W3C Vehicle API KDDI/ACCESS activity

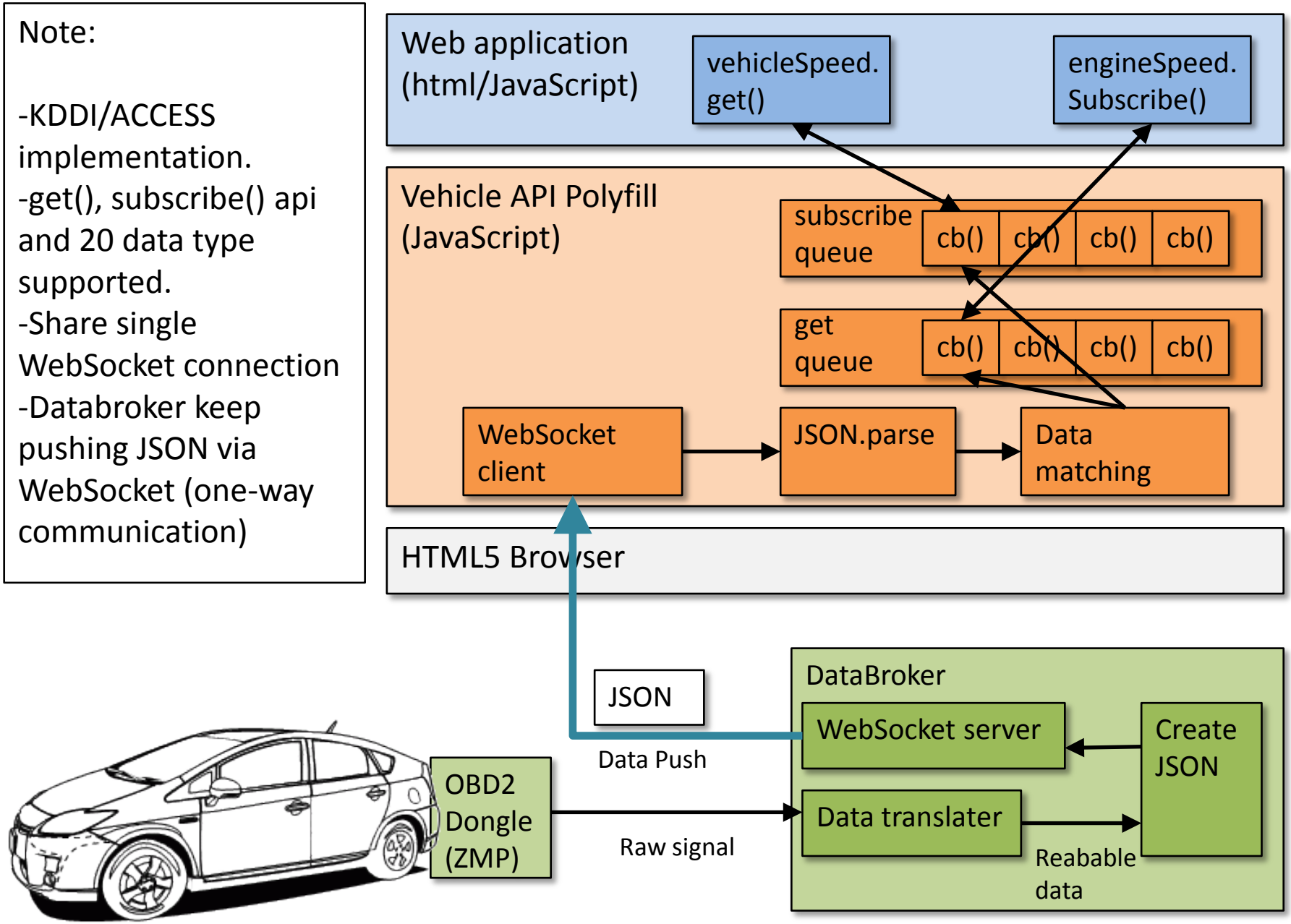
2016 Apr, 28

ACCESS

Vehicle API (polyfill) structure

Note:

