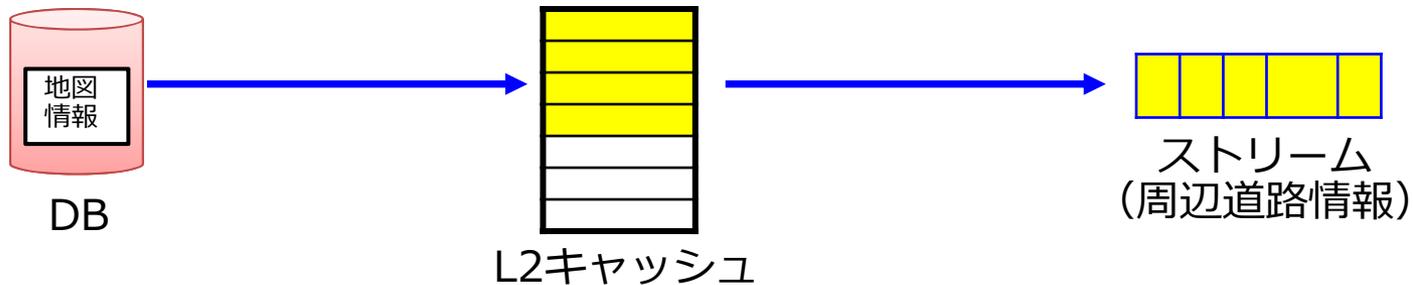
プリフェッチキャッシュ機構（初期化時）

自車両の周囲一定範囲を周辺道路としてストリーム化して出力する



周辺道路の範囲はメッシュ単位 (n×n) で設定できる (本例では3×3メッシュ)

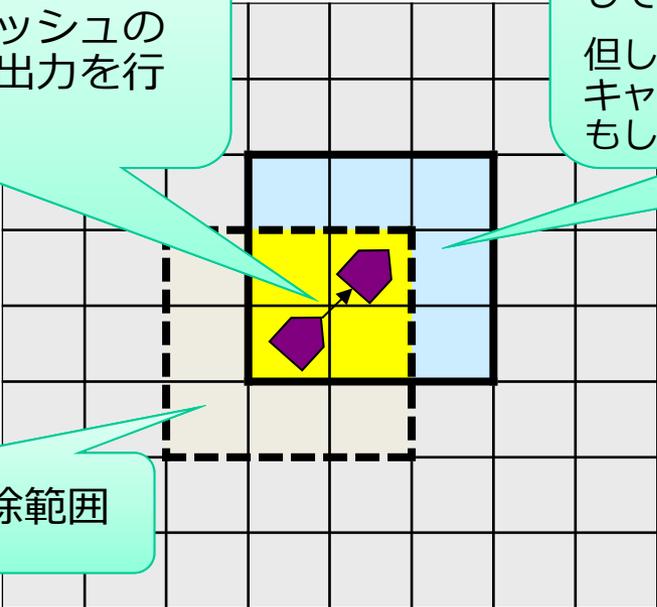
メッシュサイズもパラメータにより任意の値に設定できる



地図情報のキャッシュとストリーム化（更新時）

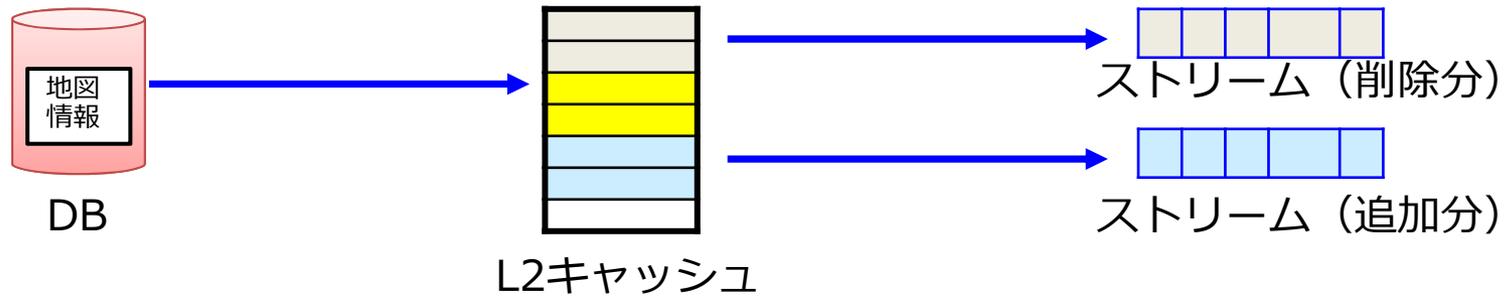
自車両メッシュ位置が変化したタイミングでキャッシュの更新及びストリーム出力を行う

キャッシュ追加範囲
 この範囲の道路情報を新たにDBより取得してキャッシュに追加する
 但し、道路がメッシュに跨っており既にキャッシュに入っている道路に対しては何もしない

