

## Pony.ai基础架构做什么？



## Pony.ai 的基础架构的总体挑战

公司当前目标

- 构建自动驾驶测试车队

车队数量

- $X \rightarrow 10X$

基础架构的挑战

- 技术栈的**可扩展性**
  - 车辆数量增长
  - 运营区域扩大
  - 数据量增长
  - 工程师数量增长
  - 模块数量增长
  - 内部代码量增长
  - ...



# 自动驾驶车载系统



# 车载系统需要做什么？

- 多模块的调度运行
- 模块间的消息通信
- 车载计算资源的分配与管理
  - 内存 / CPU / GPU / ...
- 日志记录
- 监控与报警
- ...

# 车载系统的挑战

- 可靠性
  - 对内存泄露、代码逻辑错误等严重问题容忍度低
    - 影响车队运营效率
  - 可靠的系统监控与异常报警
- 高性能
  - 模块间通信的数据量大（1080P摄像头\*6）
  - 通信延迟的要求低
- 灵活性
  - 支持多种不同类型的计算资源的接入
  - 支持不同类型模块的接入



# 车载系统的实践

## 可靠性

- 代码质量高要求
  - Code review
  - Unit test
- 合理使用工具帮助发现问题
  - 静态分析
  - ASAN
- 多重系统可靠性检查
  - 系统启动前校验
  - 系统运行时实时监控
  - 系统运行后数据分析

# 车载系统的实践

## 高性能

- 合理的架构避免大数据拷贝等严重影响性能的逻辑
- 依据模块逻辑分配合适的计算资源
  - 内存 / CPU / GPU / ...
- 定期对整个系统Profile分析系统的性能瓶颈

## 灵活性

- 定义足够通用的模块公共接口
- 定义足够通用的消息通信接口

# 仿真平台

## 为什么需要仿真？

- 低开销
  - 无需路测和人力介入来评价系统的性能变化
- 低风险
  - 避免了路测不稳定的新算法/功能时带来的风险
- 提供了基于数据驱动快速迭代算法的可行性
  - 新算法可以在仿真平台上快速评估





# 仿真平台的挑战与实践

- 仿真结果的可靠性
  - 在服务器环境模拟车载环境
  - 合理建模车辆的动力学模型
- 仿真数据的选择与管理
  - 选择合适的路测数据帮助算法的迭代
  - 规范地依据类别管理大规模的仿真数据
- 仿真系统的性能
  - 将仿真并行部署在分布式计算平台

# 数据基础架构

## 数据是自动驾驶技术进步的核心驱动力

- 数据存储与访问
- 数据处理
- 数据同步

## 核心挑战

- 数据量大
  - PB级数据  
(1080P摄像头\*6\*8hour\*200day\*100vehicles)
- 数据属性不同于互联网数据
  - 客户端产生
  - 大量传感器数据
  - 大量模块运行日志



# 数据存储的挑战

- 依据特定的使用场景设计合理的存储格式的设计
  - 便于车载系统记录
  - 便于大规模数据分析
  - 便于部分数据访问
  - 便于文件系统存储
- 选择合适的存储系统
  - 针对冷/热数据选择不同方案
  - 选择高可用的存储系统
  - 选择易于水平扩展
  - 控制成本

## 数据处理的挑战

- 减小数据采集到处理完成的延迟
  - 数据在车上收集，最终存储在数据中心里



- 依据不同类型数据处理的任务选择合适的处理系统
  - CPU密集型
  - IO密集型
- 通用的任务定义以支持灵活的添加新任务

# 车队运营基础平台的挑战

- 快速变化的复杂需求
  - 需要合理的Web框架支持快速迭代
- 大量Web service的部署与管理
  - 需要完善Web服务通用组件
    - 部署工具
    - 日志记录
    - 监控平台

# 可视化平台

- 目标
  - 方便人类理解无人车系统看到的世界
- 挑战
  - 灵活性
    - 易于适配不同需求的工具
  - 性能
    - 3D实时渲染的高效实现
  - 跨平台
    - 支持桌面/移动/Web等多平台

