

# 高齢者ドライバを遠隔地から見守るシステムの提案

山岸 弘幸<sup>†1</sup> 鈴木 秀和<sup>†2, †3</sup> 寺澤 圭史<sup>†2</sup> 渡邊 晃<sup>†1</sup>

名城大学理工学部<sup>†1</sup> 名城大学大学院理工学研究科<sup>†2</sup> 日本学術振興会特別研究員<sup>†3</sup>

## 1. はじめに

人、車両、道路を情報通信技術によって一体化する ITS (Intelligent Transport Systems) に注目が集められている。特に、ITS のサービスの一部であるテレマティクスサービスは、国内外の様々な企業によって、それぞれ独自のサービスとして展開されている。

テレマティクスサービスはドライバ自身を支援するサービスである。しかし、高齢化社会が想定される日本では、ドライバ自身に対するサービスだけではなく、ドライバを見守る周囲の人々に対するサービスが重要になると考えられる。

そこで 本稿では、ドライバの家族や親戚がドライバを遠隔地から安心して見守ることができるサービスを提案する。

## 2. 既存技術

テレマティクス (Telematics) サービスとは、自動車などの移動体に通信システムを組み合わせ、リアルタイムに情報サービスを提供するシステムである。国内ではトヨタの G-BOOK[1] や日産のカーウィングス、海外では General Motors の OnStar など、企業がそれぞれ独自のサービスを展開している。

例えば、G-BOOK では現在地から目的地までに利用する主要道路や有料道路で、新しい道路が開通された部分をカーナビゲーションに配信するサービスや VICS (Vehicle Information and Communication System) による最新の交通情報と過去の統計データから、今後の交通状況を予測するサービスなどがある。また、緊急事態発生時の車両からヘルプネットセンターへの通報を補助するヘルプネットサービスがある。

しかし、これらのサービスは全て、ドライバ

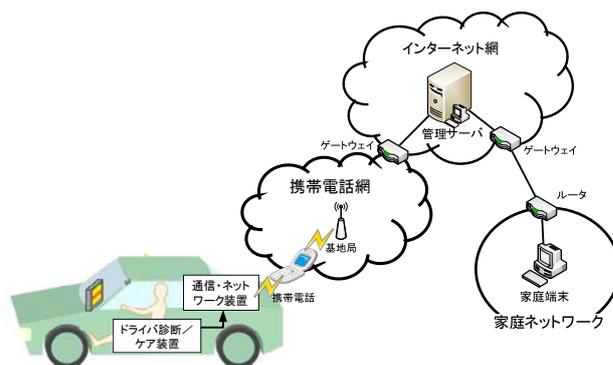


図 1. 提案システムの構成

自身を対象としたサービスである。また、独自の管理センターを利用しているため、一般ユーザには公開されていないクローズなサービス形態となっている。今後、高齢化社会が深刻化する日本では、高齢者ドライバの安全を家族や親戚がいつでも確認できるサービスが必要になると考えられる。

## 3. 提案方式

### 3.1. 提案システムの構成

本提案システムの構成を図 1 に示す。まず車両に搭載するドライバ診断/ケア装置は、ドライバの脈拍や操舵情報、位置情報などのセンサデータを収集する。通信・ネットワーク装置はこれらのセンサデータを収集し、携帯電話網経由でインターネット上の管理サーバへ送信する。管理サーバは受信したセンサデータを整理し、ドライバ情報として保持しておく。家族や親戚などの見守る人は管理サーバにアクセスすることにより、蓄積されたドライバ情報をいつでも閲覧できる。

### 3.2. センサデータの通信経路

ドライバ診断/ケア装置と通信・ネットワーク装置の間はイーサネットにより接続する。次に、通信・ネットワーク装置から管理サーバへのセンサデータの送信には携帯電話網を利用する。これは、現在車両から直接インターネットへ接続する環境が十分に整備されていないためである。このとき、PCCOM[2]によってセンサデータの改ざんや漏洩を防止する。

“Proposal of a Watching for Senior Drivers from Remote Places”

†1 Hiroyuki Yamagishi and Akira Watanabe  
Faculty of Science and Technology, Meijo University

†2 Hidekazu Suzuki and Keiji Terazawa  
Graduate School of Science and Technology, Meijo University

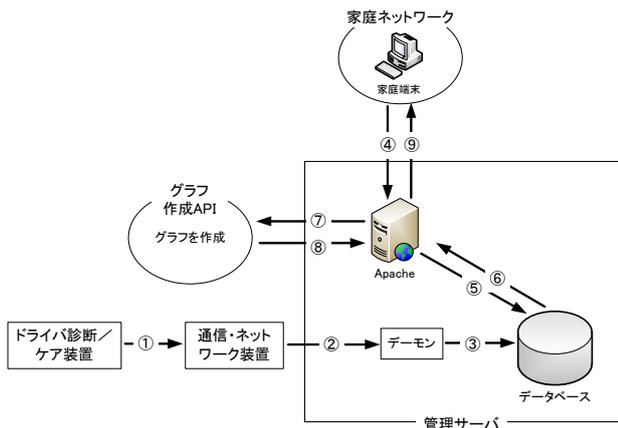


図 2. 提案システムの動作

### 3.3. センサデータ閲覧

見守る人が管理サーバにアクセスする際は、家庭端末から管理サーバを指定してユーザ ID とパスワードを入力する。管理サーバは家庭端末からセンサデータの閲覧要求を受信すると、グラフ作成 API (Application Program Interface) によってセンサデータをグラフ化して表示する。ドライバの閲覧には SSL (Secure Socket Layer) を利用してドライバ情報の改ざんや漏洩を防止する。

## 4. 車両側・管理サーバ側の動作

### 4.1. 車両側の動作

通信・ネットワーク装置は、まずドライバ診断/ケア装置からセンサデータを取得する (図 2. ①)。このとき、ドライバ診断/ケア装置による各センサからのセンサデータの収集と、通信・ネットワーク装置によるドライバ診断/ケア装置からのデータ収集が同時に起こり、ファイルアクセスの衝突が発生する可能性がある。そこで提案システムでは、NFS (Network File System) を用いることで、同時処理によるファイルアクセス衝突を防止する。