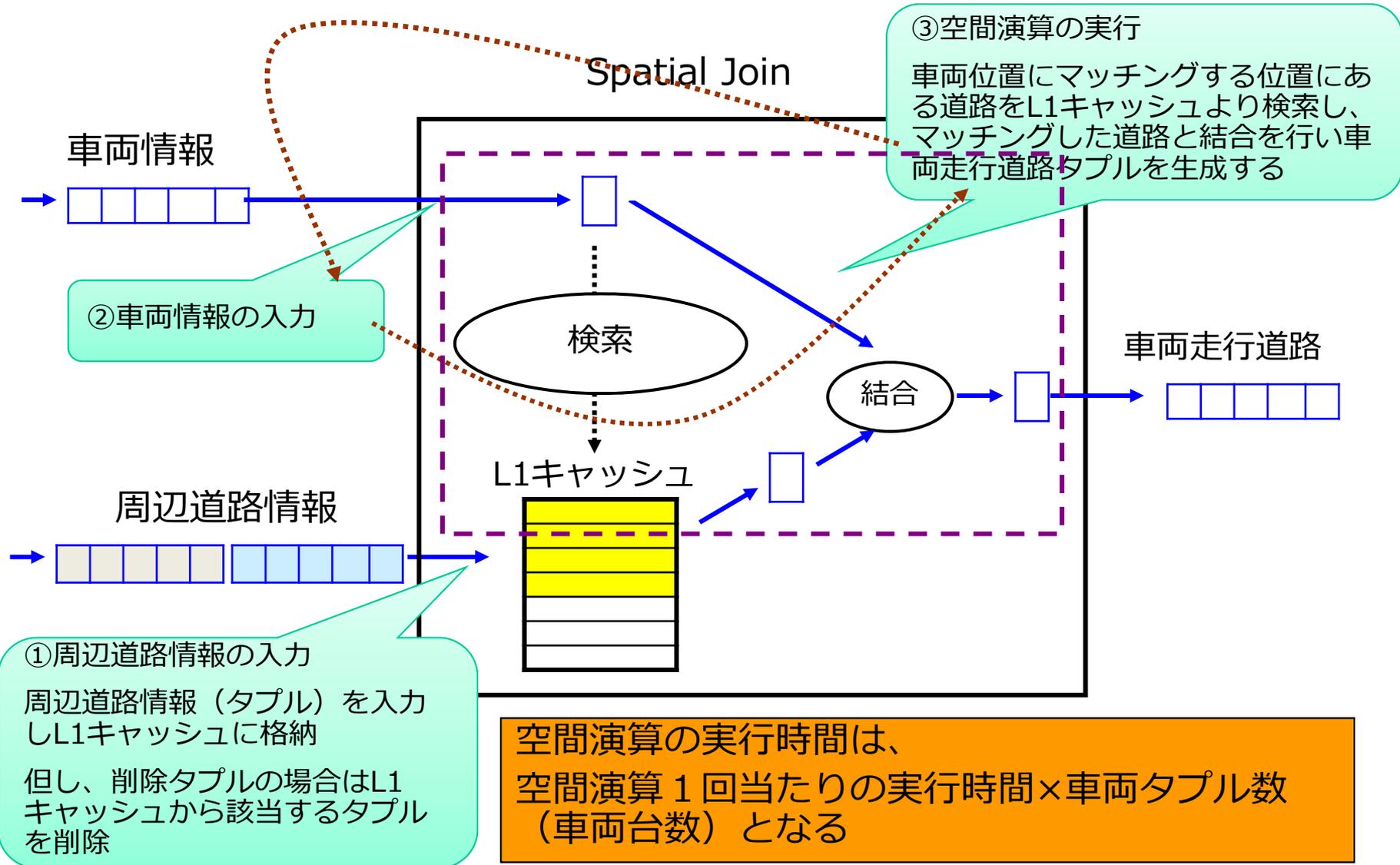


キャッシュ削除範囲

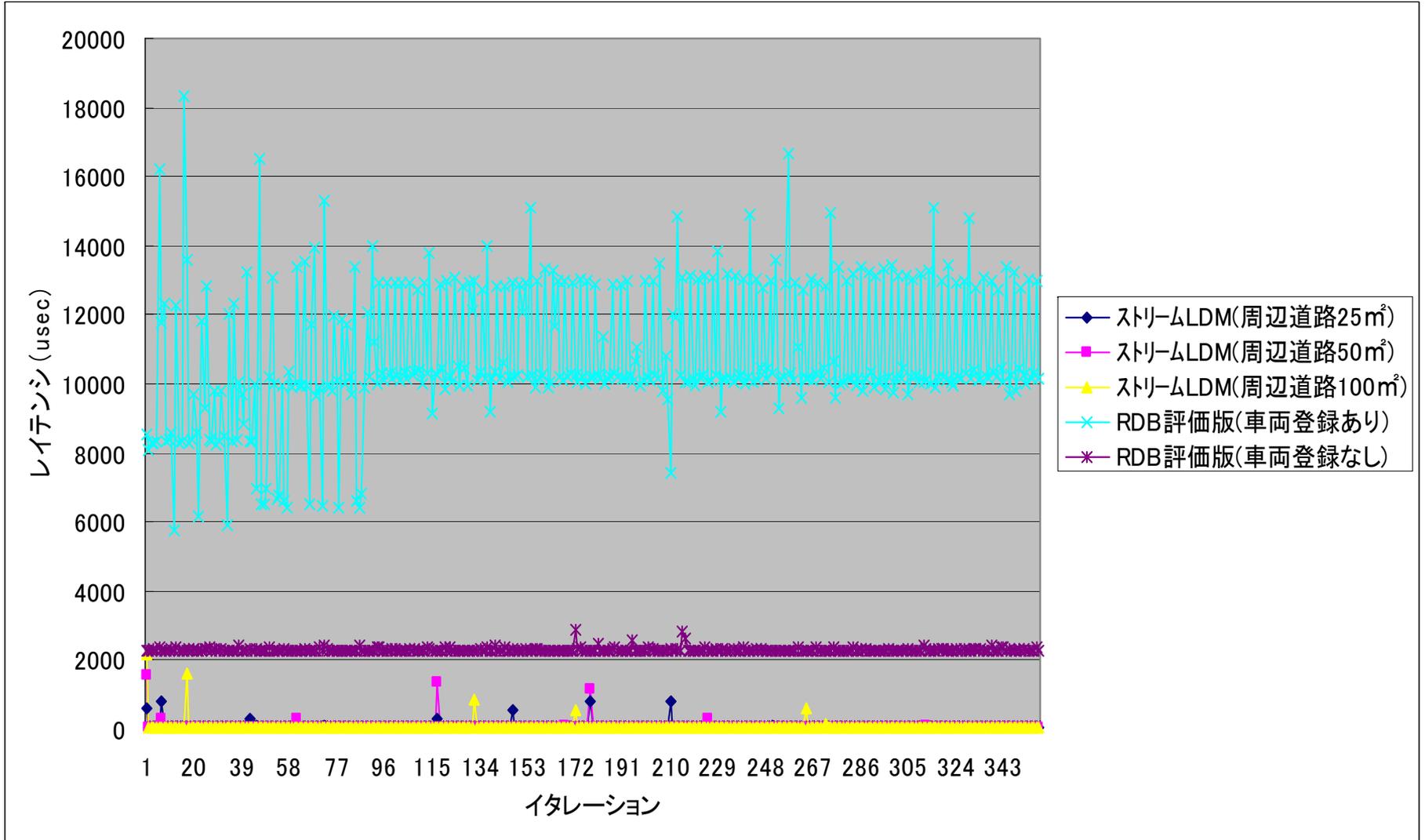
キャッシュの更新時間はキャッシュ追加範囲に含まれる地図情報のデータ量（必ずしも一定ではない）に依存する



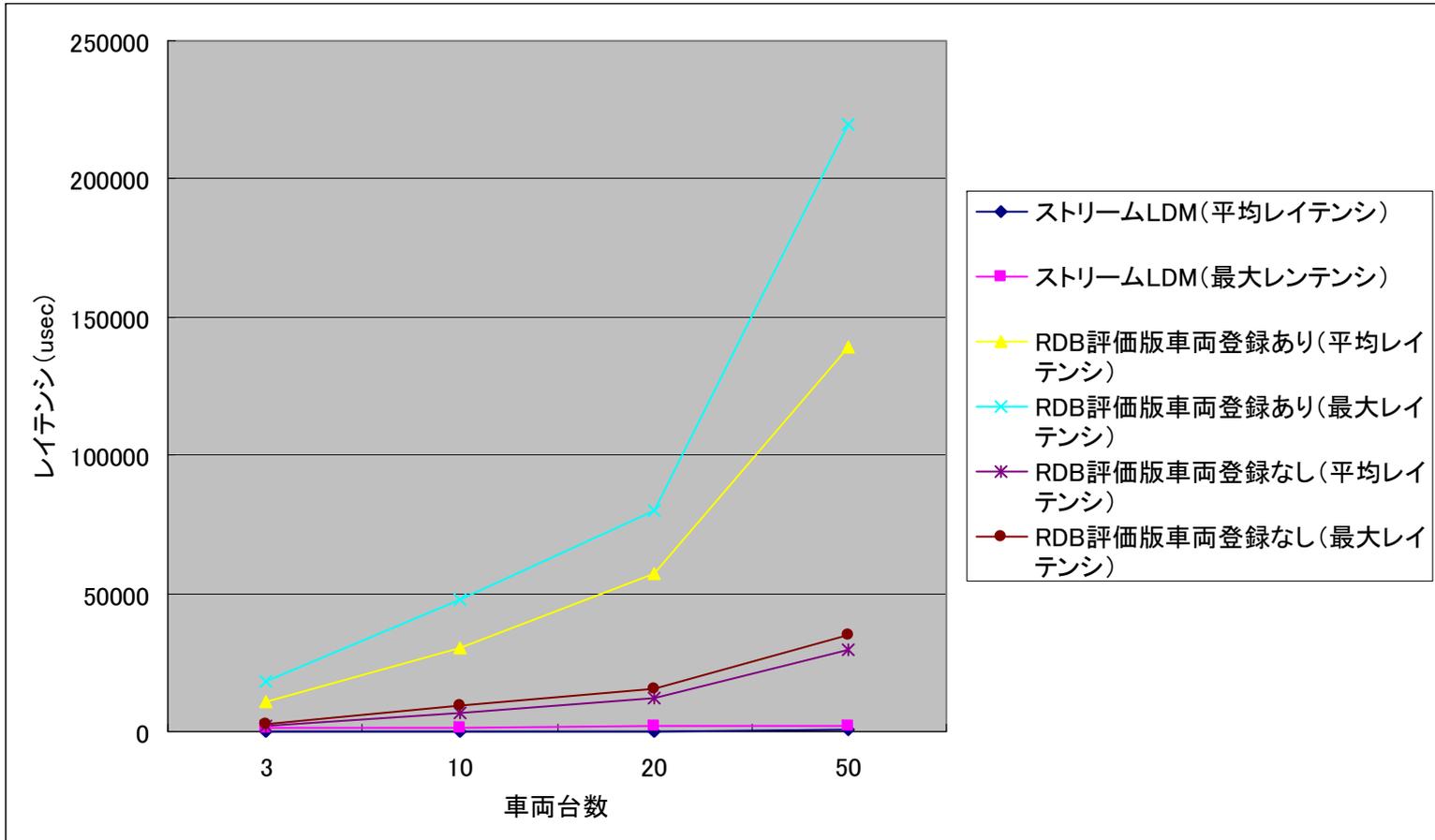
空間演算（位置マッチング）実行



遅延時間評価 (周辺車両3台)



遅延時間評価 (周辺道路100m²)