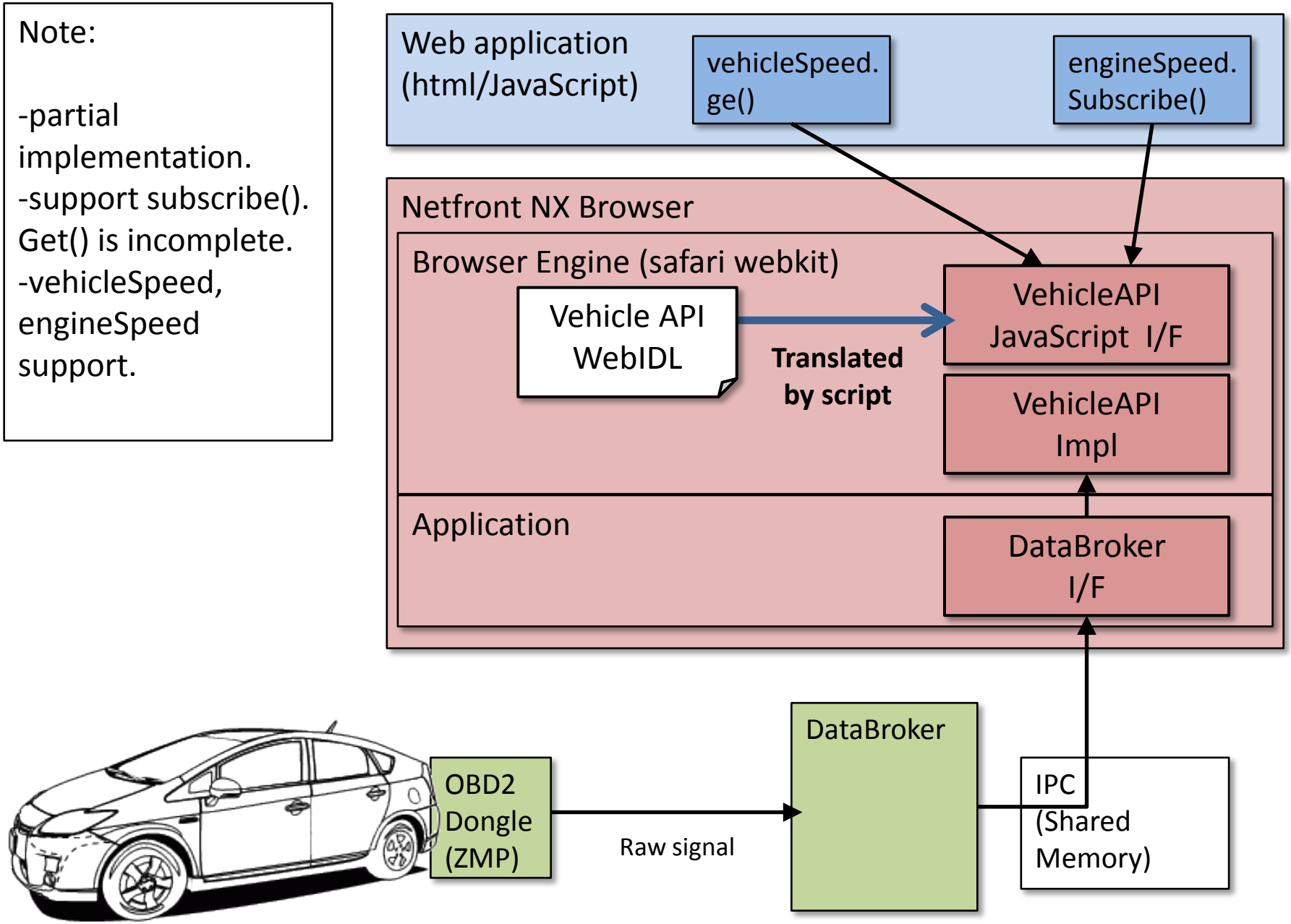
-KDDI/ACCESS implementation.
-get(), subscribe() api and 20 data type supported.
-Share single WebSocket connection
-Databroker keep pushing JSON via WebSocket (one-way communication)

