



# 基于.net core的农业无人 机调度指挥中心

主讲：陈志寅/ 极目机器人中国区软件负责人





# 陈志寅

极目机器人中国区（苏州）软件负责人&武汉极目CTO

Unicom云模式系统架构师

Unicom离线模式架构师

Unicom链路融合架构师

从0搭建极目软件研发团队

强大的自驱力，追求技术，追求用户体验





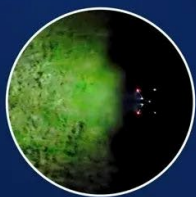


## 丘陵山地飞防专家

- 智能植保无人机



双目环境感知  
三重保障



夜航照明系统  
昼夜无差别作业



常温双峰弥雾喷头  
粒径可调



动力升级  
续航提升



精准播撒  
搅拌无阻，出料顺畅

### 极目智农PRO3.0 测绘 作业以及信息查询APP三合一 界面简单易学, 作业高效便捷



## 三链路合一

- 快速连接, 开机即飞。
- 数传链路结合全网通双卡链路, 无惧信号遮挡; 弱网环境, 链路稳定; 无网环境配合全新基站 (EAV-BAS3.0), 一站多机, 协同作业, 畅飞无阻。



▪ 测绘器



▪ 基站



▪ 遥控器



全新丘陵山地飞防专家 呵护作物全程生长  
EA-30XP再出发



# 目录

01 通信架构的设计

02 极目大脑能力

03 服务器端调度

04 QA



# PART 01

.NET Conf China  
2022  
开源 · 安全 · 赋能

## 通信架构

---



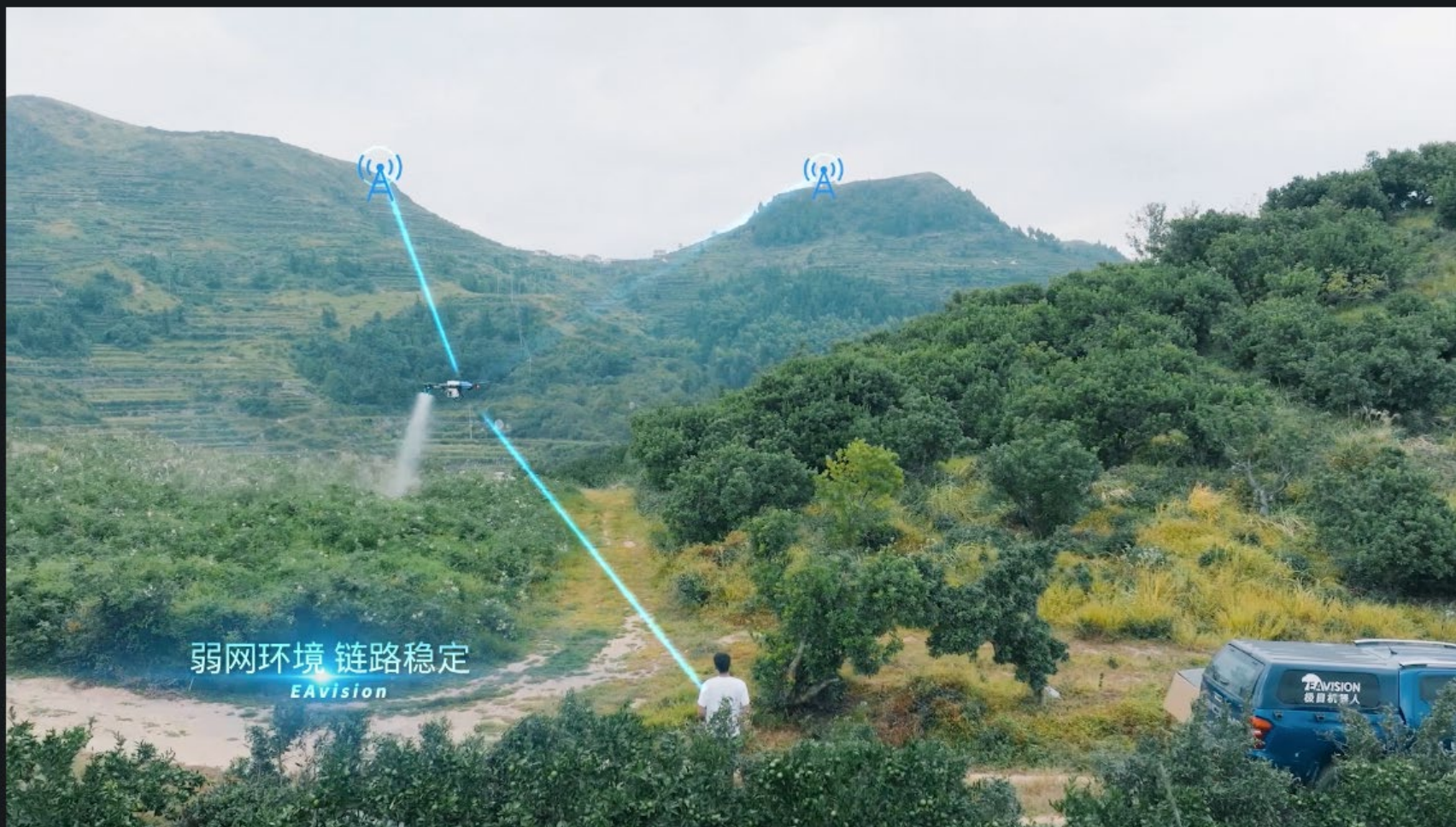




# 融合链路架构

## 三链路合一

快速连接，开机即飞；数传链路结合全网通双卡链路，无惧信号遮挡；弱网环境，链路稳定  
无网环境配合全新基站（EAV-BAS30），一站多机，协同作业，畅飞无阻



# » 为什么用.net core

Why?

背景：

之前的版本基于C/C++ 进行开发，主要受制于开发效率与调试效率。

决议：

有.net 的开发基础，.net core 开发效率更高，语法更易接受。

更好的基于http协议支持调试工具开发。

支持单文件打包成unix可执行文件，更易升级部署。

占用内存和cpu的开销更低。





# 技术栈

## 开发语言：

.Net Core -- 极目大脑

Dart、Kotlin -- 地面站

Java -- Broker, Consumer

C/C++ -- Sub App, 算法库