地図データストリーム化 まとめ

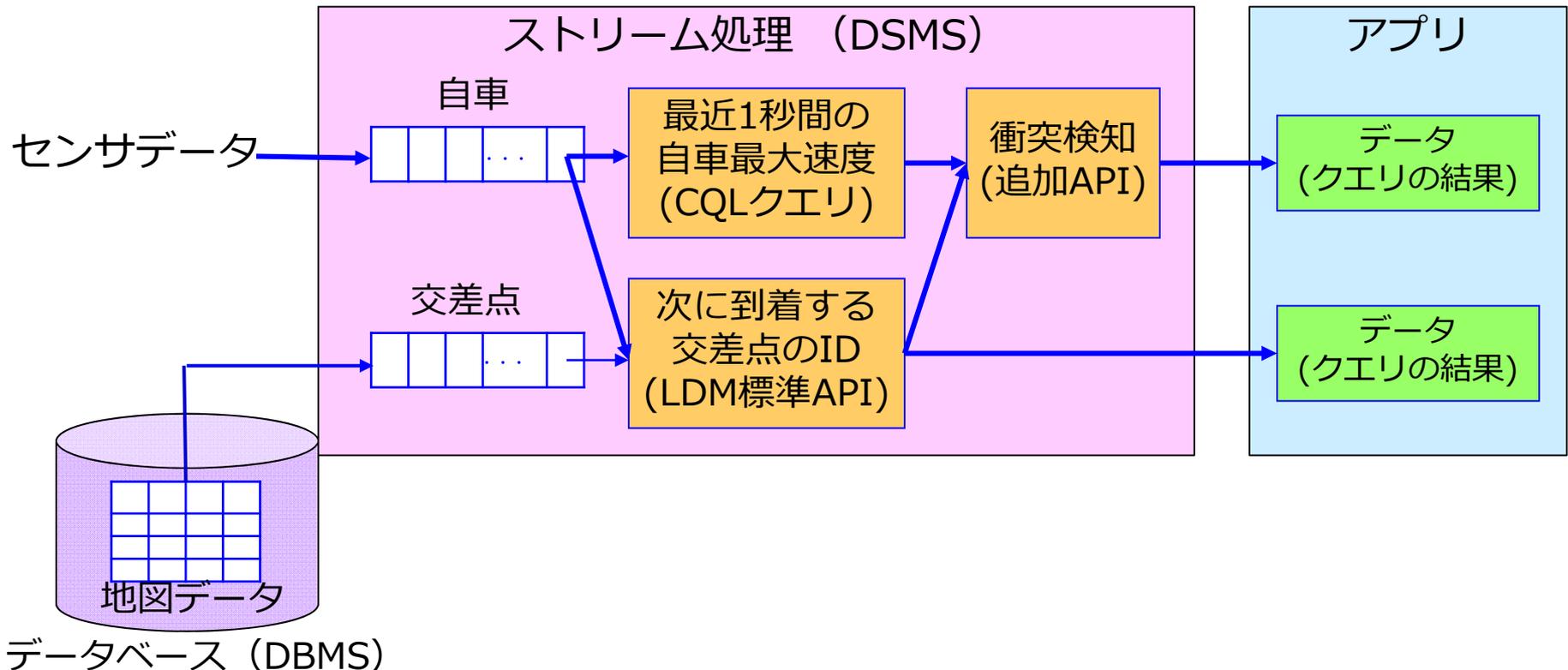
- ストリームLDMの従来方式に対する優位性を確認
 - 特に車両台数が増加した場合、処理が追いつかないという従来方式の問題点は解決可能
 - ストリームLDMの性能を決める主要因は、キャッシュの更新時間とクエリの実行時間（本評価では空間演算のみ）
 - 本評価ではメッシュサイズのチューニングによりキャッシュの最大更新時間を1ミリ秒以内に抑制可能（キャッシュ範囲を本評価の数倍にした場合でも数ミリ秒のオーダーになると予想）
 - 空間演算は車両台数最大50台に対してマイクロ秒のオーダーで実行可能
 - （衝突検知クエリでの評価が必要であるが、既存クエリの実行時間からみて数ミリ秒のオーダーで実行可能と予想）
- ⇒ 車両データの想定入力レート（車車間通信） 100ミリ秒の元で少なくとも数十台の車両に対して、ストリームLDMにより十分処理可能
- キャッシュプリフェッチにより更に応答時間を削減可能

ストリーム処理ソースコード生成 (ストリームLDM パート2)



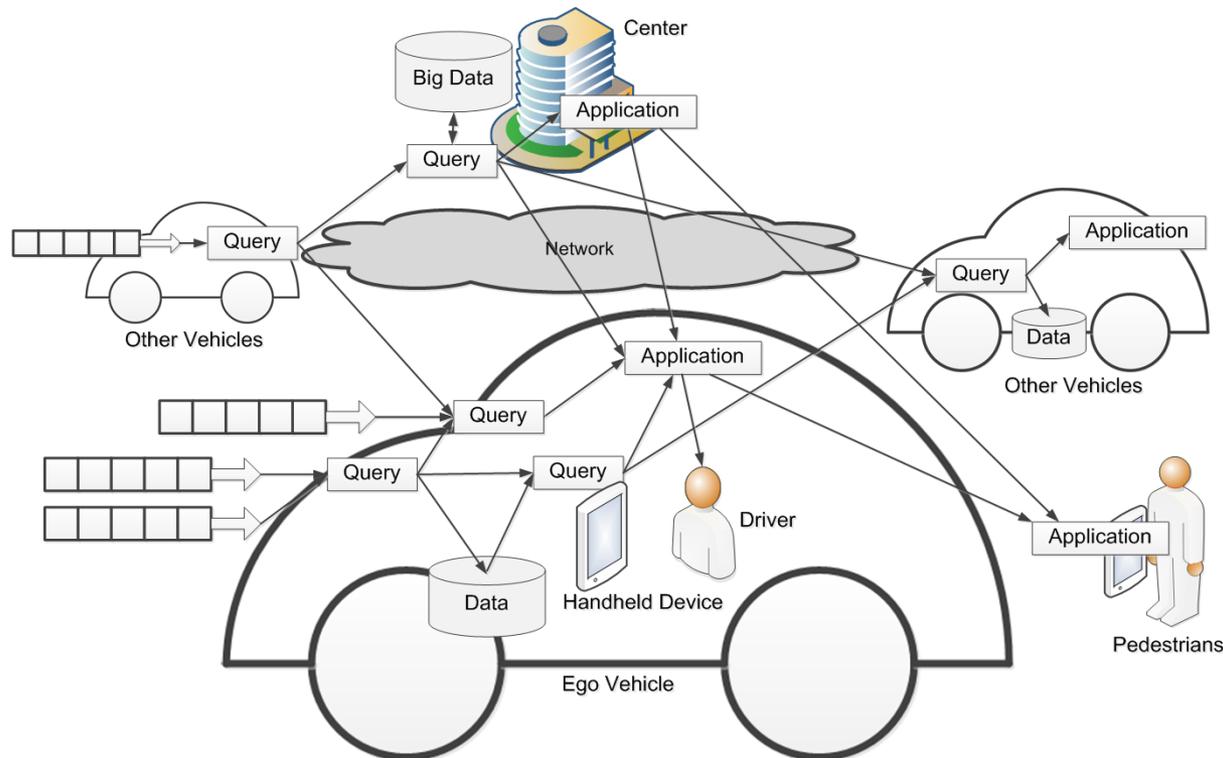
LDM APIのDSMSクエリによる実現

- ストリームからのデータ取得にLDMの標準API, 追加APIを利用
- LDMの標準API, 追加APIはDSMSのクエリを用いて定義



データ処理の最適配置

- ストリーム処理により分散環境の構築が容易
 - データ処理モジュールの最適配置
 - 通常DBMSではデータを集約する必要あり
 - 分散環境における位置透過, 高可用性, 負荷分散の容易性を活かし, 自車両 (ECU) と車外の区別無くストリームを実現可能