次に、取得したセンサデータを管理サーバに送信する (図 2. ②)。このとき、拡張性の高い xml (Extensible Markup Language) 形式で送信することにより、今後センサ情報の種類が増加しても十分対応できるものとする。

### 4.2. 管理サーバ側の動作

管理サーバ内のソフトウェア (デーモン) は通信・ネットワーク装置からセンサデータを受信し、管理サーバ内のデータベースに登録する (図 2. ③)。

家庭端末からドライバのセンサデータ閲覧要求を Web サーバソフトウェア (Apache) が受信すると (図 2. ④)、管理サーバ内のデータベースからセンサデータを読み出す (図 2. ⑤、

⑥)。

その後、読み出したセンサデータをグラフ作成 API によりグラフ化 (図 2. ⑦, ⑧) し、家庭端末へ送信すること (図 2. ⑨) で見守る人にドライバの現状を把握しやすくする。

現在、Google 社がサービスを提供しているグラフ作成 API 「Google Chart API」がある。しかし、このサービスは管理サーバから Google Chart API までの経路が暗号化されていない。また、家庭端末に送信されるグラフは画像形式で生成されるため、リアルタイム性に欠けている。そこで、本提案では管理サーバ内でグラフ作成が可能でリアルタイム性のある Flex をグラフ作成 API とする。

管理サーバ内のデータベースには、見守る人が閲覧したいセンサデータごとのグラフが表示できるようなテーブルを定義する。

また、グラフ作成 API でセンサデータを読み込めるよう、データベースと Web サーバソフトウェア間のインタフェースは xml とする。

### 4.3. システム拡張

図 2 に示すシステムのままでは、見守る人がドライバの現在の状態を管理サーバにアクセスしに行かなければ把握することができない。そこで本提案では、管理サーバ側から見守る人に対してドライバの現在の状態を絵文字等で表現したメールを定期的に配信することにより、見守る人が管理サーバへアクセスする手間を省くことができる。

## 5. まとめ

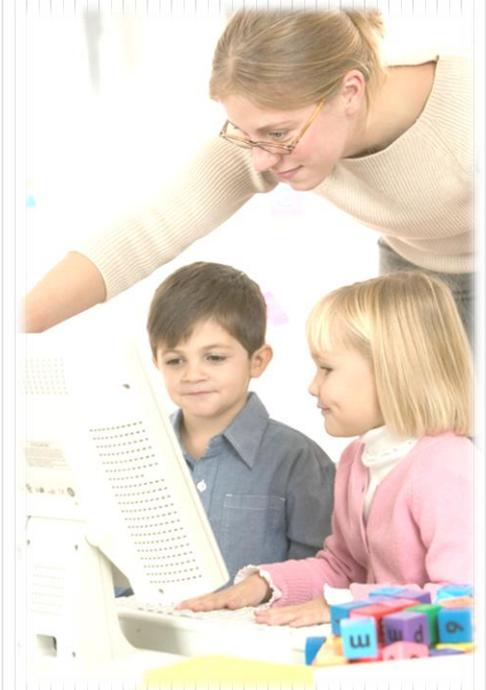
本稿では、高齢者ドライバを遠隔地から見守るシステムの概要、センサデータの送信および閲覧時の通信経路、各装置の動作処理について述べた。

今後はセンサデータを管理サーバに送信する際のパケットフォーマットの検討とグラフ作成 API の検討、更にドライバの緊急時に対する処理の検討を行い、本提案システムの実装と評価を行う。

### 参考文献

- [1] “G-BOOK”, <http://g-book.com/pc/default.asp>
- [2] 増田. 他: “NATやファイアウォールと共存できる暗号通信方式 PCCOM の提案と実装”, 情報処理学会論文誌, Vol. 47, No. 7, pp. 2258-2266, 2006

# 高齢者ドライバを遠隔地から 見守るシステムの提案



名城大学 理工学部

山岸 弘幸 寺澤 圭史 鈴木 秀和 渡邊 晃

# 研究背景

- ITS (Intelligent Transport Systems)
  - 「人」、「車両」、「道路」を情報通信技術によって  
一体化したシステム
    - 例: 自動料金收受システム (ETC<sup>1</sup>)  
道路交通情報通信システム (VICS<sup>2</sup>)
- テレマティクス (Telematics)
  - 自動車などに通信システムを組み合わせて、  
リアルタイムに情報サービスを提供
    - 例: トヨタ「G-BOOK」、日産「カーウイングス」、  
General Motors「OnStar」など

