Conceptual Architecture

Group 1 - HexTech
Presenter: E Ching Kho (Noon) and Fuwei Zhuang (Elina)

Video Presentation: https://youtu.be/kgDoH_HyrRE

Team Members

Benjamin Hui

Role: Group Leader, Report (Abstract, Introduction & Overview, Version Control, Data Dictionary)

Yixin Su

Role: Report (Subsystems Breakdown, Responsibilities, Lessons Learned)

Zewen Zheng

Role: Report (Subsystems Breakdown, Conclusion, References)

Fuwei Zhuang

Role: Report (Use Case 1, Subsystems Breakdown), Presenter, Video Editor

E Ching Kho

Role: Report (Architecture styles), Presenter, Presentation Creator

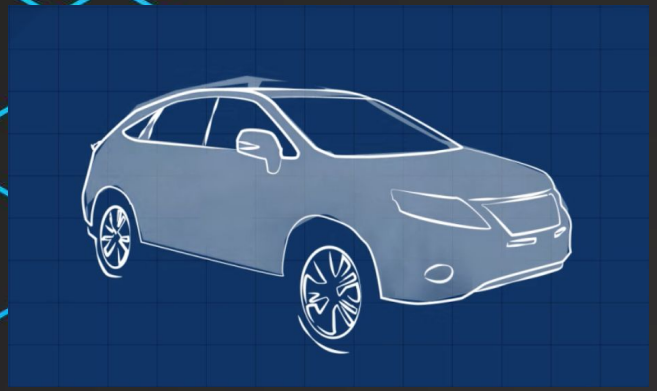
Ruiyang Su

Role: Report (Use Case 2, Subsystems Breakdown)

Report Link: https://docs.google.com/document/d/1UHGi-gtUFRV183ZPcmrYb2zuR-PvqD81nQ5lyotiyGg/edit?usp=sharing

Introduction

Source: https://youtu.be/EY3yVgLecf0



Level 1 - Driver Assistance

Level 2 - Partial Automation

Level 3 - Conditional Automation

Level 4 - High Automation

Level 5 - Full Automation

A high performance, flexible architecture which accelerates the development, testing, and deployment of Autonomous Vehicles

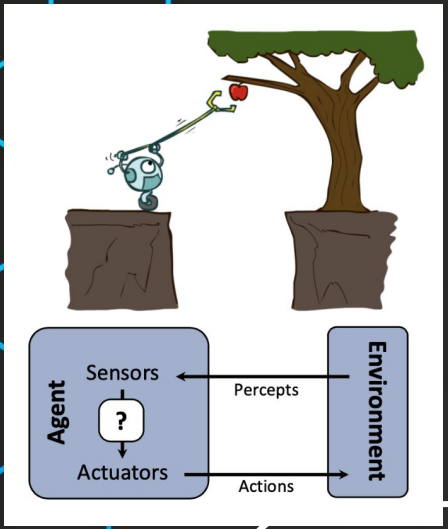
- Owned by Baidu
- Open Source (Apache-2.0)
- Since 2017 (version 7.0)