## 操作系统&架构：

ROS -- 导航、控制、避障、仿地

MQTT -- 云链路通信

UART -- 数传链路通信

ZeroMQ -- 大小脑通信

RocketMQ -- MQTT 消息处理

## 运行环境：

Ubuntu -- 极目大脑

Android -- 地面站

Ubuntu -- broker

ROS -- 惯导系统





# 极目调度指挥中心

.NET Conf China  
2022  
开源 · 安全 · 赋能

极目实时调度指挥中心

极目实时调度指挥中心

2022-11-28 星期一 13:56:30

2022-11-28 星期一 14:27:23

输入机架号

作业记录

EAVUAV352260580

持续作业: 79h3m 飞手: 黄崇波/18076783329

当前配对数量: 1 云基站: D0000044K8H

按架次查询 按时间段查询

输入架次查询 输入地块查询 查询

地块: 小牛眠 (2200361)  
taskCode: 123993t221128140729996

地块: 大六相 (2198711)  
taskCode: 123848t221124173055975

地块: 1 (2198610)  
taskCode: 123848t221124150915396

地块: 凤凰坪花厂5电线 (2195713)  
taskCode: 123671t221121181756336

地块: 凤凰坪花厂3 (2195624)  
taskCode: 123671t221121180646362

地块: 凤凰坪花厂2 (2195616)  
taskCode: 123671t221121163032853

切换为海拔

轨迹

高清地图

作业点

borderIndex

barrierIndex

legs+index

missionBorder

missionBarrier

mission+index

legs

测距工具 添加断点

作业回放

0 857

轨迹时间: 17:34:19 上个点 下个点

x1 播放 暂停 继续 结束

基本信息 作业信息 操作记录 设备信息

操作记录 错误记录

请选择消息类型

19:19:28 missionId:1669288335158  
BaseStation\_Response-BaseStationReport-基站汇报  
信息:基站id[0]-命令[连接到QX服务]-连接状态[连接良  
好]-消息号: 12443