# G-BOOK

- 概要

- 携帯電話やパソコン、車載器などから利用できるコンテンツサービス

## ~MUSIC On Demand~



## ~MAP On Demand~



## ~Safety & Security~



## ~PROBE Communication~



ドライバ自身に対するサービス

# 研究の目的

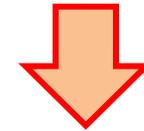
現在の提供サービス

ドライバ自身に対するサービス



要求されるサービス

高齢者ドライバを見守るサービス



見守る人がどこからでも、高齢者ドライバの現在の状態を知る



位置情報、生体情報、舵情報

高齢者人口の増加

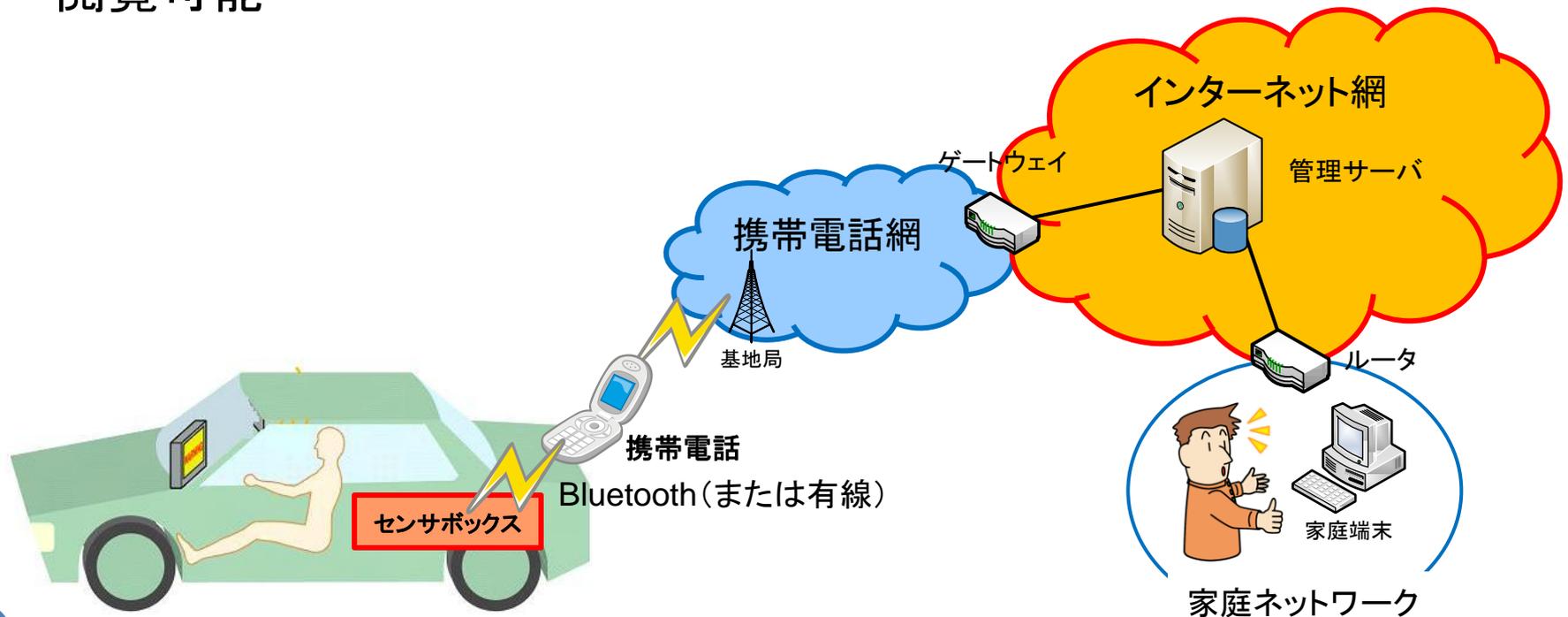


- ・高齢者が運転する機会が増加
- ・身体機能の低下により、運転時に体調が悪化

今後の高齢化社会

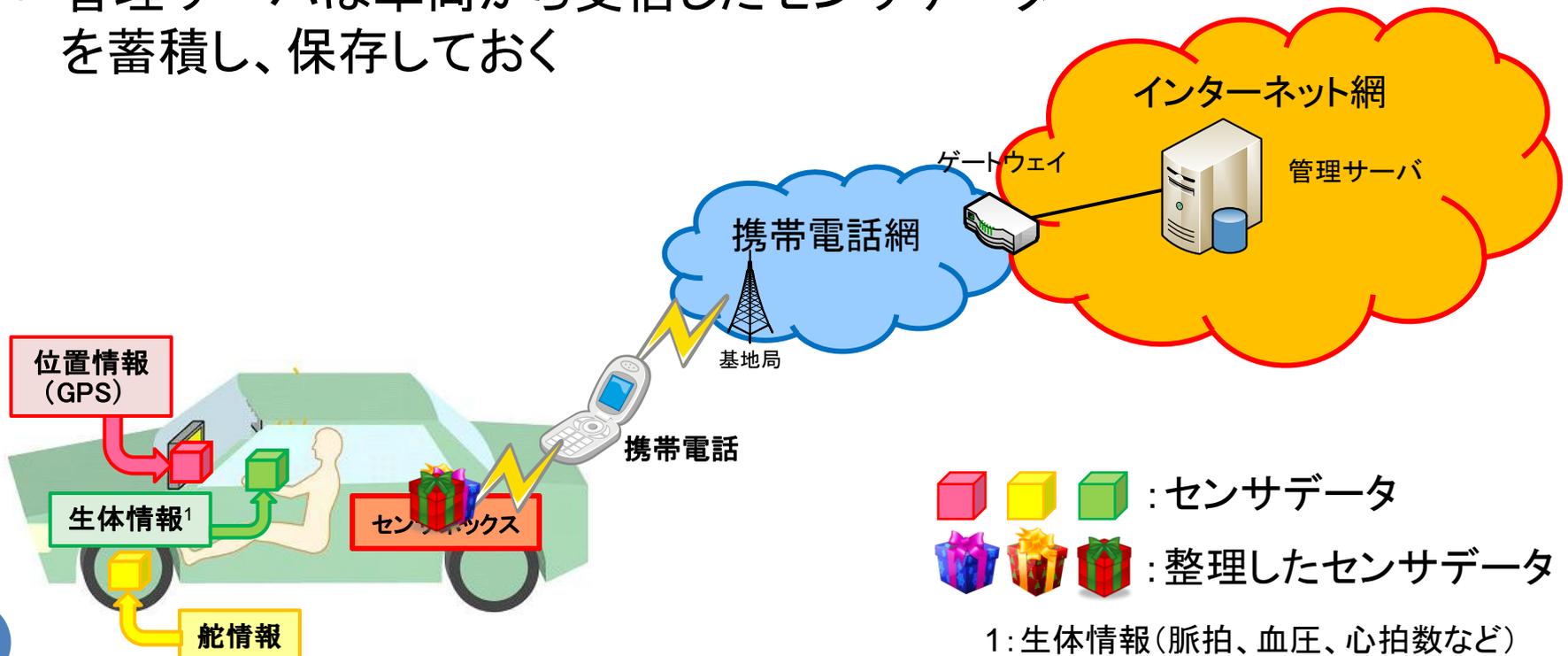
# 提案システム概要

- センサボックスに蓄積情報(センサデータ)を収集
- インターネット上にある管理サーバへ定期的にセンサデータを蓄積
- 見守る人(家族、親戚の方)のみが管理サーバに蓄積した情報を閲覧可能



