Connected Vehicle Blueprint

Sponsored by Tencent, Arm, Intel, Nokia

Tencent CSIG Future Network Lab
2018.03.29
Table of Contents

- General Blueprint Introduction
- Blueprint Use Cases
- Network Architecture
- MEC Architecture/Deployment/Major Components
- Future Plan
General Blueprint Introduction

- Connected Vehicle Blueprint focuses on establishing an open source MEC platform, which is the backbone for V2X Application.
- Connected Vehicle Blueprint was established in the Akraino Community on March 14th, 2019. The first release is targeted for NOV 30, 2019.
- Connected Vehicle Blueprint is sponsored by Tencent, Arm, Intel, Nokia.
- Contact: Robert.Qiu (robertqiu@tencent.com)
- Refer to: https://wiki.akraino.org/display/AK/Connected+Vehicle+Blueprint
## Blueprint Use Cases

<table>
<thead>
<tr>
<th>Use Cases</th>
<th>Value Proposition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accurate Location</td>
<td>Accuracy of location improved by over 10 times than today’s GPS system. Today’s GPS system is around 5-10 meters away from your real location, &lt;1 meter is possible with help of edge compute.</td>
</tr>
<tr>
<td>Smarter Navigation</td>
<td>Real-time traffic information update, reduce the latency from minutes to seconds, figure out the most efficient way for drivers.</td>
</tr>
<tr>
<td>Safe Drive Improvement</td>
<td>Figure out the potential risks which can NOT be seen by the driver. See below.</td>
</tr>
<tr>
<td>Reduce traffic violation</td>
<td>Let the driver understand the traffic rule in some specific area. For instance, change the line prior to narrow street, avoiding opposite way drive in the one way road, avoiding carpool lane when single driver etc.</td>
</tr>
</tbody>
</table>

### Location Accuracy

![GNSS Diagram](image1.png)

### Smarter Navigation

![Traffic Information Update](image2.png)

### Safe Drive Improvement

![V2X Message](image3.png)

### Reduce traffic violation

![Smart Edge System](image4.png)
UPF is disaggregated from core network, which is closer to end users.
MEC Software Architecture

MEC Edge Cluster

- Application run in Bare metal
  - Vehicle Application
  - Tars Agent
  - Linux OS
  - NIC

- Application run in VM
  - Vehicle Application
  - Tars Agent
  - Guest OS
  - Hyper-V
  - Host OS
  - NIC

- Application run in Container
  - Vehicle Application
  - Tars Agent
  - Linux OS
  - NIC

5G CP

Dispatch Policy Configure

MEC Engine

Data Distribution (Configurable Policy)

Packet Send/Receive

Tars Master

Host OS

Hyper-V

UPF

Control Flow

Data Flow

Core Network

DC
OpenNESS is an open source reference toolkit to develop, securely on-board and manage new edge services on Network Edge & On-Premise.

More details @ https://www.openness.org/

* Other names and brands may be claimed as the property of others
Major Components - Tars Platform

Service Management

- Service Register/Discovery
- Service Configuration
- Set Mode
- Metric Monitor
- Log Aggregation
- Distributed Trace
- Zone Optimization
- Flow Control/Load Balance/Disconnect Protect

RPC

- Synchronize
- Asynchronous
- Response
- None-Response

Mode

Protocol

- TARS
- SSL
- PB
- Http1/2
- Third Party

DevOps

- CI/CD
- Gated Launch
- Auto Test
- Auto Deployment
- Monitor

M-PL

- C++
- JAVA
- Nodejs
- PHP
- GO

Infrastructure (Physical Machine, Virtual Machine, Container)
Major Components - Tars Platform

2007 Incubation

2008 Product

- Scale-out Improvement
- Performance optimization
- HA optimization

2017

- GitHub Release
  https://github.com/TarsCloud/Tars

2017

2018

- JAVA
- C++

2018

2018

2018

- Linux Foundation Project

2018

2019

- .NET
- Python
- Service Mesh

Adopted by

Tencent, VIVO, Iflytek, China Literature Limited et al

Use Cases

- Access Business: 300+
- Access Service: 20,000+
- Server Number: 100,000+
- Visitor per day: 1 Billion+

Developers

More than 8000 Developers

Developers

Game

Connected-Vehicle

Finance

Live Video

Scale-out Improvement

Performance optimization

HA optimization

GitHub Release

https://github.com/TarsCloud/Tars

JAVA

C++

PHP

NodeJS

GO

.NET

Python

Service Mesh
Future Plan

• The first demo will be rolled out in Tencent Cloud Innovation Conference, May 22, 2019.

• The first Akraino version will be released in NOV 30, 2019.