Link: https://github.com/ApolloAuto/apollo

Derivation Process

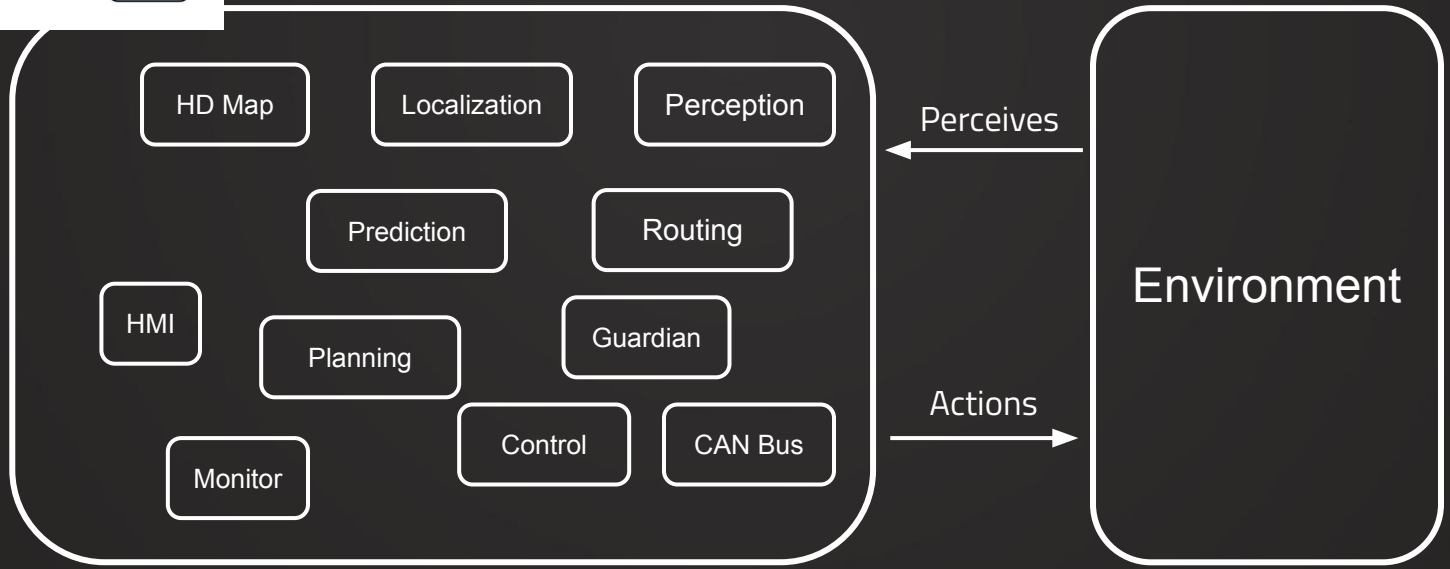
01 The Making of Components

02 Architecture Styles



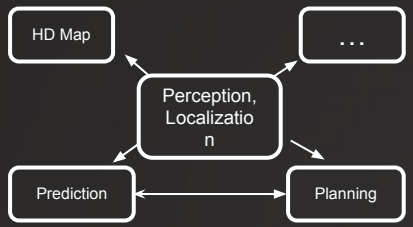
The Making of Components

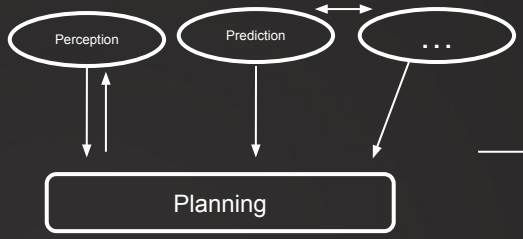
Purpose of AI: Maximize your expected utility

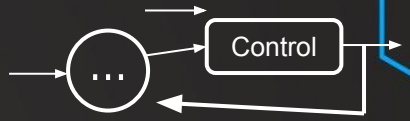


Architecture styles

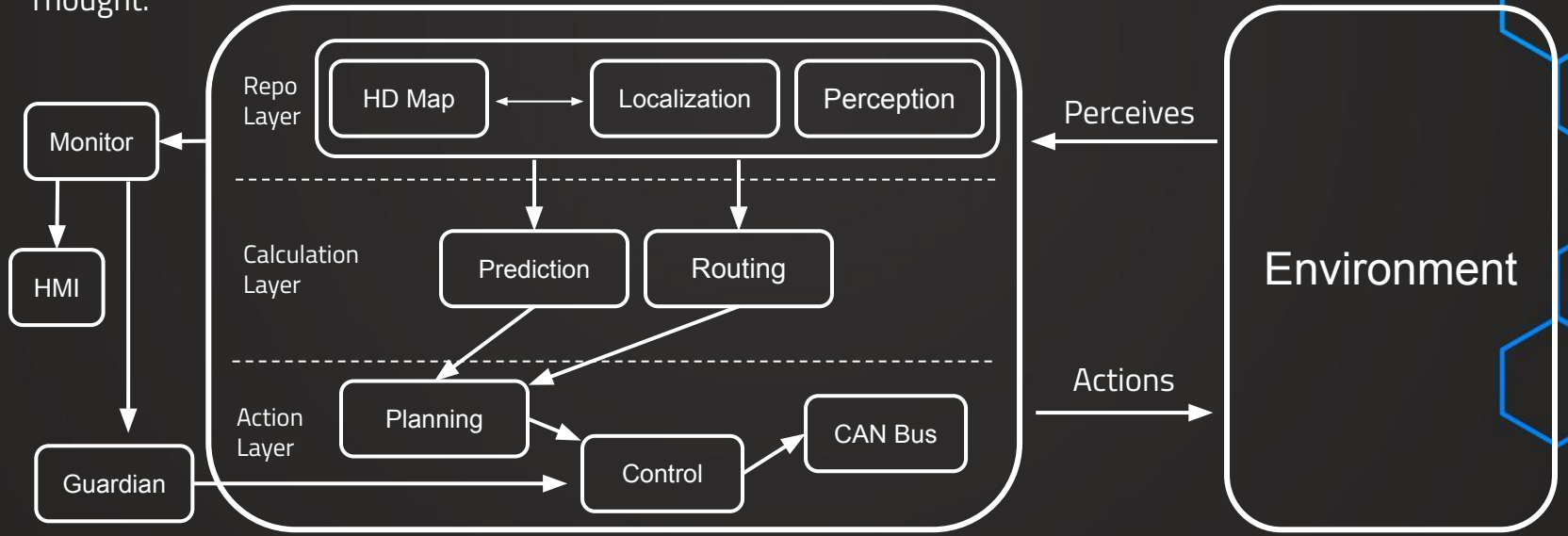
Previous thoughts:







Current Thought:



Conceptual Architecture

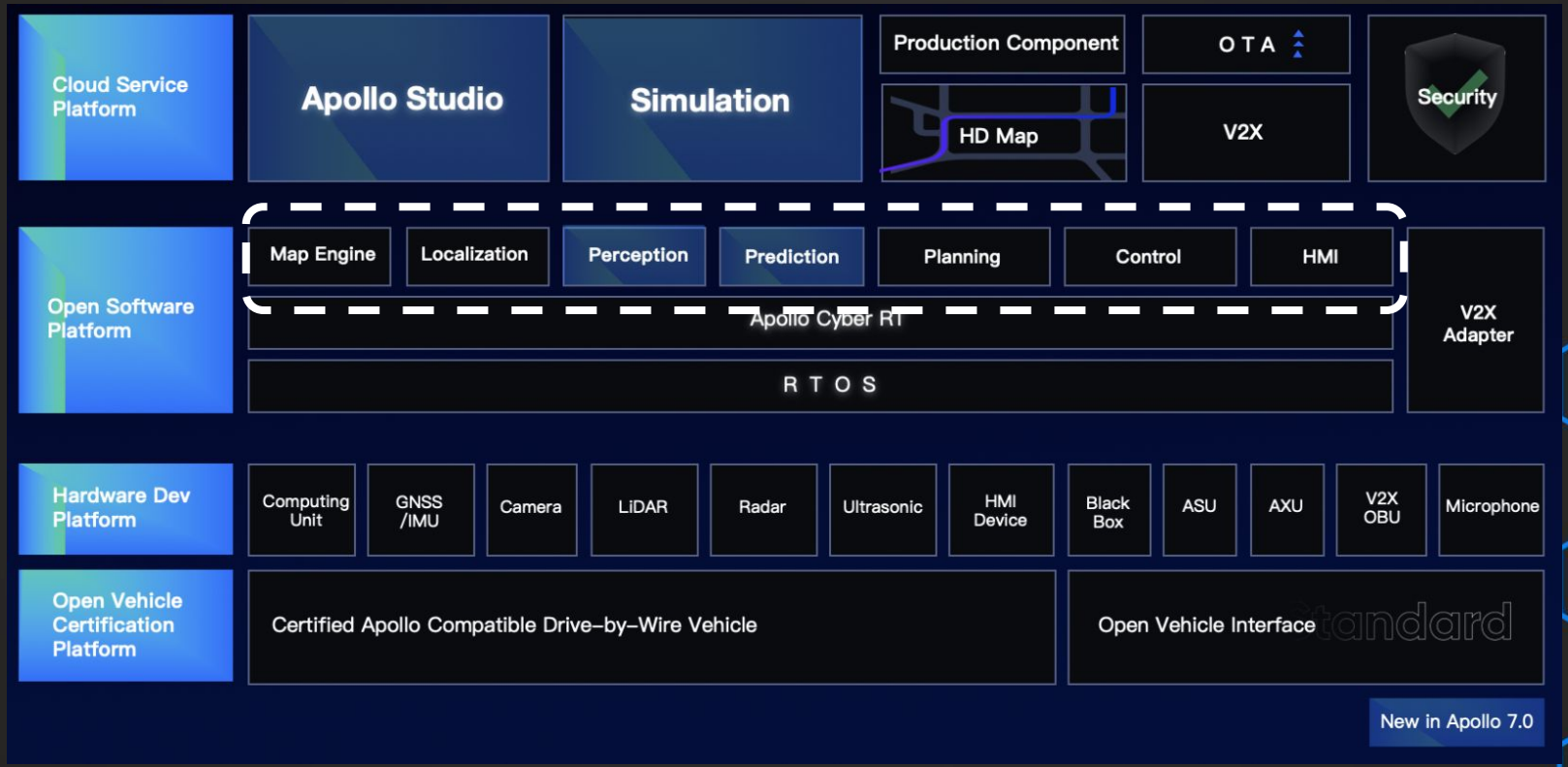
01 **Apollo**
Architecture Overview

02 **Architecture**
Subsystems breakdown & Interactions

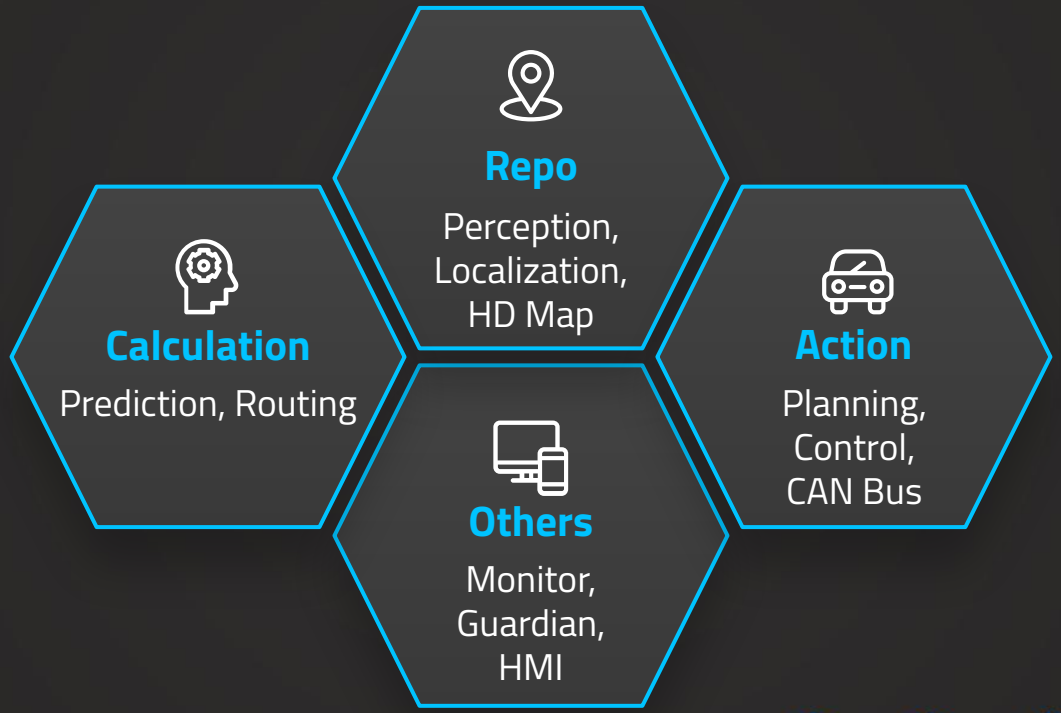
03 **Use Cases**

04 **Versions Evolution**

Conceptual Architecture Overview



Architecture Subsystems Breakdown & Interactions



Architecture - Repository Layer

Perception

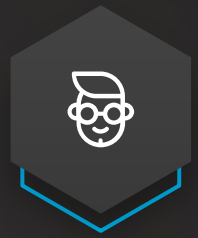
Identifies the world surrounding the autonomous vehicle