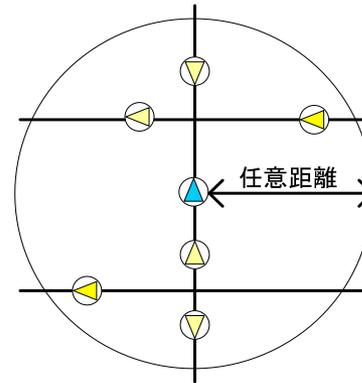
車載データ処理コンポーネント

- 事前定義した車載データ処理するためのコンポーネントを定義
- 各コンポーネントはオペレータの組み合わせで構成

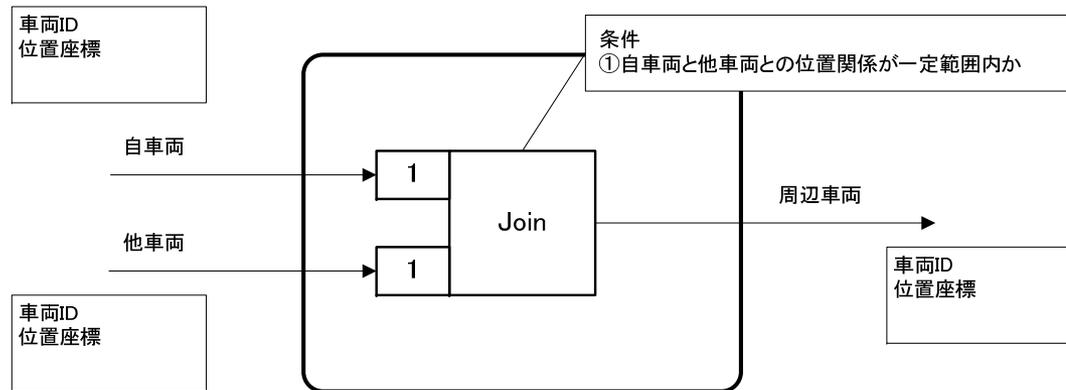
- 例：周辺車両特定

①

- 定義



- コンポーネント仕様

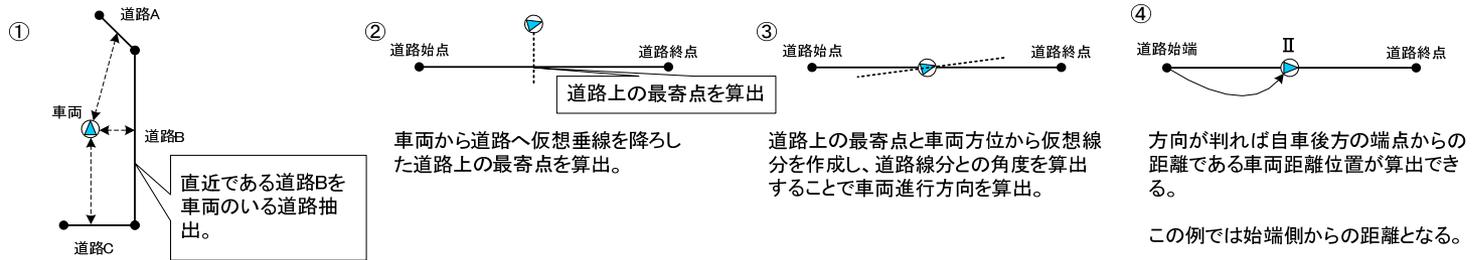


- コンポーネントの組み合わせによりアプリケーションを開発

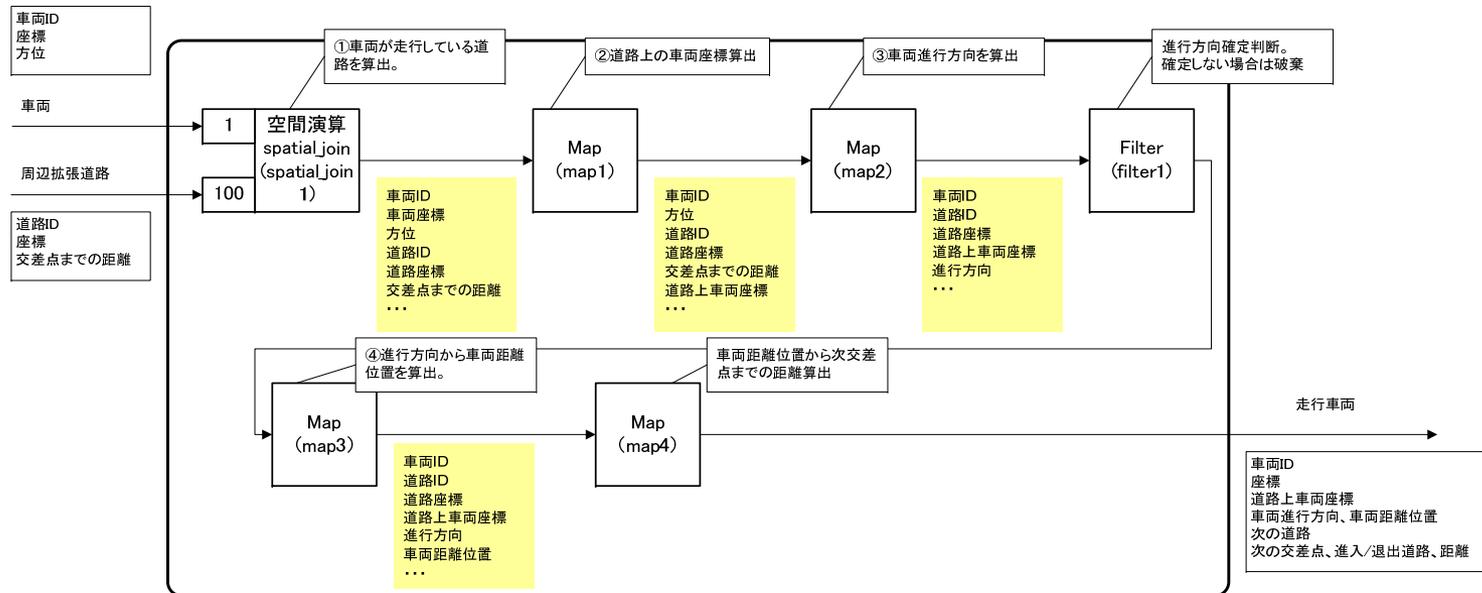
データ処理モジュール（例）

例：自車両走行道路

定義

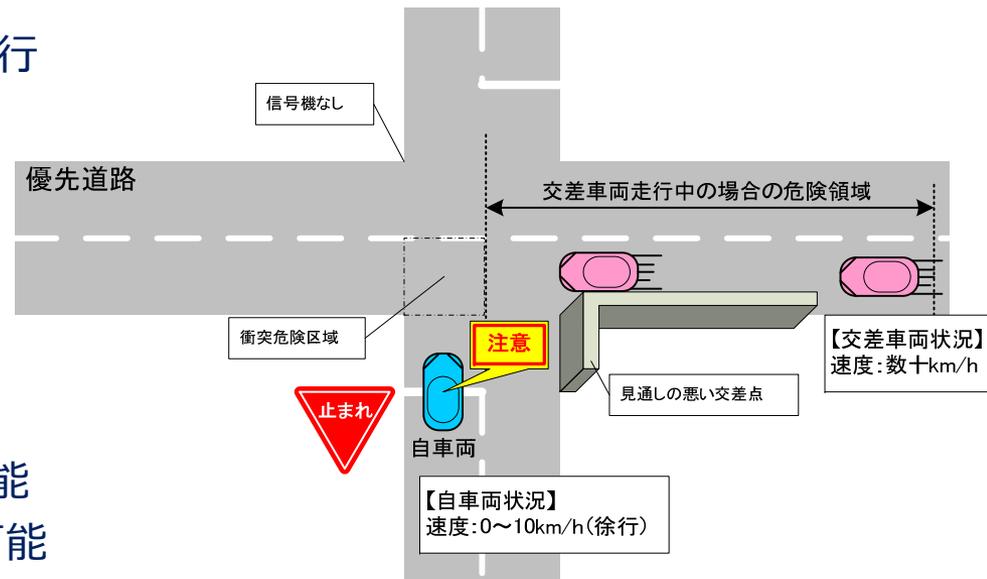


データ処理モジュール P-DSC（オペレータの組み合わせ）



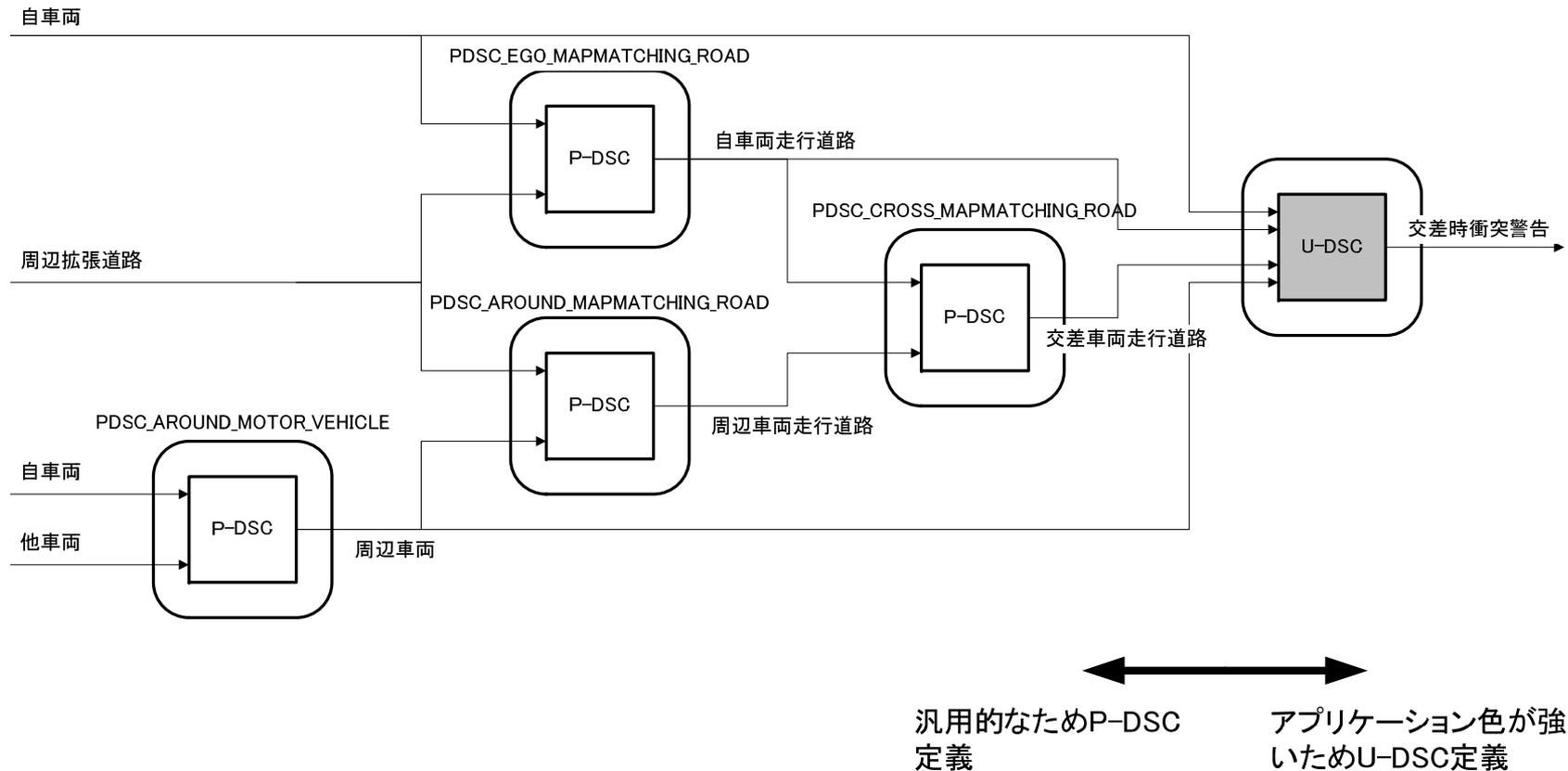
アプリケーション設計

- アプリケーション仕様の定義
 - 例：交差点衝突警告シナリオ
 - 信号のない交差点
 - 自転車側が非優先道路を走行
 - 一時停止が存在
 - 他車両が優先道路を数十km/hで走行
 - 自転車が交差点まで距離10m以内
 - 自転車の速度が徐行程度
 - 信号の有無が判別可能
 - 見通しの悪い交差点かの判別可能
 - 優先道路/非優先道路の走行判別可能
 - 交差車両接近の方向(左右)を判別可能