19:19:27 地面站请求-TurnOnForceSpray-[85]开启强  
制喷洒-消息号: 140

19:19:26 System\_Response-TurnOnForceSpray-命令:  
[开启强制喷洒]返回:命令执行成功-消息号: 12438

19:19:26 missionId:1669288335158 无人机请求-  
Mission-准备执行任务-消息号: 12439

19:19:23 missionId:1669288335158  
BaseStation\_Response-BaseStationReport-基站汇报  
信息:基站id[0]-命令[连接到QX服务]-连接状态[连接良  
好]-消息号: 12432

19:19:18 missionId:1669288335158  
BaseStation\_Response-BaseStationReport-基站汇报  
信息:基站id[0]-命令[连接到QX服务]-连接状态[连接良  
好]-消息号: 12423

0:2

故障数量(架)

架次: 1020/78%

机架次: 1/0%

空闲

531

3186

面积:3212Mu

距离:555km

量:2212L

5

解决工单

林业防控大屏 实时调度大屏 作业统计大屏

TAIVISION

单品统计大屏 飞防统计大屏

林业防控大屏 实时调度大屏 作业统计大屏

TAIVISION

单品统计大屏 飞防统计大屏



# PART 02

.NET Conf China  
2022  
开源 · 安全 · 赋能

## 极目大脑能力

---



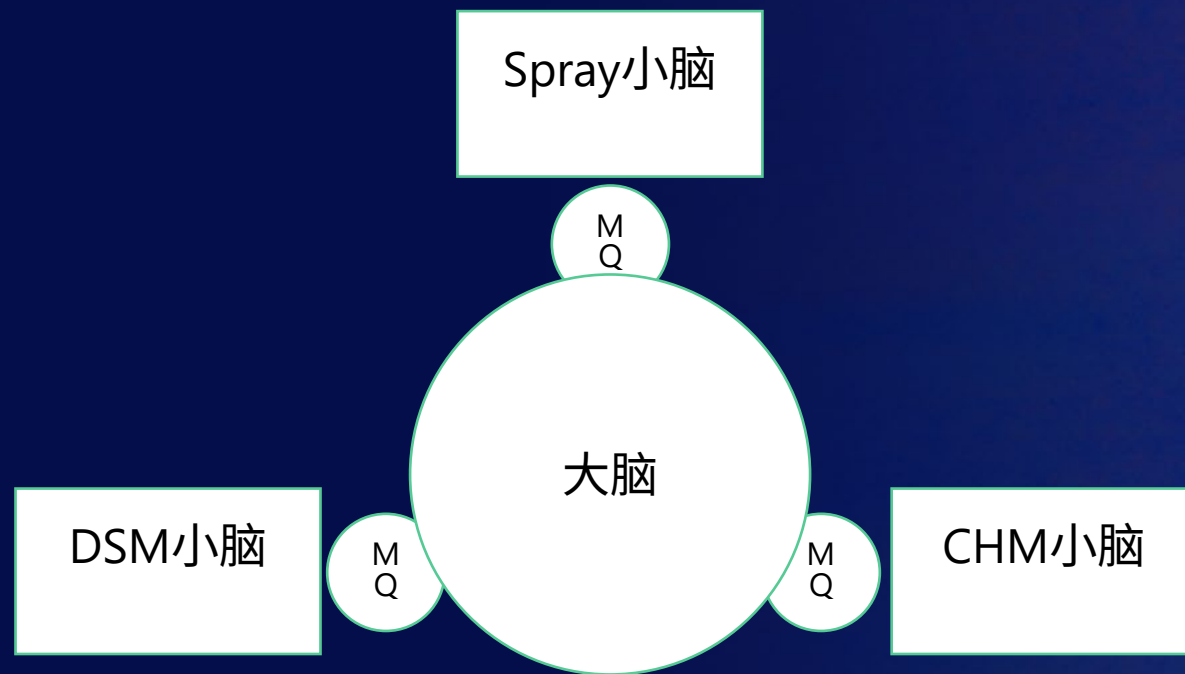


# » 大脑能力 -- 服务管理

服务注册：  
通过ZeroMQ的REQ/REP模式进行小脑注册

服务监管：服务注册后定时发送心跳，检测不到心跳后，进行服务重启

去中心化：  
每个大小脑，都是独立个体，继续zeroMQ进行通信串联