- Obstacle detection
- Traffic light detection

Localization

Estimate where the autonomous vehicle is located

- GPS
- IMU
- LiDAR



HD Map

Frequently functions as a query engine support to provide ad-hoc structured information regarding the roads

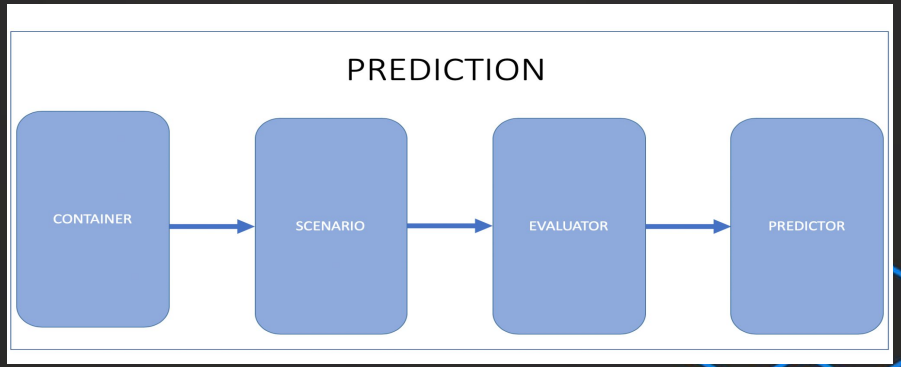
Architecture - Calculation Layer

Prediction

Anticipates the future motion trajectories of the perceived obstacles

Routing

How to reach its destination from current position via a series of lanes or roads