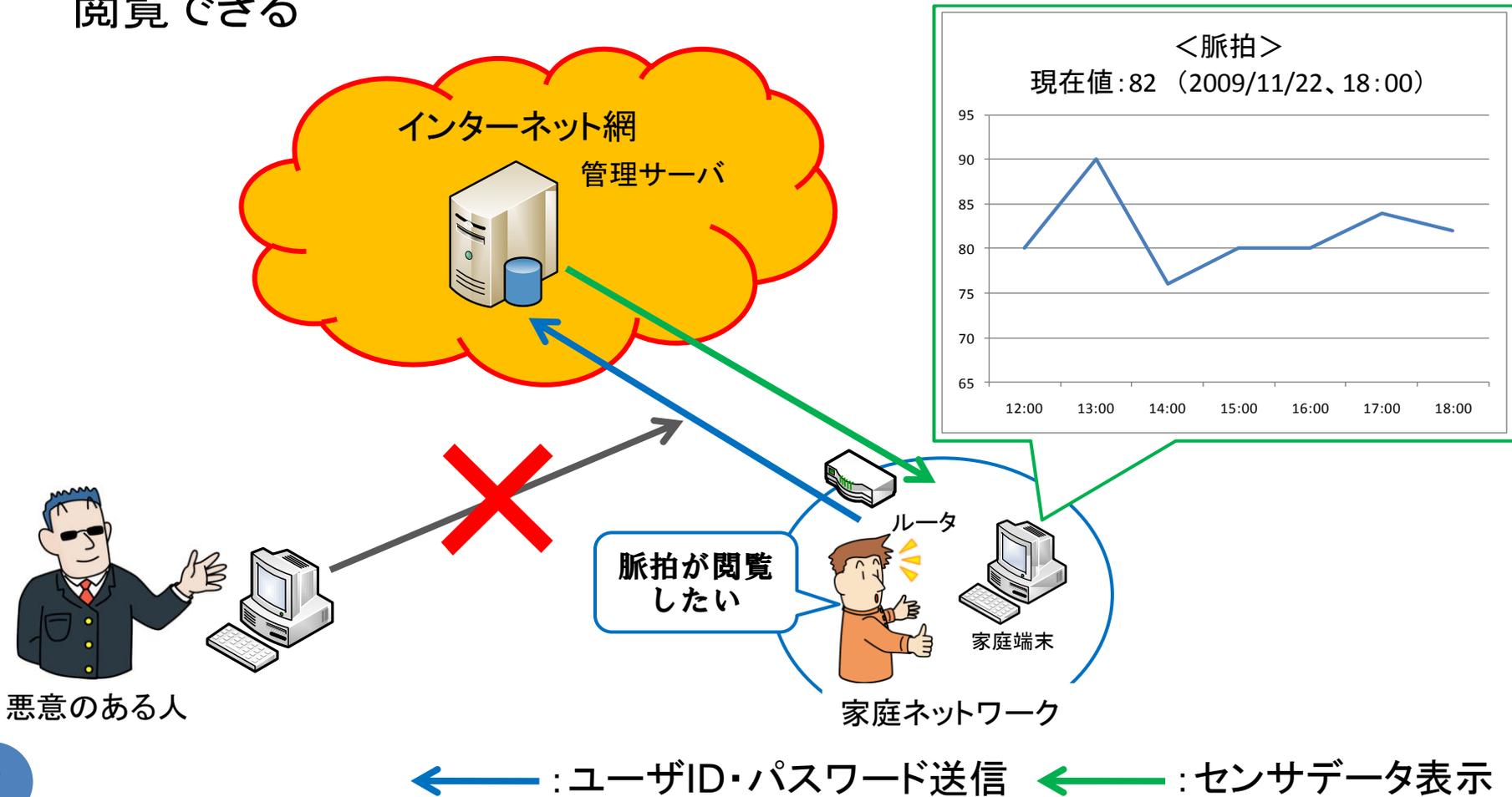
# センサーデータ送信

- センサボックスで様々なセンサーデータを収集
- センサーデータを整理したのち、携帯電話網を経由して定期的にインターネット上の管理サーバへUDPで送信(暗号化通信)
- 管理サーバは車両から受信したセンサーデータを蓄積し、保存しておく



# センサーデータ閲覧

- 見守る人は管理サーバへログインすれば、ドライバのセンサーデータを閲覧できる



# 緊急時の動作

- センサデータが異常な値が続く場合、携帯電話へアラームメールを送信



|  |
|--|
| To : ○○○○  |
| From : △△△△  |
| 件名 : 緊急事態です！！  |
| ドライバの方が危険な状態です。<br><脈拍>の数値が異常な値を続けています。<br>すぐに連絡してください。<br>ドライバの状態 : <a href="http://www.●●●●.▲▲.××">http://www.●●●●.▲▲.××</a> |