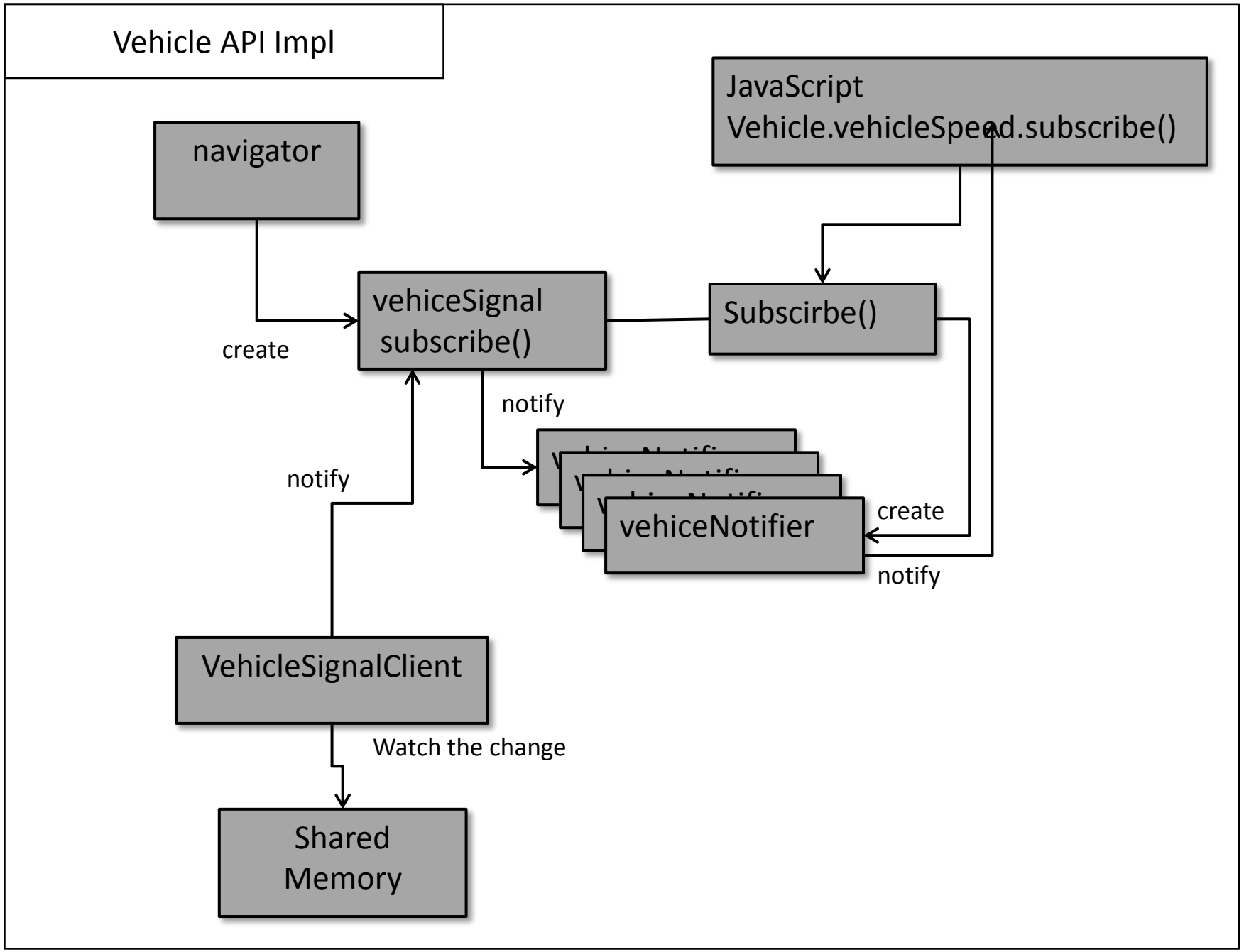


Vehicle API (native) structure

Note:

-partial implementation.
-support subscribe(). Get() is incomplete.
-vehicleSpeed, engineSpeed support.