Architecture - Action Layer

Planning

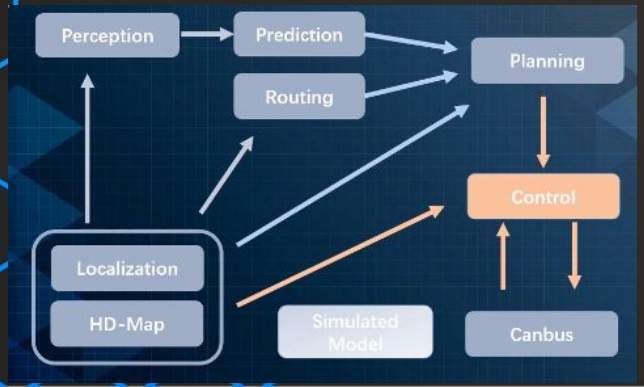
Plans the spatio-temporal trajectory for the autonomous vehicle to take

Control

Executes the planned trajectory by generating control commands

CAN Bus

Interface that passes control commands to the vehicle hardware.



Architecture - Others

Monitor

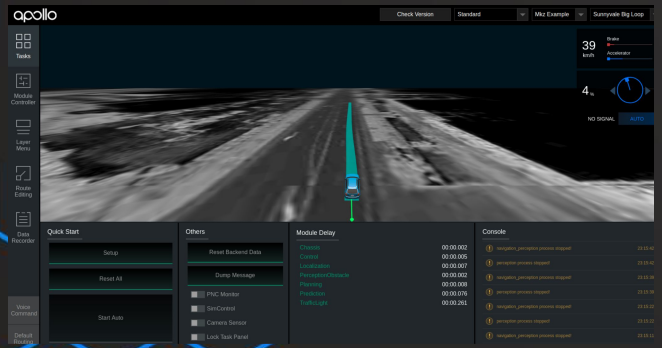
The surveillance system of all the modules in the vehicle including hardware

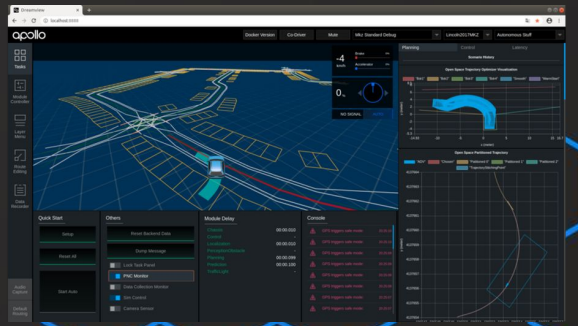
Guardian

Safety purpose module that performs the function of an Action Center and intervenes when Monitor detects a failure

HMI

Web APP for viewing the status of the vehicle and controlling functions of the vehicle in real-time