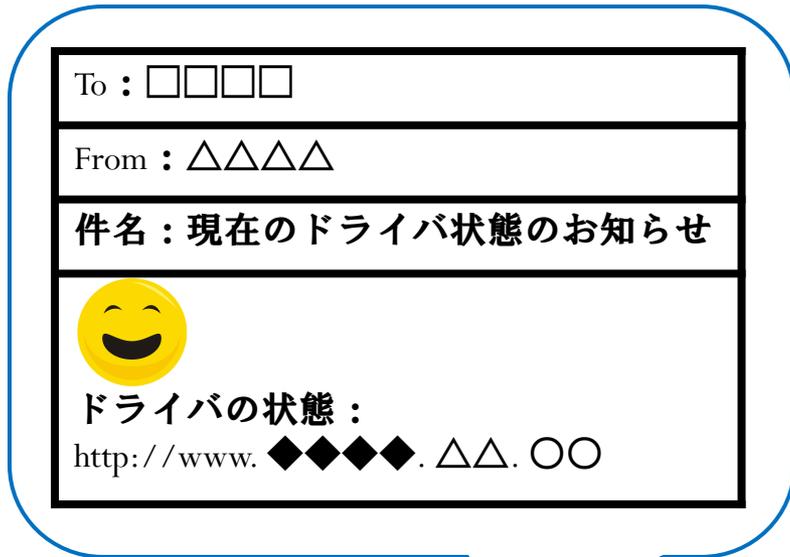
# 緊急時の動作～遠隔操作～

- ホットラインを確保
  - 管理サーバを経由することなく、直接、車両と家庭端末間で通信を行う
  - 遠隔操作によって車両を路肩に寄せる
- ホットラインは暗号化通信によりセキュリティを確保
- ドライバ自身に意識がある場合、ドライバの操作を優先



# メール通知

- 家庭端末、携帯電話へメールを定期的送信
- メールの本文中にドライバの現在の状態を絵文字等で表現
  - ログインすることなく、ドライバの状態を把握可能

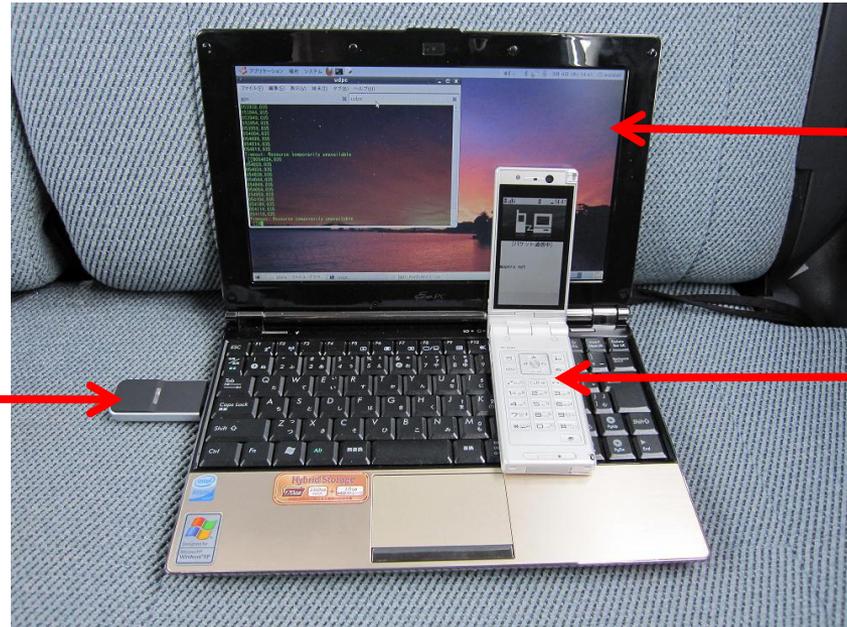


携帯電話



: 定期的に配信されるメール

# 実装



Linux版 PC

NTT docomo  
「P-01A」

(GPSセンサ)  
CanMore社「GT-730F(L)」

## Linux PCで動作するプログラム

- センサ機器の取得情報をファイルへ出力
- 出力したファイルの読み込み、送信用パケットの作成、サーバへ送信

# Google Maps APIで出力した結果（運転時）



# むすび

- まとめ
  - ITSとテレマティクスの概要
  - システムの構成
  - センサデータの送信、閲覧方式
  - 緊急時の動作
  - 実装
- 今後の検討課題
  - 詳細仕様の確定
  - システムの実装と評価



# センサデータ送信用フォーマット

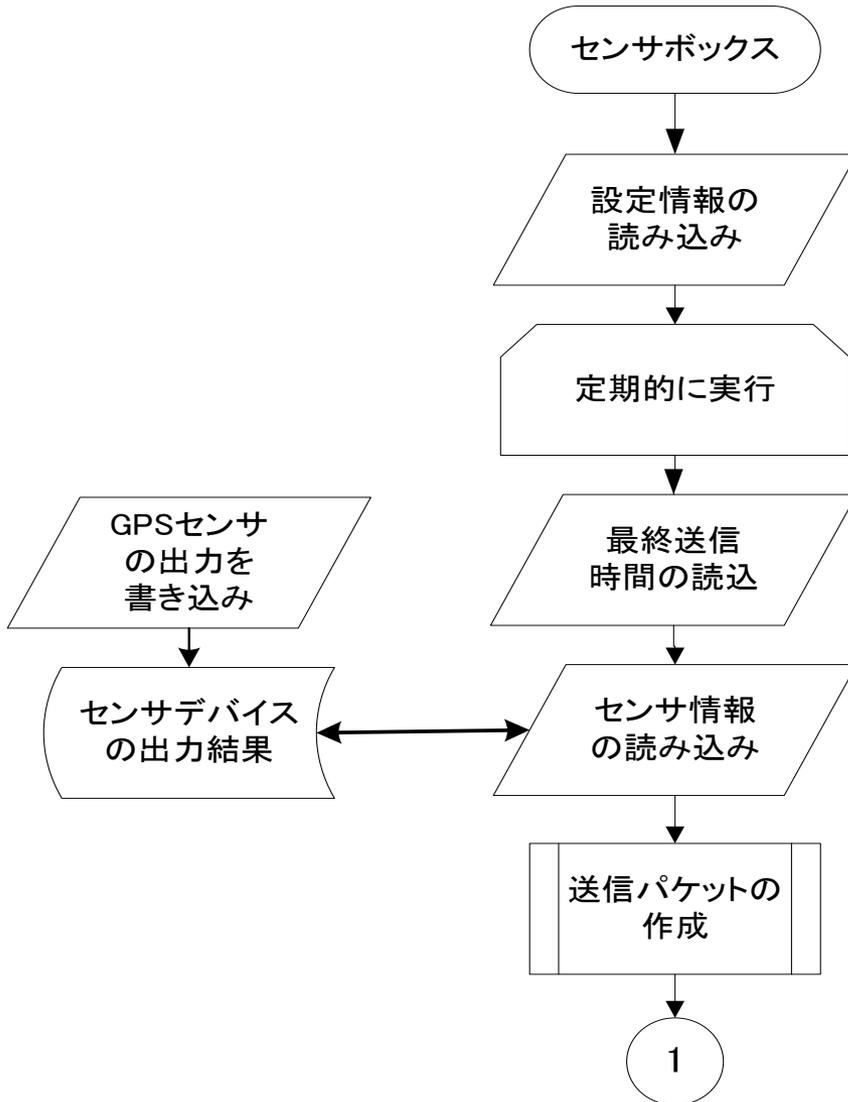
```

<root>
  <user>
</user>
  <sensors>
    <sensor>
      <type>1</type>
      <device>
        :
      </device>
      <data>
        :
      </data>
    </sensor>
    :
  </sensors>
</root>
  