# » 大脑能力 -- 项目概述

README.md

## kongming [孔明] EAVison main brain project

### 极目大脑

#### Functions

- ☒ Communication between each application 与各个app通信
- ☒ Register center 注册中心
- ☐ Backend MQTT communication
- ☐ Flight command & system command transfer
- ☒ Log server
- ☒ Firmware update
- ☒ Processes monitor

#### Environment Build

- Install .net framework6 <https://dotnet.microsoft.com/zh-cn/download>
- **Windows** : Install visual studio & **Mac** Install visual studio for mac

#### Solution

- host kongming 主项目, 启动引导主程序
- core 核心库, 提供公共方法等
- mqtt
- sdk
- client
- test

#### How to build firmware

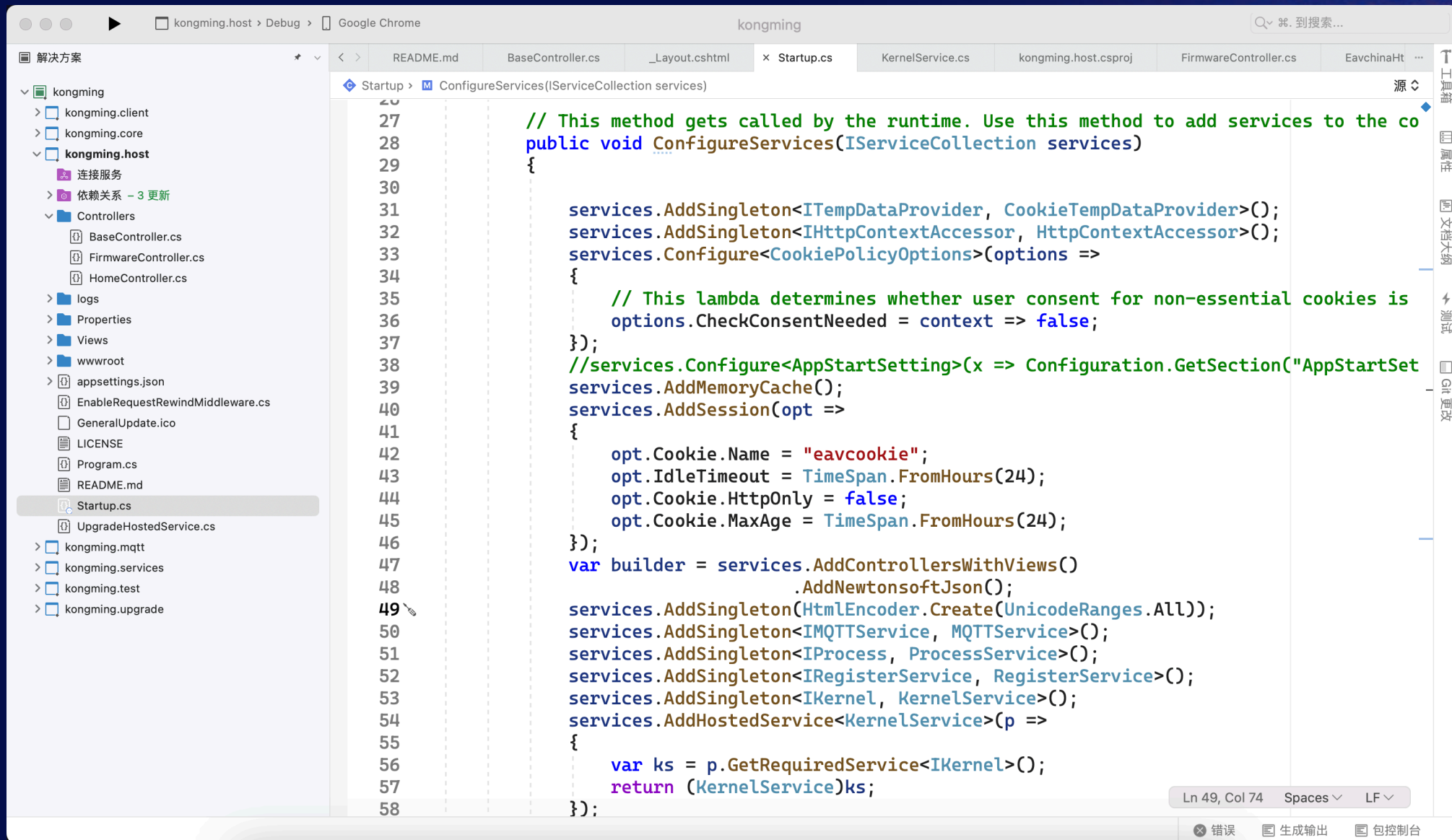
- multiple files : `dotnet publish -c release -r linux-arm64`
- single : `dotnet publish -c release -r linux-arm64 -p:PublishTrimmed=true`
- osx : `dotnet publish -c release -r osx-x64 -p:PublishTrimmed=true`