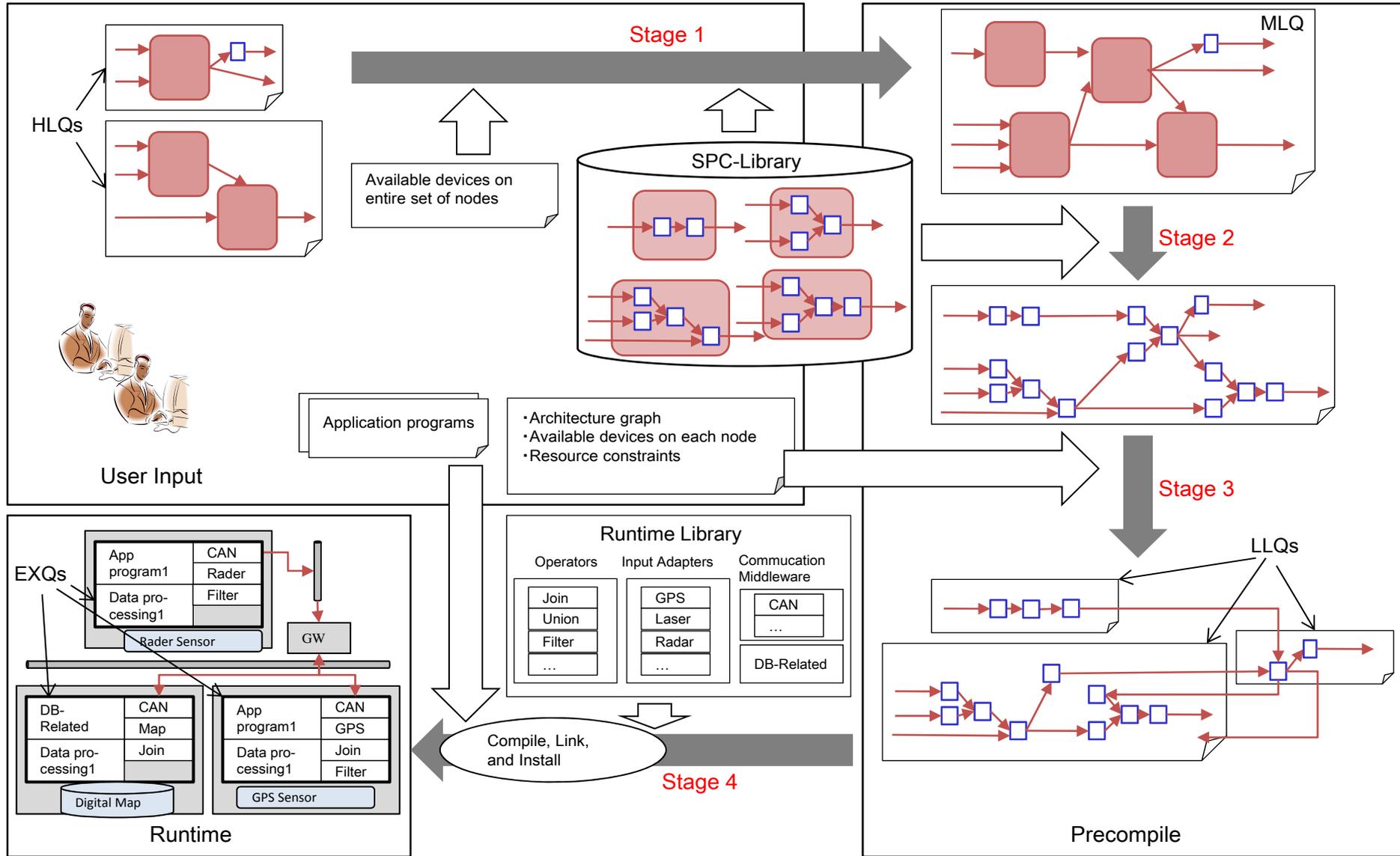


アプリケーション設計

- 事前定義した汎用データ処理モジュール(P-DSC)の組み合わせでアプリケーション設計
 - アプリケーション独自のモジュール(U-DSC)は新規設計の必要あり



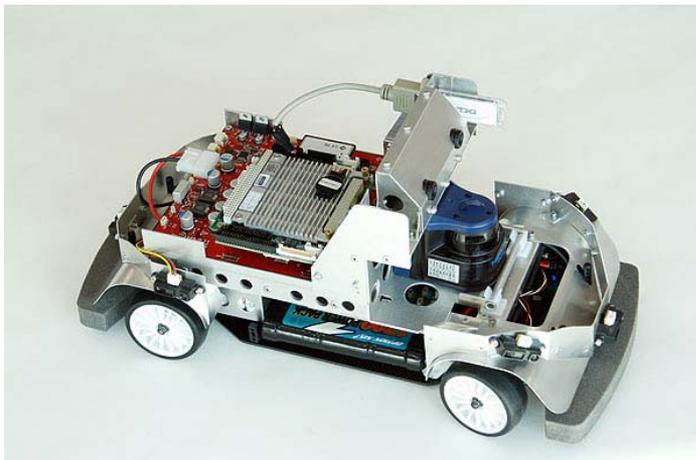
ターゲットに合わせてプログラム自動生成



組み込みシステムへの適用検討状況（1）

- ストリームLDMの実行環境
 - ZMP社 RoboCar® 1/10 ロボットカー（情報系ECU相当）
 - CPU：AMD Geode LX800 Processor 500MHz、メモリ：512MByte
 - OS：Linux (Fedore 10)
 - Altera 社 Nios II Cyclone III 3C25 FPGAボード（制御系ECU相当）
 - CPU：Nios II/f processor core、メモリ：DDR SDRAM 133 MHz, 32 MBytes
 - OS：TOPPERS/ATK2 (AUTOSAR 準拠)

RoboCar



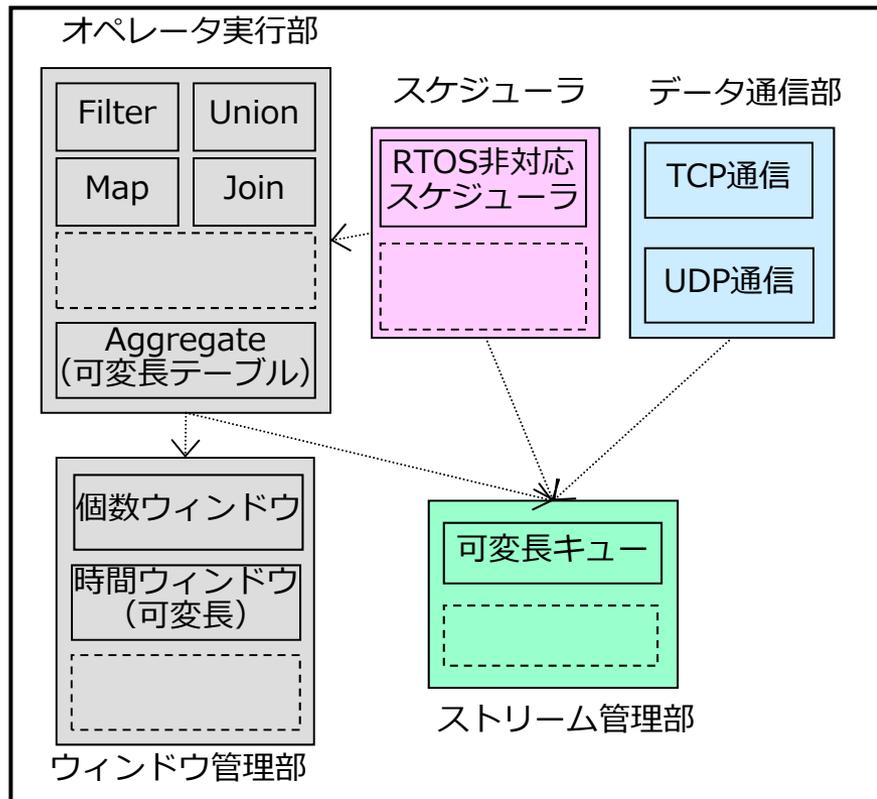
Alteraボード



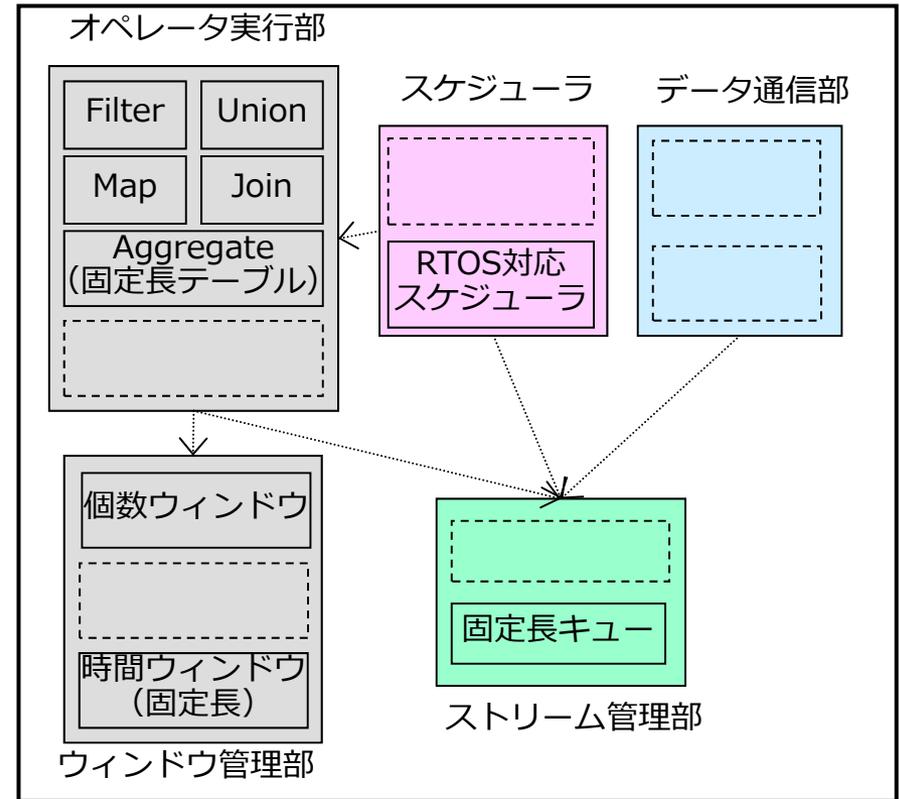
システム依存性の低減

- RoboCar, および, Alteraボード上において, 複数のコンポーネントを選択し動作を確認
- コンポーネント置換え時に, 他コンポーネントの変更は発生せず