```

- <sensors>→<sensor>→<type>
  - センサデータの種類に応じたID
- <sensors>→<sensor>→<device>
  - デバイス情報
- <sensors>→<sensor>→<data>
  - 取得データ

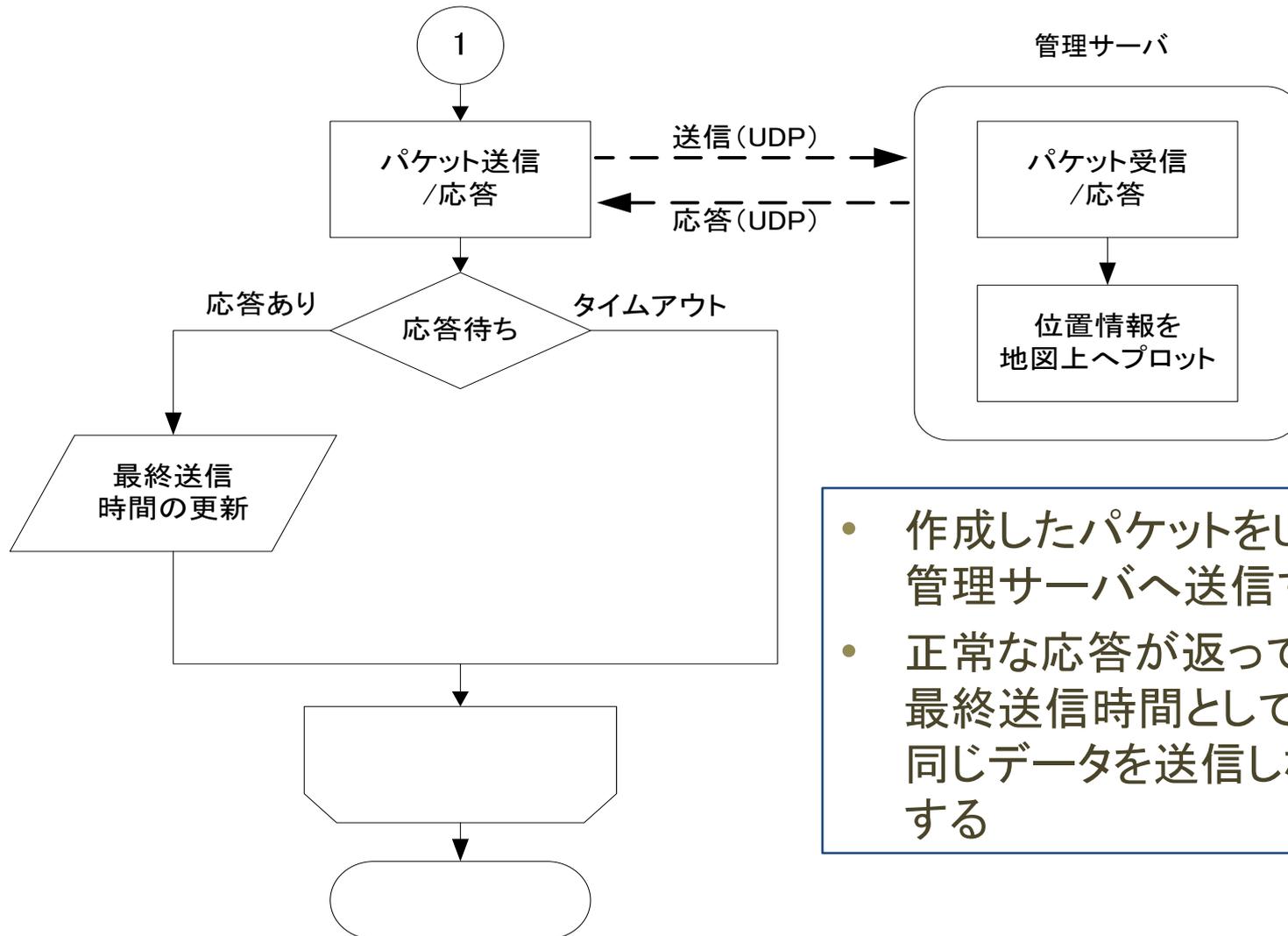
- センサ機器・メーカーに依存せずに利用することができる
- 異なるセンサタイプのデータを、まとめて送信することができる
- 今後のセンサデータの種類が増加しても、柔軟に対応することができる

# センサボックスのモジュール図



- 最終送信時間の値から未送信のデータを読み込む
- センサ機器の取得情報を一定の周期でファイルへ出力する
- センサデータを、送信用パケットフォーマットに変換する