Vehicle API Impl
JavaScript
Vehicle.vehicleSpeed.subscribe()
navigator
vehiceSignal
subscribe()
Subscirbe()
create
notify
notify
vehiceNotifier
create
notify
VehicleSignalClient
Watch the change
Shared
Memory

Pros Cons

	Pros	Cons
WebIDL (JavaScript API)	-Easy to use for developer -Developer don't have to know how it is implemented (agnostic).	-Need implementation in Browser engine (or maybe Polyfill implementation is possible)
Service based API (Websocket-ish, REST-ish)	-Flexible to create, modify -No need to change Browser engine code	-If using websocket, using many connections consume resources. -Developer may have to take care of connection.

My questions

- What is the strong reason that WebIDL style API is not good and we should choose Service-based API?
- Service-based API should be implemented with REST API(HTTP)? or WebSocket? or both?
- How much part of the spec must be covered to be recognized as a reference implementation? Is there a rule?(in our polyfill, get(), subscribe() and 20 datatypes are supported.)
- How mush spec must be covered by test suite?

Web and automotive hackathon In Tokyo

- Hackathon to utilize W3C vehicle API(FPWD).
- Planned and lead by KDDI-RI.

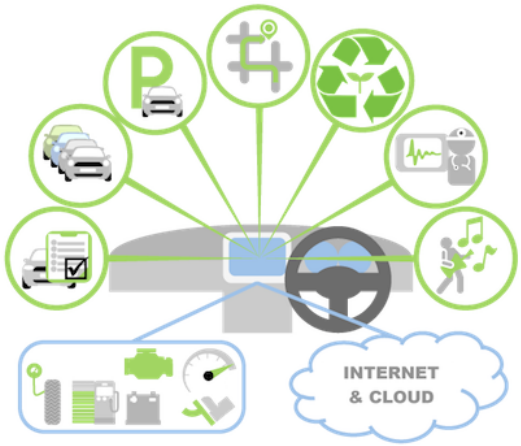
“Webとクルマのハッカソン”

コネクテッド・カー時代におけるWebと車の連携アプリ／サービスを創発しよう！

今世界中でインターネットとつながるクルマが増えつつあるとともに、W3CでもVehicle APIの標準化が進められるなど、クルマがWeb技術を利用する取組みに期待が高まっています。

そこで、本ハッカソンでは自動車の走行状態に関するデータ（車速、アクセル、ブレーキ、エンジン回転数、オドメータ、ハンドル角度、燃費、車両位置、ドア状態、加速度、など）を利用しつつ、HTML5などWebプラットフォームを活用したアプリケーション開発を競っていただきます。

クルマの情報とWeb技術の融合から生まれる新しいサービスやアプリを仲間と考えることで、クルマを取り巻く素晴らしい未来を共に創りましょう！



(http://www.kddi-ri.jp/hackathon/2016/result sorry, all Japanese)