RoboCar



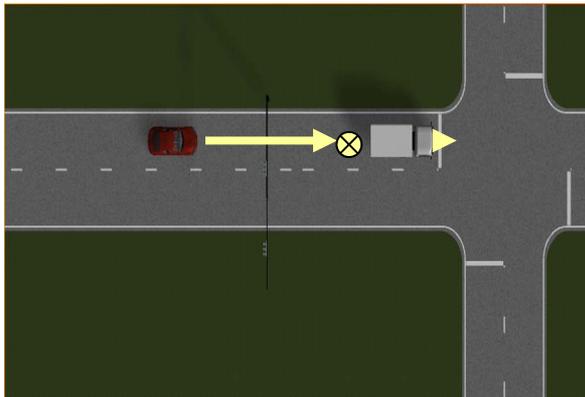
Alteraボード



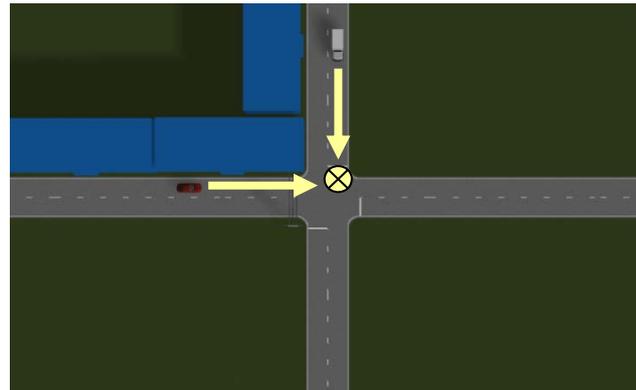
安全運転支援アプリケーション例

- 車載データ処理クエリとして以下の衝突検知処理を想定
 - 前方衝突における衝突検知
 - 交差時衝突における衝突検知
 - 右折時衝突における衝突検知

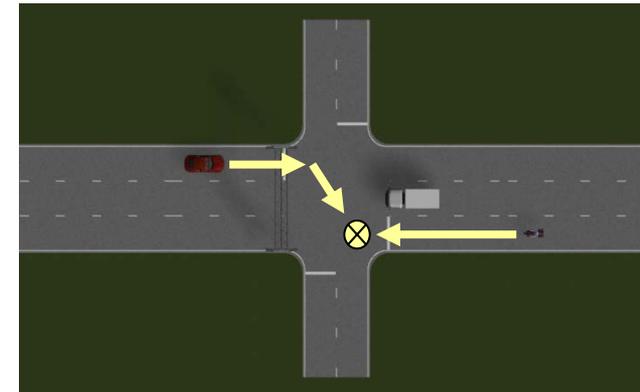
前方衝突検知



交差時衝突検知



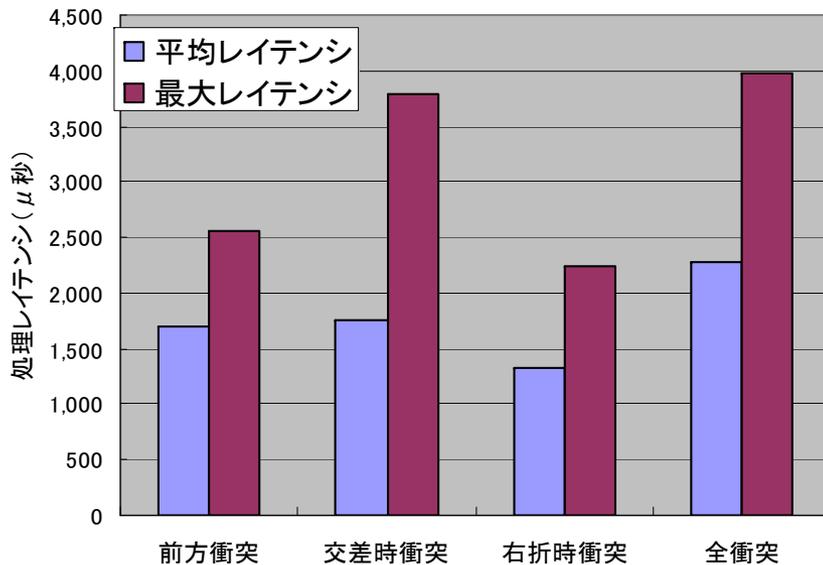
右折時衝突検知



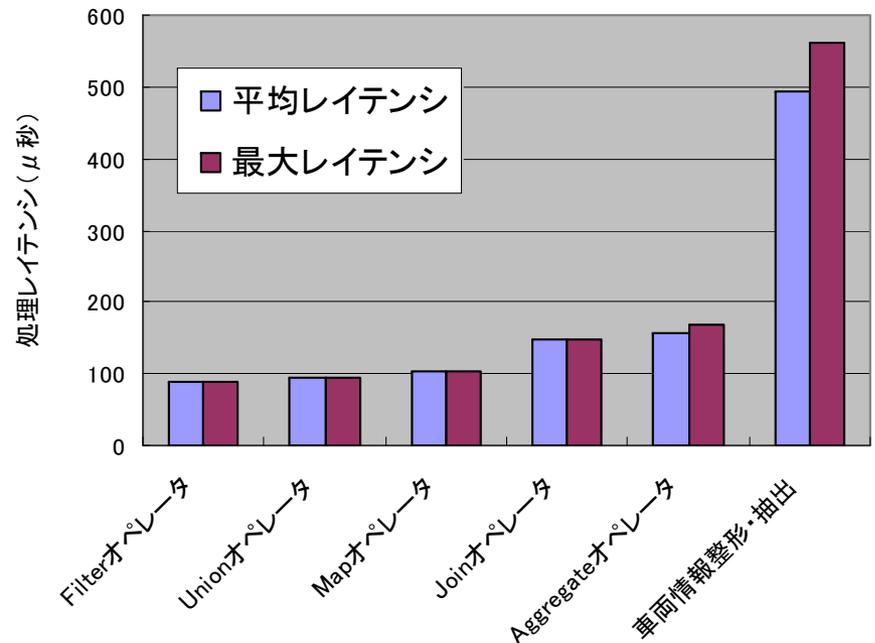
アプリケーション開発プラットフォーム 性能評価

- 組み込みDSMSの性能評価
 - RoboCar、Alteraボード上で車載データ処理を実行
 - 妥当な処理レイテンシに収まることを確認

RoboCar