# センサボックスのモジュール図



- 作成したパケットをUDPで管理サーバへ送信する
- 正常な応答が返ってきた場合、最終送信時間として記憶し、同じデータを送信しないようにする

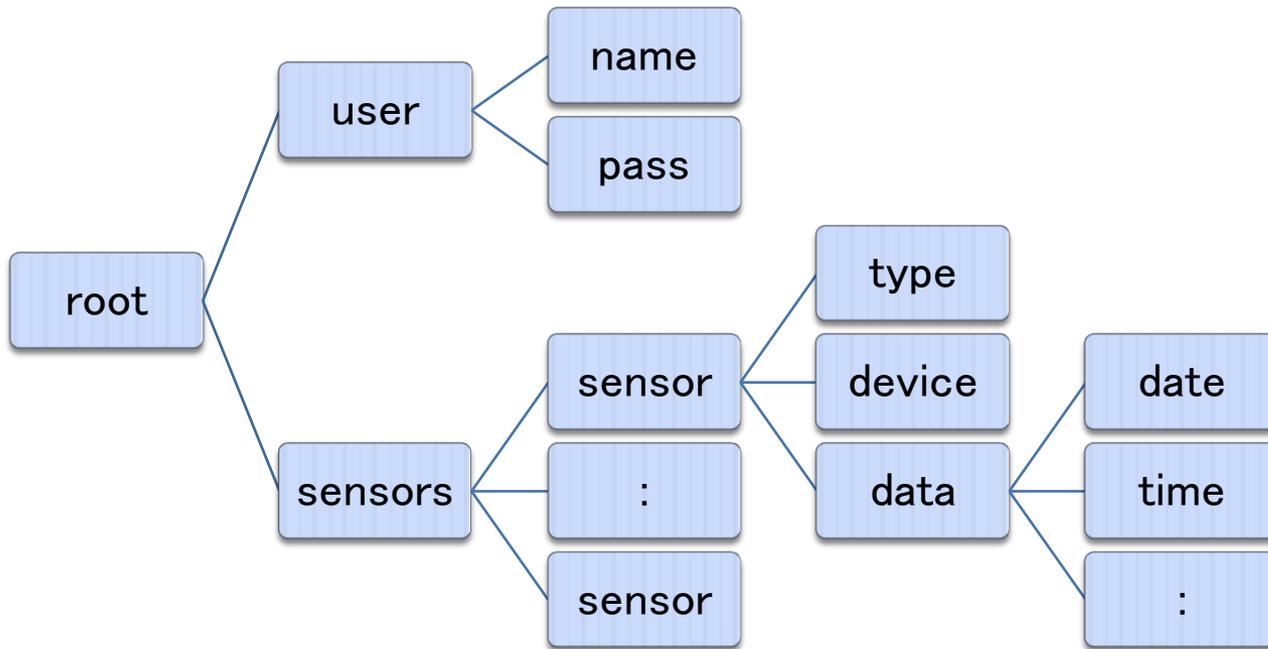
# Google Maps APIで出力した結果（歩行時）



# Google Maps APIで出力した結果（乗車時）



# センサーデータ送信用フォーマット

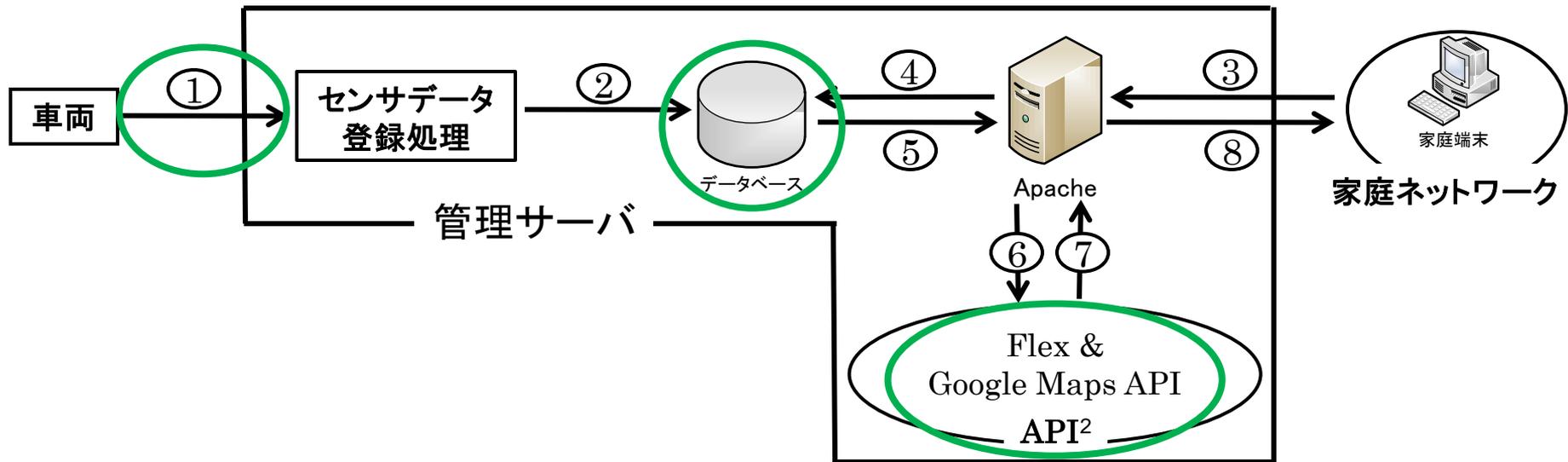


```

<root>
  <user>
    :
  </user>
  <sensors>
    <sensor>
      :
    </sensor>
  </sensors>
</root>
  