# » 大脑能力 -- 服务注入

.NET Conf China  
2022  
开源 · 安全 · 赋能



```
20
27 // This method gets called by the runtime. Use this method to add services to the co
28 public void ConfigureServices(IServiceCollection services)
29 {
30
31     services.AddSingleton<ITempDataProvider, CookieTempDataProvider>();
32     services.AddSingleton<IHttpContextAccessor, HttpContextAccessor>();
33     services.Configure<CookiePolicyOptions>(options =>
34     {
35         // This lambda determines whether user consent for non-essential cookies is
36         options.CheckConsentNeeded = context => false;
37     });
38     //services.Configure<AppStartSetting>(x => Configuration.GetSection("AppStartSet
39     services.AddMemoryCache();
40     services.AddSession(opt =>
41     {
42         opt.Cookie.Name = "eavcookie";
43         opt.IdleTimeout = TimeSpan.FromHours(24);
44         opt.Cookie.HttpOnly = false;
45         opt.Cookie.MaxAge = TimeSpan.FromHours(24);
46     });
47     var builder = services.AddControllersWithViews()
48         .AddNewtonsoftJson();
49     services.AddSingleton(HtmlEncoder.Create(UnicodeRanges.All));
50     services.AddSingleton<IMQTTService, MQTTService>();
51     services.AddSingleton<IProcess, ProcessService>();
52     services.AddSingleton<IRegisterService, RegisterService>();
53     services.AddSingleton<IKernel, KernelService>();
54     services.AddHostedService<KernelService>(p =>
55     {
56         var ks = p.GetRequiredService<IKernel>();
57         return (KernelService)ks;
58     });
59 }
```