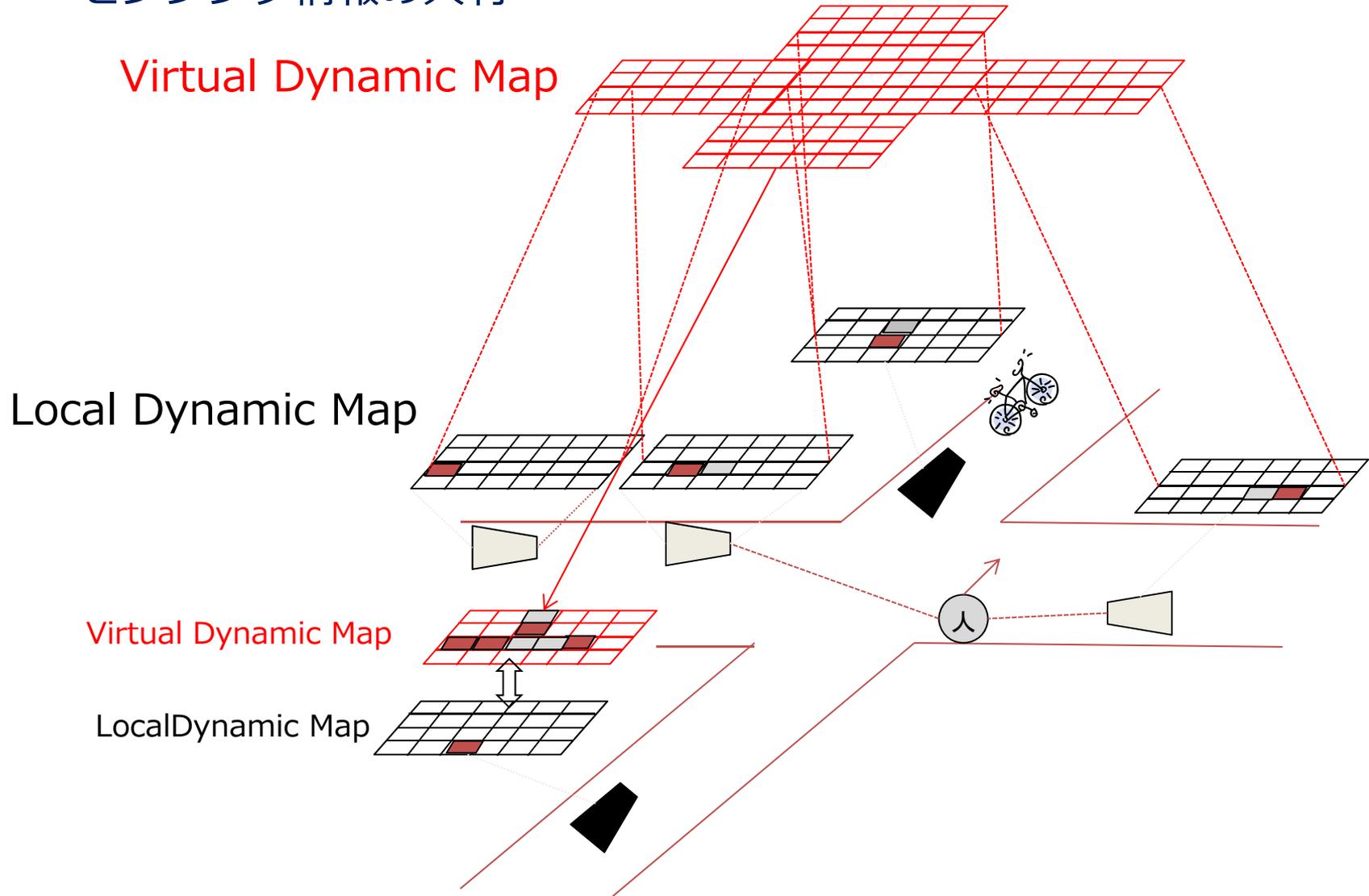
Alteraボード



センシング情報の共有 (仮想ダイナミックマップ)

仮想ダイナミックマップ

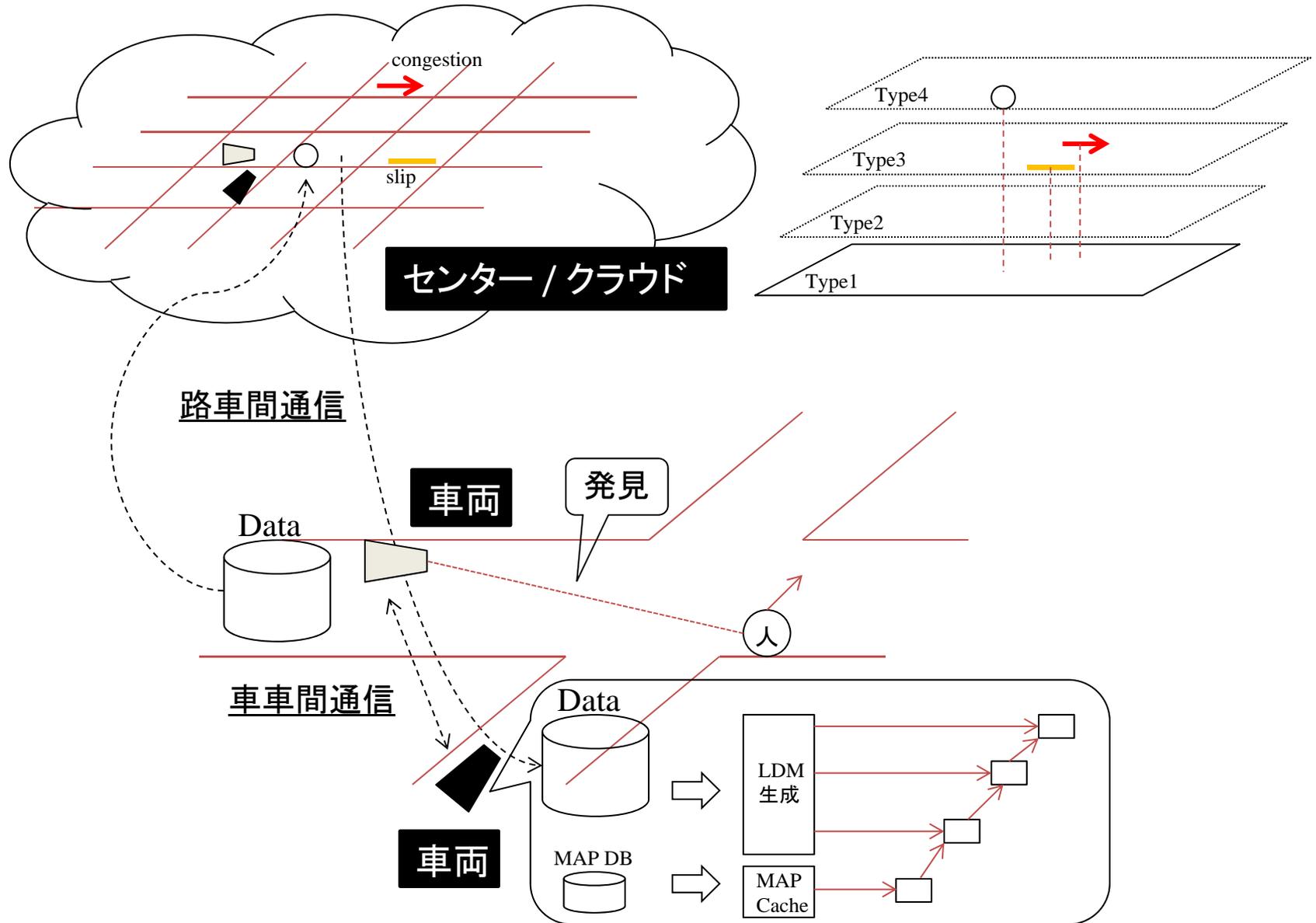
- センシング情報の共有



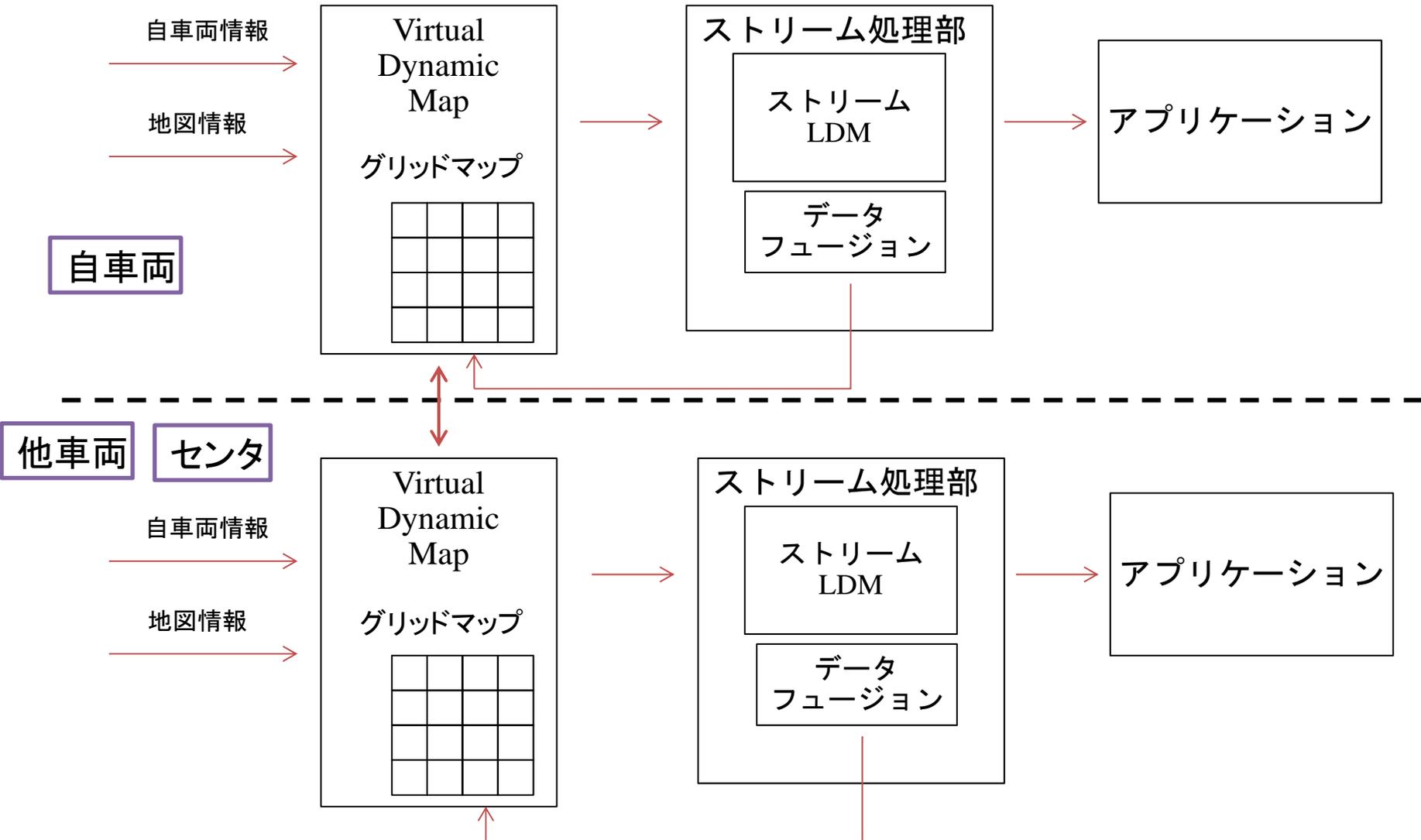
仮想ダイナミックマップ概要

- 機能
 - ストリームLDMによって生成されるストリームデータを車両やセンターで分散管理
 - 車両の位置情報, 時間情報に応じて適切な情報を出力する
- 入力
 - 自車両のセンサなどから生成されるストリームLDMデータ
 - 他車両, センターから通信で管理されるストリームLDMデータ
- 内部処理
 - データの同期/共有
 - データの更新/フュージョン
- 出力
 - 入力のストリームデータを車両の位置, 時間に応じて対応する情報をマップマッチング機構に提供
 - アプリケーションは, マップマッチング機構にて対応するType1データと関連付けられ提供