• More will come soon… Stay tuned!
### Appendix: Connected Vehicle Blueprint Criteria

<table>
<thead>
<tr>
<th>Case Attributes</th>
<th>Description</th>
<th>Informational</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Type</strong></td>
<td>New Blueprint for the Edge</td>
<td></td>
</tr>
<tr>
<td><strong>Blueprint Family - Proposed Name</strong></td>
<td>It is a independent blueprint, NOT a blueprint family yet.</td>
<td></td>
</tr>
<tr>
<td><strong>Use Case</strong></td>
<td>MEC platform used for Connected Vehicle.</td>
<td></td>
</tr>
<tr>
<td><strong>Blueprint proposed Name</strong></td>
<td>Connected Vehicle Blueprint</td>
<td></td>
</tr>
<tr>
<td><strong>Initial POD Cost (capex)</strong></td>
<td>The Minimum Configuration: 4 Servers in total MEC Platform(1 Server) + 1 App Server(1 Server) + 2 Simulators(2 Server)</td>
<td></td>
</tr>
<tr>
<td><strong>Scale &amp; Type</strong></td>
<td>Up to 4 Arm/X86 server</td>
<td></td>
</tr>
<tr>
<td><strong>Applications</strong></td>
<td>The MEC platform which can be used to connect vehicles, the general data flows are itemized below:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1) Grab the traffic/vehicle information</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2) Dispatch the traffic/vehicle information to the corresponding edge process unit. Note well: The dispatch policy can be configurable.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3) Process the data in the Edge or Cloud and figure out the suggested action item for the vehicle driver</td>
<td></td>
</tr>
<tr>
<td></td>
<td>4) Send the suggested action items to the vehicle driver</td>
<td></td>
</tr>
<tr>
<td><strong>Power Restrictions</strong></td>
<td>Less than 6KW.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>The Maximum Power consumption for each server is around 1500W, 1500 * 4 = 6000W</td>
<td></td>
</tr>
<tr>
<td><strong>Infrastructure orchestration</strong></td>
<td>○Docker + K8s</td>
<td></td>
</tr>
<tr>
<td></td>
<td>○VM and OpenStack/StarlingX</td>
<td></td>
</tr>
<tr>
<td><strong>PaaS</strong></td>
<td>Tars</td>
<td></td>
</tr>
<tr>
<td><strong>Network</strong></td>
<td>OVS, DPDK, VPP</td>
<td></td>
</tr>
<tr>
<td><strong>Workload Type</strong></td>
<td>Bare metal, VM, Container</td>
<td></td>
</tr>
<tr>
<td><strong>Additional Details</strong></td>
<td>OpenNESS</td>
<td></td>
</tr>
</tbody>
</table>
## Appendix: Assessment Criteria

<table>
<thead>
<tr>
<th>Criteria</th>
<th>Connected Vehicle Blueprint</th>
</tr>
</thead>
<tbody>
<tr>
<td>Each initial blueprint is encouraged to take on at least two committers from different companies</td>
<td>Tencent, Arm, Intel, Nokia</td>
</tr>
<tr>
<td>Complete all templates outlined in this documents</td>
<td>Detailed in this slide</td>
</tr>
<tr>
<td>A lab with exact configuration required by the blueprint to connect with Akraino CI and demonstrate CD. User should demonstrate either an existing lab or the funding and commitment to build the needed configuration.</td>
<td>A test and simulation lab will be provided in Tencent Cloud Silicon Valley.</td>
</tr>
<tr>
<td>Blueprint is aligned with the Akraino Edge Stack Charter</td>
<td>All opensource, Edge use case, Aligned with the Akraino Charter</td>
</tr>
<tr>
<td>Blueprint code that will be developed and used with Akraino repository should use only open source software components either from upstream or Akraino projects.</td>
<td>Yes, all open source.</td>
</tr>
<tr>
<td>For new blueprints submission, the submitter should review existing blueprints and ensure it is not a duplicate blueprint and explain how the submission differs. The functional fit of an existing blueprint for a use case does not prevent an additional blueprint being submitted.</td>
<td>A edge platform for deploying connected vehicle application does not exist in Akraino yet.</td>
</tr>
</tbody>
</table>

### Criteria

| Criteria of the project is appropriate (no trademark issues etc.); Proposed repository name is all lower-case without any special characters. | Connected Vehicle Blueprint |
| Project contact name, company, and email are defined and documents | Robert Qiu, Tencent robertqiu@tencent.com |
| Description of the project goal and its purpose are defined. | Establishing an MEC edge platform for connected vehicle use cases. |
| Scope and project plan are well defined. | Target for Release2, 30 July, 2019. |
| Resource committed and available | There is a team, resources and lab in place. |
| Contributors identified | Tencent, Arm, Intel, Nokia |
| Initial list of committers identified (elected/proposed by initial contributors) | Tencent, Arm, Intel, Nokia |
| Meets Akraino TSC policies | The project will operate in a transparent, open, collaborative, and ethical manner at all the times. |
| Proposal has been socialized with potentially interested or affected projects and/or parties | • Have already reached a consensus with sponsors. • Talk with chair/co-char |
| Cross Project Dependencies. | OpenStack, K8s, Docker, DPDK, OpenNESS, OVS et al. |
Thanks