# 大脑能力 -- 基于zeroMQ的服务注册

The screenshot shows the Visual Studio IDE with the 'RegisterService.cs' file open. The file is part of the 'kongming' project, specifically under the 'kongming.services' namespace. The code defines a 'RegisterService' class that inherits from 'BaseSrv' and implements 'IRegisterService'. It includes a 'StartServer()' method that initializes a 'ResponseSocket' and a 'NetMQPoller', and a 'RegSocket\_ReceiveReady()' method that processes incoming messages and registers new applications.

```
10 namespace kongming.services
11 {
12     public class RegisterService : BaseSrv, IRegisterService
13     {
14         ResponseSocket? regSocket;
15         ConcurrentDictionary<String, List<UavAppModel>> uavAppModels;
16         public RegisterService()
17         {
18             uavAppModels = new ConcurrentDictionary<String, List<UavAppModel>>();
19         }
20
21         public void StartServer()
22         {
23             regSocket = new ResponseSocket();
24             regSocket.ReceiveReady += RegSocket_ReceiveReady;
25             var poller = new NetMQPoller();
26             poller.Add(regSocket);
27             poller.RunAsync();
28             regSocket.Bind("tcp://127.0.0.1:5556");
29         }
30
31         private void RegSocket_ReceiveReady(Object? sender, NetMQSocketEventArgs e)
32         {
33             var msg = e.Socket.ReceiveMultipartMessage();
34
35             if (msg.FrameCount == 3)
36             {
37                 var appName = msg[0].ConvertToString();
38                 var appTopic = msg[1].ConvertToString();
39                 var appPort = msg[2].ConvertToInt32();
40
41                 Console.WriteLine("From App: {0}", appName);
42                 Console.WriteLine("Topic : {0} : {1}", appTopic, appPort);
43                 e.Socket.SendFrame("200:OK");
44                 //存储启动的应用
45                 AddApp(appName, new UavAppModel { AppName = appName, Topic = appTopic, Port = appPort });
46             }
47             else
48                 return;
49         }
50     }
51 }
```



# » 大脑能力 -- 看门狗

```
Welcome to Ubuntu 20.04.5 LTS (GNU/Linux 5.10.66 aarch64)
```

```
* Documentation: http://wiki.t-firefly.com
```

```
* Management:   http://www.t-firefly.com
```

```
System information as of Mon Nov 28 09:23:19 UTC 2022
```

```
System load:  0.07 0.37 0.59   Up time:       1:20 hour           Local users:   3
```

```
Memory usage: 26 % of 7664MB   IP:           192.168.8.8
```

```
Usage of /:    47% of 51G
```

```
Last login: Mon Nov 28 08:45:10 2022 from 192.168.5.18
```

```
root@firefly:~# supervisorctl status
```

```
uav_brain          RUNNING   pid 1377, uptime 1:20:13
```

```
root@firefly:~#
```

命令输入 (按ALT键提示历史,TAB键路径,ESC键返回,双击CTRL切换)