ストリームLDMとの結合



実装モデル



他車両 センタ

ドライバ（人）とシステムの融合 （ドライバエージェント）

ドライバモデル

- ドライバの操作と自動運転システム操作の融合

車両周辺環境情報

Driving Environment

- Scenery
 - Location
 - Map
 - Lane
 - Road condition
- Restriction
 - Traffic lights
 - Signs
- Vehicle status
 - Velocity
 - Yaw rate
- Obstacles
 - Other vehicles
 - Pedestrians
 - Road hazard
 - Miscellaneous

Judgement



ドライバの操作と自動運転システム(ドライバモデル)操作の融合

車両制御情報

- ### Vehicle Control
- Gas pedal
 - Break pedal
 - Steering

Recognition

Driver

Operate

Personalized

Monitoring

Data Fusion

Control

Sensing

Driver Model

Actuate



自動運転システム(ドライバモデル)操作データ生成

Feedback

融合データのフィードバック

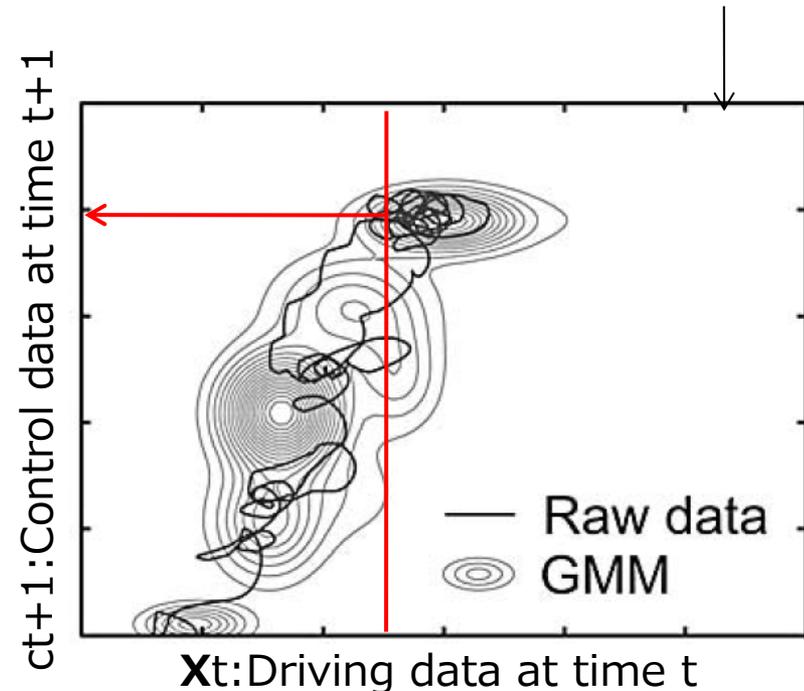


Driver model

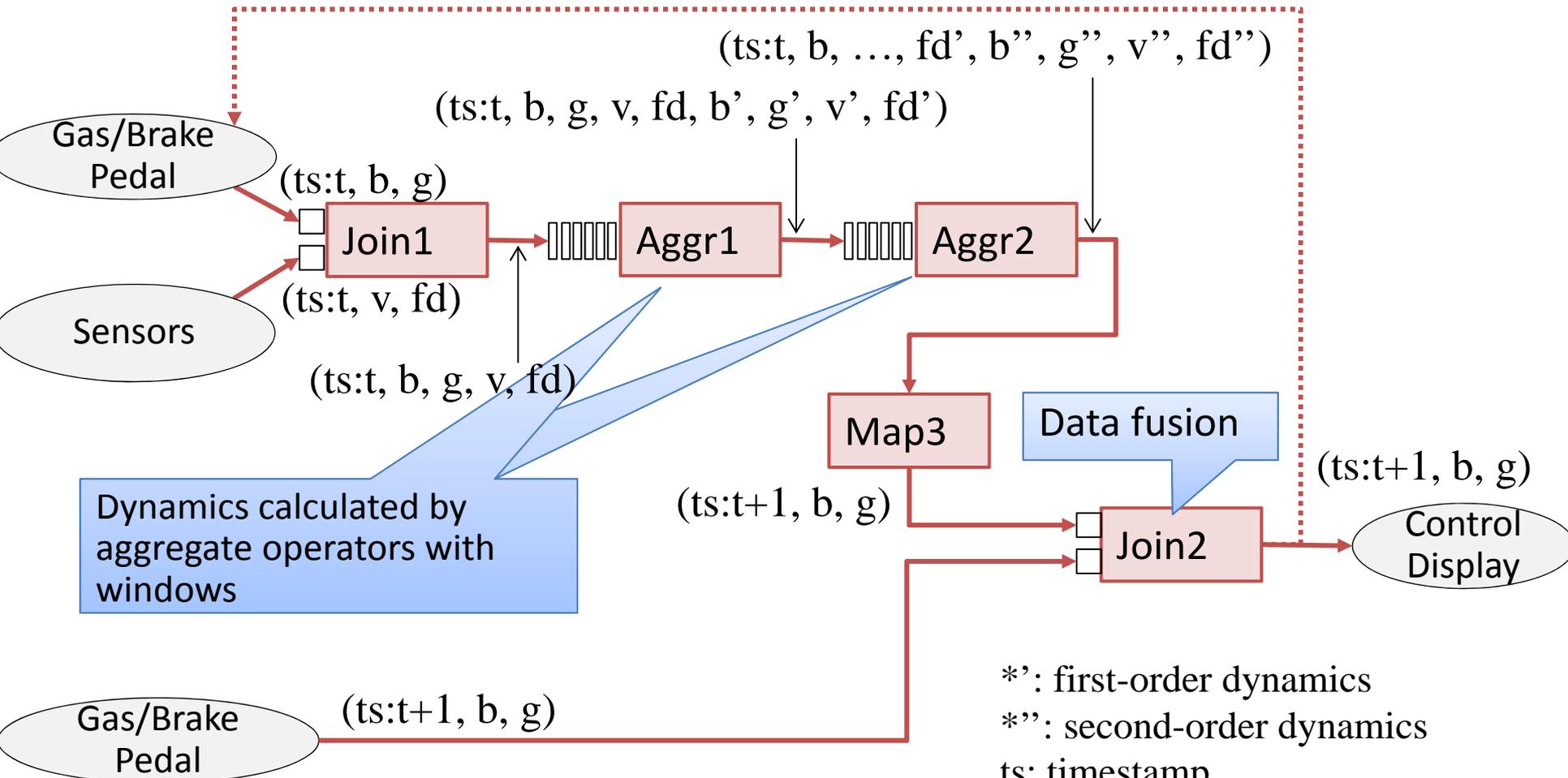
- Generated as a joint probability distribution $p(\mathbf{x}, c)$ from driving data and control data at next time, beforehand
 - $p(\mathbf{x}, c)$: GMM
 - \mathbf{x} : driving data, c : next control data

Miyajima et al: Driver modeling based on driving behavior and its evaluation in driver identification, Proceedings of the IEEE, 2007

- Predicts next control data c_{t+1} maximizing conditional probability $p(c_{t+1} | x_t)$, given current driving data x_t

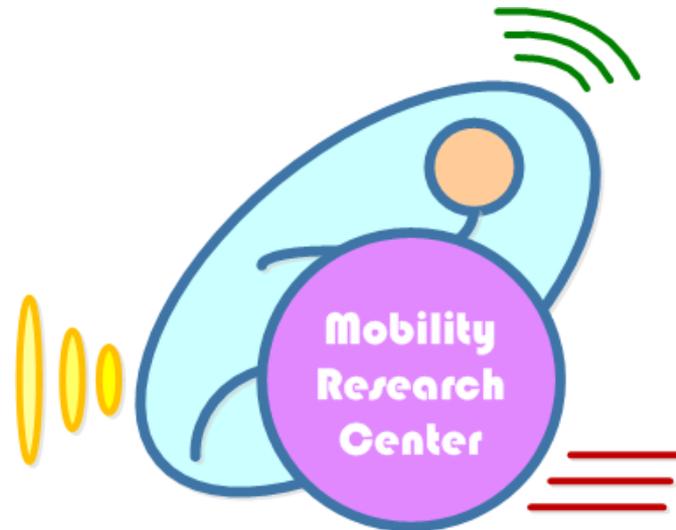


ストリームによる処理



Our DSMS enables operators' parameters described in C language

Doshisha Mobility Research Center



Move to the Future

- URL <http://ics.doshisha.ac.jp/mrc/>
- E-Mail rc-mblty@mail.doshisha.ac.jp