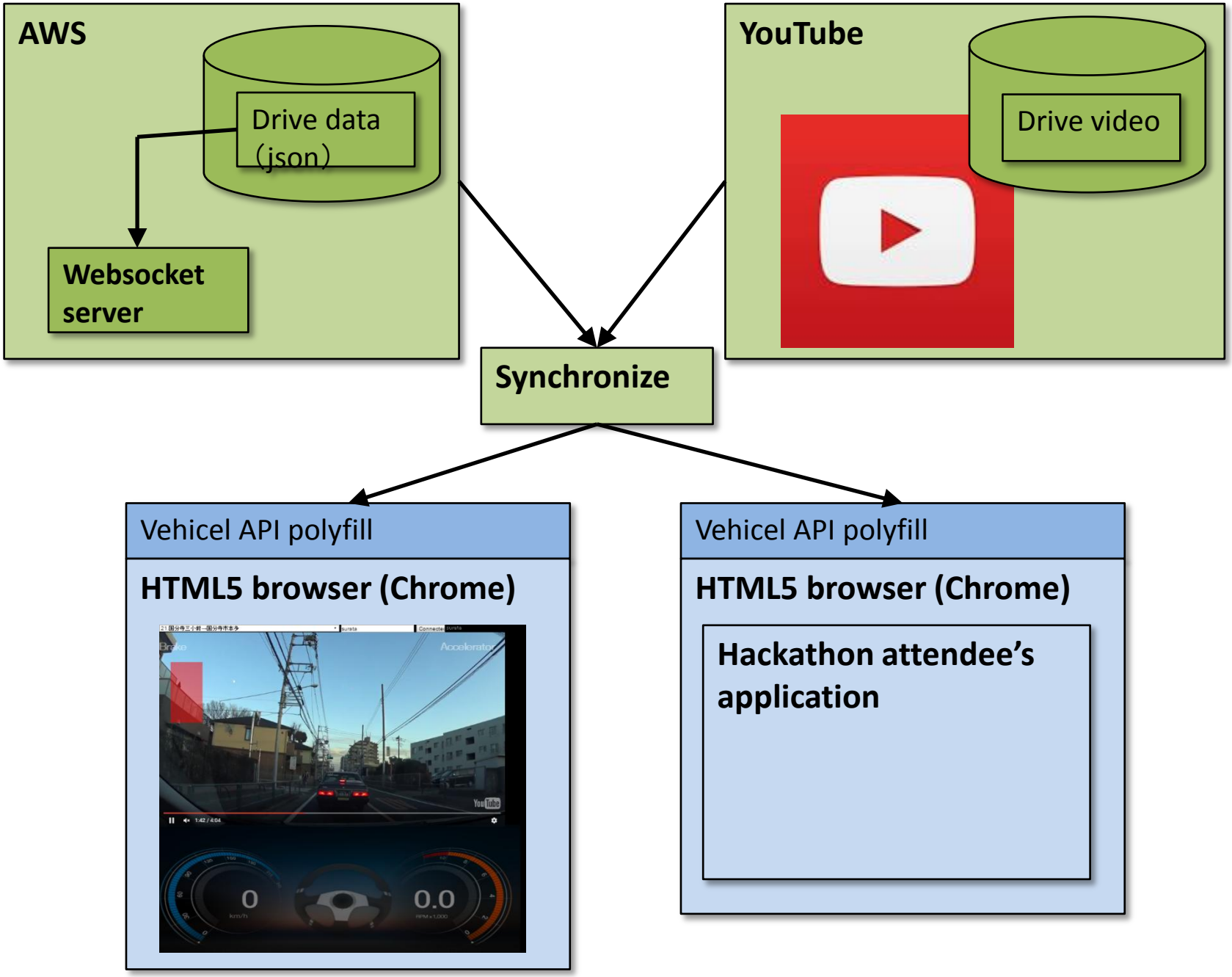
Web and automotive hackathon In Tokyo

- Held on January 30th, 31st, 2016
- 50+ people attended. Tier1 engineers, Web engineers, Students, etc.

- Supported by OEMs(ToyotaITC, Nissan, Honda), Tier1s(Alpine, Pioneer, FujitsuTEN), many other companies and MIC(Ministry of Internal Affairs and Communications)

Deleted

(Because of security reason, we don’t publish this URL. Sorry.)


AWS
Drive data
(json)
Websocket
server
YouTube
Drive video
Synchronize
Vehicel API polyfill
HTML5 browser (Chrome)
Vehicel API polyfill
HTML5 browser (Chrome)
Hackathon attendee’s
application

Thanks!