历史 选

文件

命令

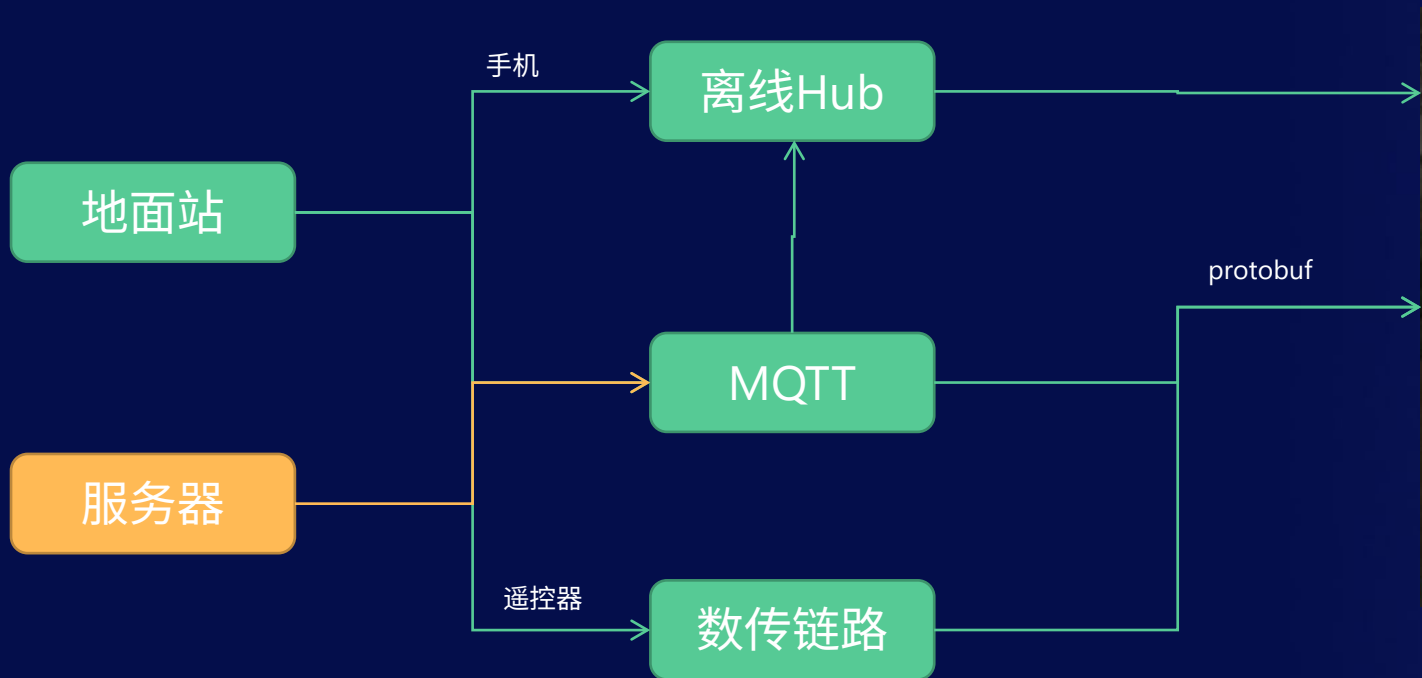
/root/brain

历史

历史 选

| 文件名              | 大小      | 类型  | 修改时间             | 权限         | 用户/用户组    |
|------------------|---------|-----|------------------|------------|-----------|
| logs             |         | 文件夹 | 2022/11/28 13:54 | drwxr-xr-x | root/root |
| wwwroot          |         | 文件夹 | 2022/11/17 13:48 | drwxr-xr-x | root/root |
| appsettings.json | 146 B   |     | 2022/10/31 10:44 | -rwxrwxrwx | root/root |
| kongming.client  | 24.9 MB |     | 2022/11/17 14:52 | -rwxrwxrwx | root/root |
| log.txt          | 28.6 MB |     | 2022/11/26 09:05 | -rw-r--r-- | root/root |
| uav_brain        | 45 MB   |     | 2022/11/18 17:50 | -rwxrwxrwx | root/root |

# » 大脑能力 -- 任务调度





# » 大脑能力 -- 调试中心

极目大脑控制台 ve

极目大脑控制台 version:1.0.1.0 设备状态 运行指令 模块管理 固件更新 日志下载

极目大脑状态

服务器版本: Lin  
使用 内存: 143 M  
使用 CPU: 0 %  
启动 时间: 2022  
总 存 储: 50 G

```
08:15:44@root: # ls
08:15:44@root: # 006.png
appsettings.json
kongming.client
logs
log.txt
uav_brain
wwwroot
```