```

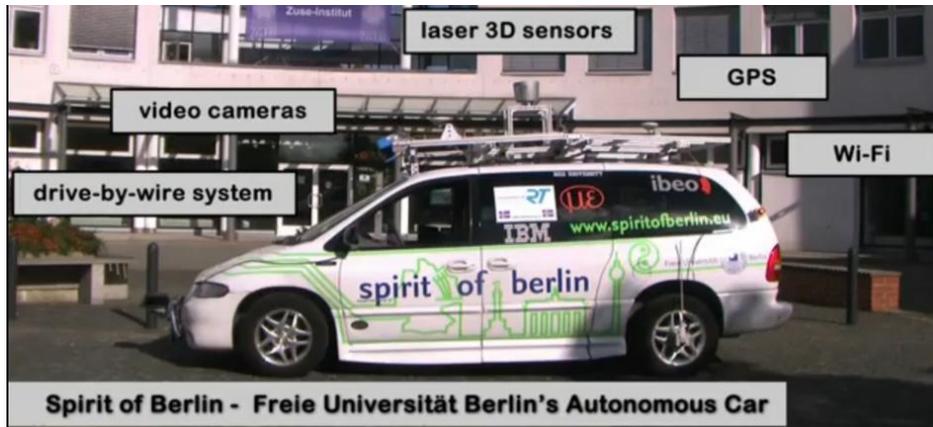
センサーデータ送信用フォーマットは、XML (Extensible Markup Language) 形式で定義する。

# 管理サーバ内の動作



| 動作番号 | 動作内容                              |
|------|-----------------------------------|
| ①    | XML <sup>1</sup> 形式でセンサデータをUDPで送信 |
| ②    | XML解析後、SQLでデータベースに登録              |
| ③、⑧  | センサデータ閲覧要求 / グラフ化データ送信            |
| ④、⑤  | センサデータ読み出し要求 / センサデータ送信           |
| ⑥、⑦  | グラフ化要求 / グラフ化データ送信                |

# iDriver (iPhone remote controlled car)



- ドイツの大学で実装された技術
  - iPhoneで車をリモートコントロールができるアプリケーション
  - Wi-Fiを介してアクセルやブレーキなどの駆動回路を操作
  - ハンドルは加速度センサを利用
  - 車の前部にビデオカメラを設置し、iPhoneにリアルタイムで送信

# iDriver (iPhone remote controlled car)

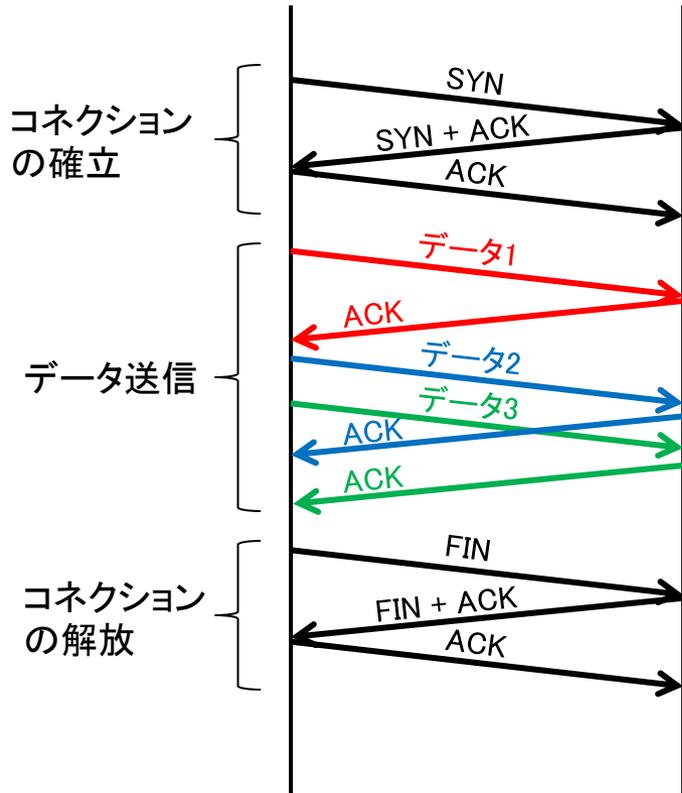


# データベース登録情報

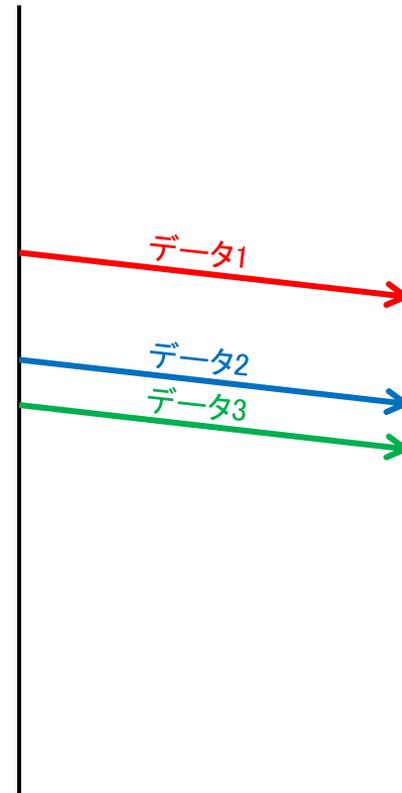
| テーブル名        | 登録情報                    |
|--------------|-------------------------|
| ユーザ情報テーブル    | ユーザIDとパスワードを定義          |
| センサデータ登録テーブル | 取得するセンサデータ毎(脈拍、血圧など)に定義 |
| 測定機器情報テーブル   | 企業毎の製品を定義               |

# TCPとUDPの比較

TCPの場合



UDPの場合



# TCPとUDPのヘッダパケット

## TCPヘッダの場合

|          |    |         |          |
|----------|----|---------|----------|
| 送信元ポート番号 |    | 宛先ポート番号 |          |
| シーケンス番号  |    |         |          |
| 確認応答番号   |    |         |          |
| ヘッダ長     | 予約 | コードビット  | ウィンドウサイズ |
| チェックサム   |    | 緊急ポインタ  |          |
| オプション    |    |         |          |

## UDPヘッダの場合

|          |         |
|----------|---------|
| 送信元ポート番号 | 宛先ポート番号 |
| データ長     | チェックサム  |

# Google Maps APIで出力した結果 (運転時)



# Google Maps APIで出力した結果（歩行時）



# Google Maps APIで出力した結果（乗車時）