Use Cases

01

Use case 1

Automatic Rerouting

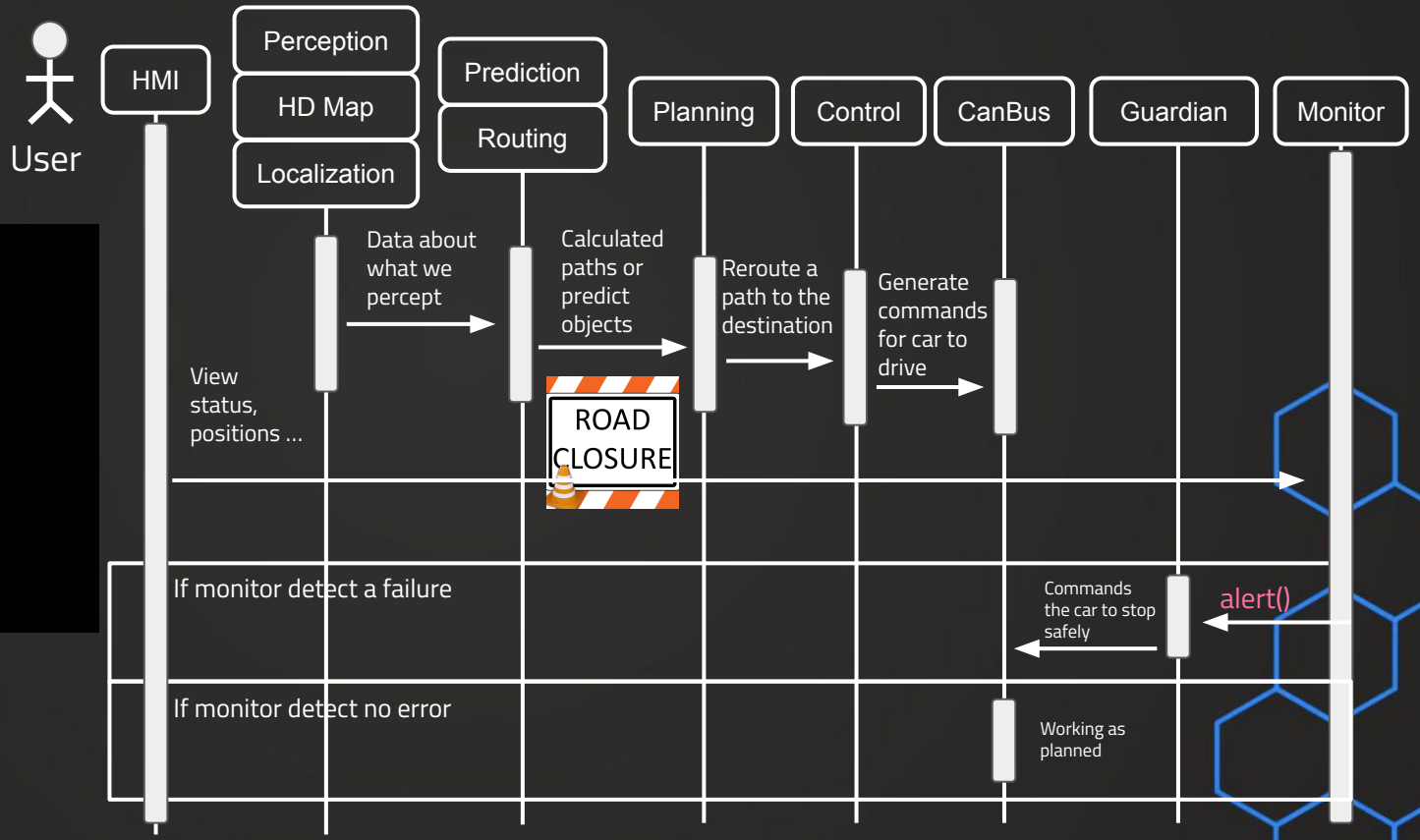
02

Use case 2

Automated Valet Parking

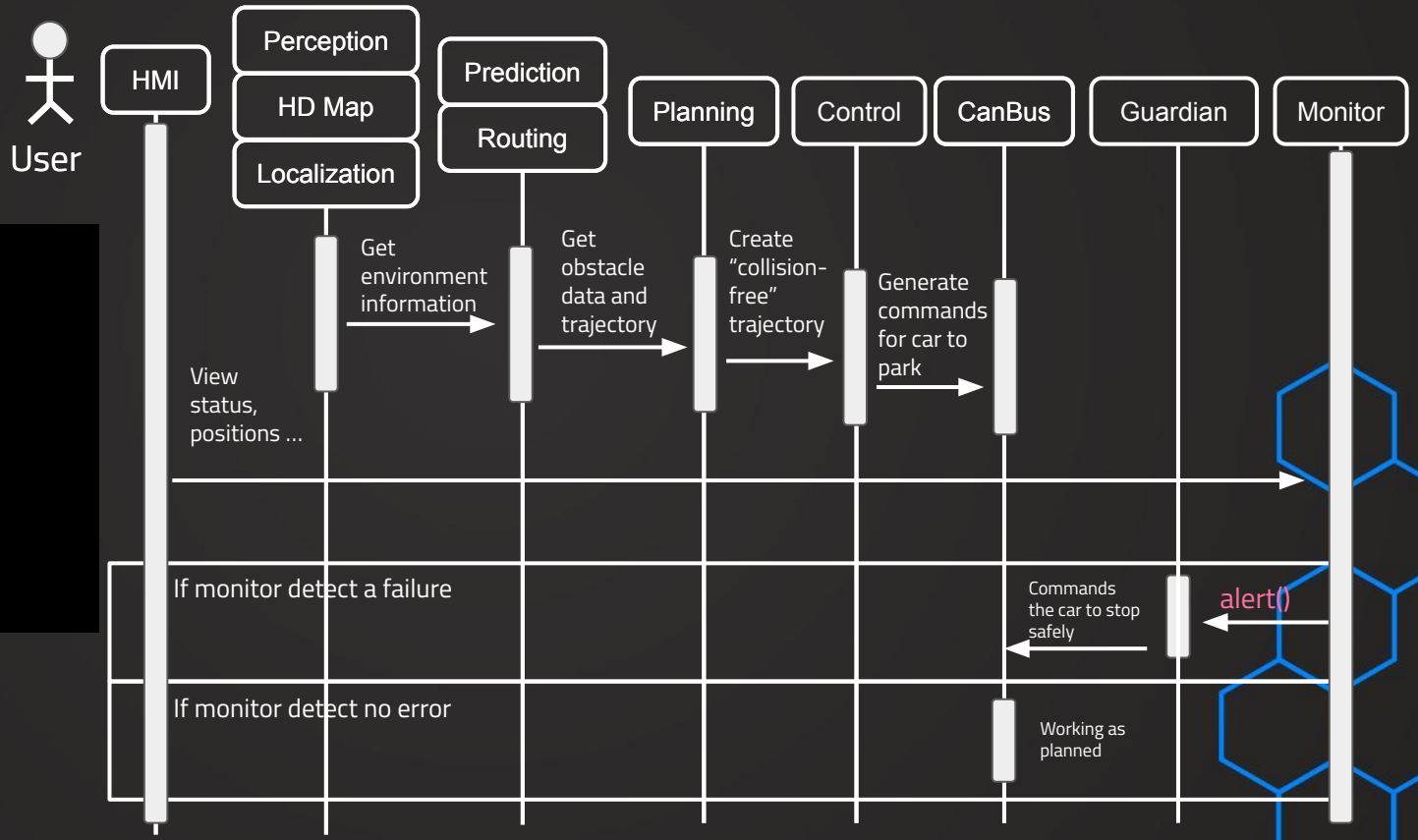
Use Case 1 - Automatic Rerouting

Simplified sequence diagram, view details

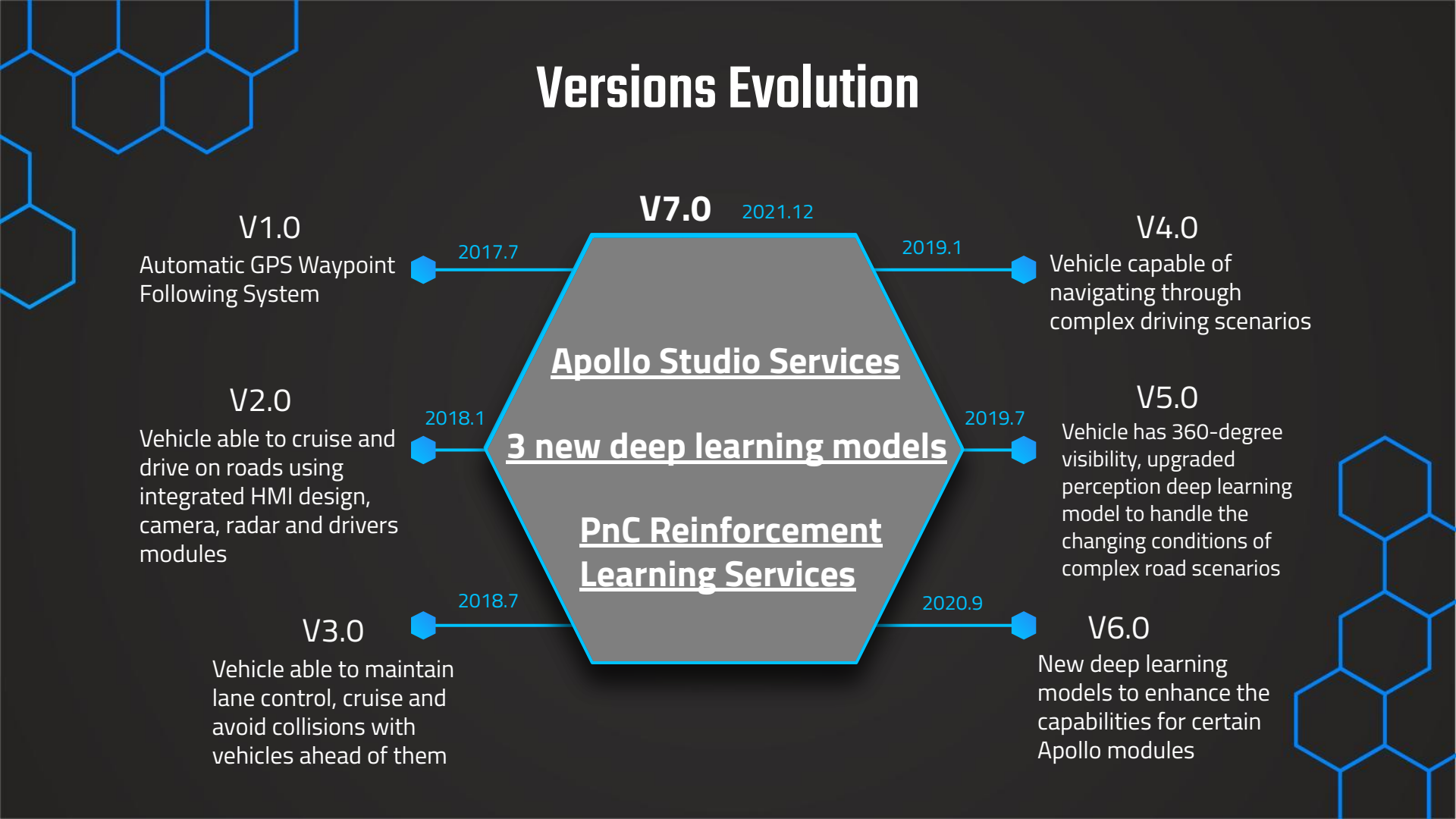


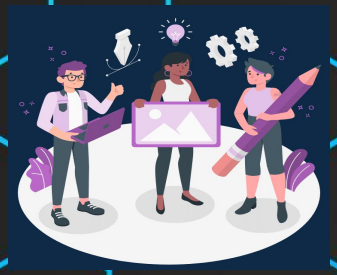
Use Case 2: Automated Valet Parking

Simplified sequence diagram, view details

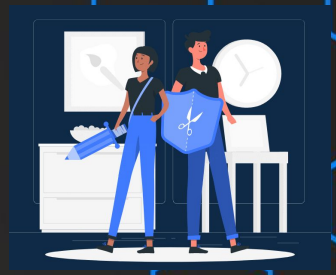


Versions Evolution





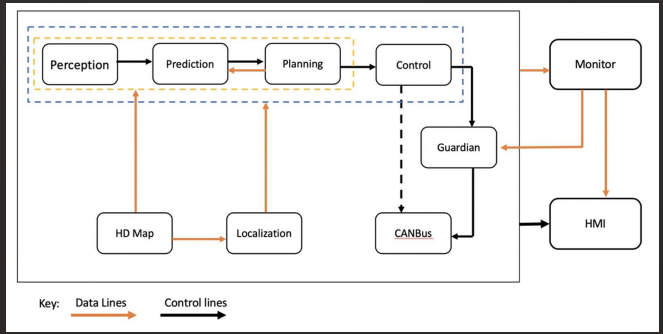
Lessons Learned



- Knowledge on Autonomous Driving (Architecture, components, functionality, components interactions)
- Importance of cooperation, brainstorm ideas and solutions can be achieved much faster (if performed correctly)
- The effectiveness of running concurrency
- Work distribution
- Time management

Conclusion

- We believe that currently the 12 modules components are sufficient enough to perform accurate autonomous driving
- We believe that the architecture styles are a combination of Pipe & Filter, Pub & Sub, Process Control, Client/Server, Repository, Interpreter
- We believe the effect of concurrency for each layer can fasten the process of vehicle execution
- We believe the interactions and data flows are based on this diagram



References

- Apollo Auto. “Apollo 5.0 Technical Deep Dive.” Medium, Apollo Auto, 3 July 2019, https://medium.com/apollo-auto/apollo-5-0-technical-deep-dive-d41ac74a23f9.
- “Apollo Governance.” Apollo, https://apollo.auto/docs/manifesto.html.
- ApolloAuto. “ApolloAuto/Apollo: An Open Autonomous Driving Platform.” GitHub, https://github.com/ApolloAuto/apollo.
- Ma, Changjie. “Baidu ApolloHD Map - UN-GGIM.” Intelligent Transportation and Autonomous Vehicles, https://ggim.un.org/unwgic/presentations/2.2_Ma_Changjie.pdf.
- Swords, Simon. “Software Development Team Roles and Responsibilities: Atlas.” Atlas Computer Systems Ltd, 19 Jan. 2020, https://www.atlascode.com/blog/software-development-project-roles-and-responsibilities/#SOFTWARE_DEVELOPERS.
- WEI, Hao. “How Baidu Apollo Builds HD (High-Definition) Maps for Autonomous Vehicles.” Medium, Towards Data Science, 27 Oct. 2021, https://towardsdatascience.com/how-baidu-apollo-builds-hd-high-definition-maps-for-autonomous-vehicles-167af3a3fea3.