@root: ls



# PART 03

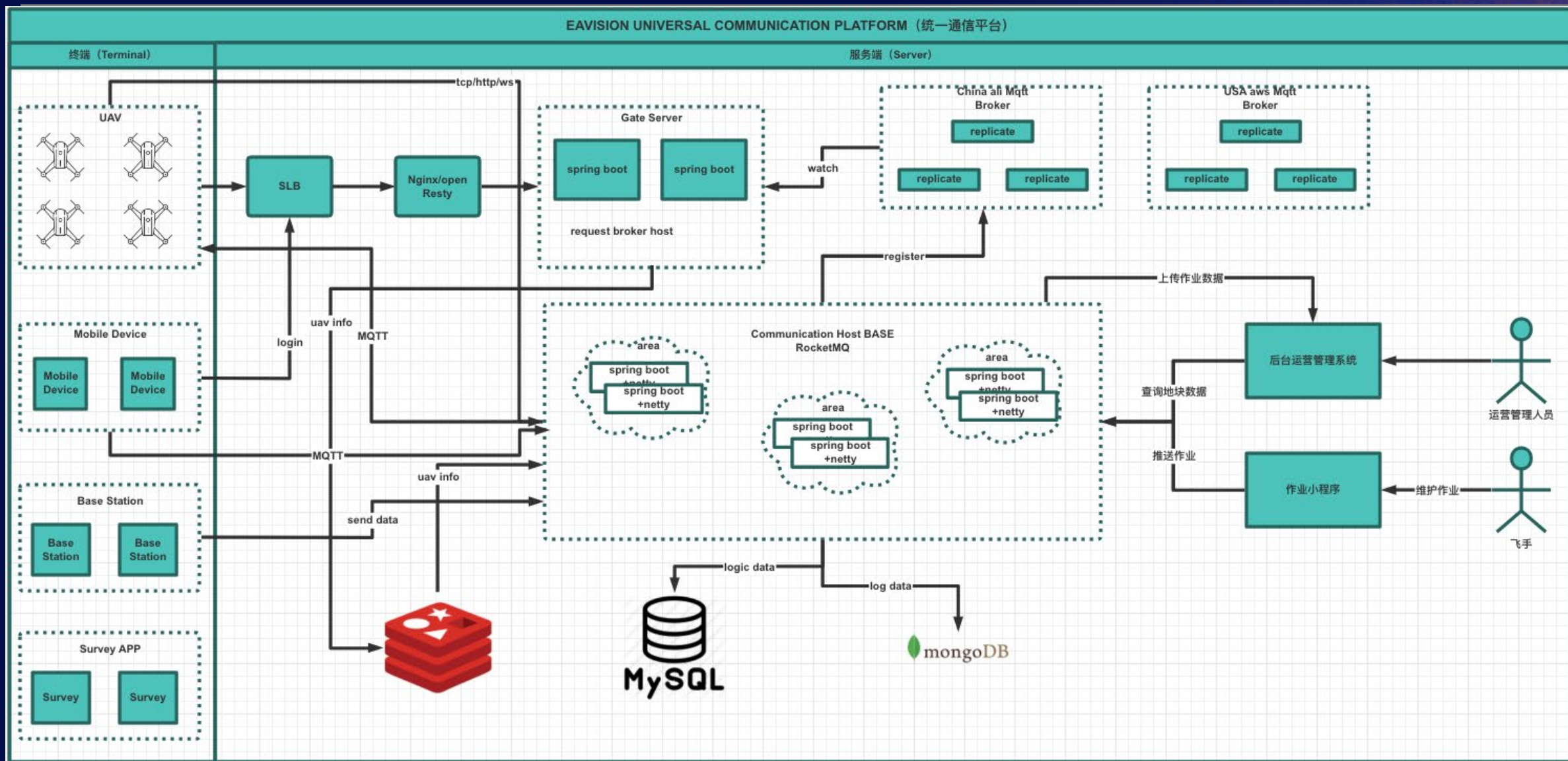
## 服务端调度

---





# 服务端调度-数据流向





# 服务端调度-能力

运行  
作业  
故障  
推送  
OTA

## 作业记录

架次: 1669621110138

架次时间: 2022-11-28 15:38:51—15:46:18

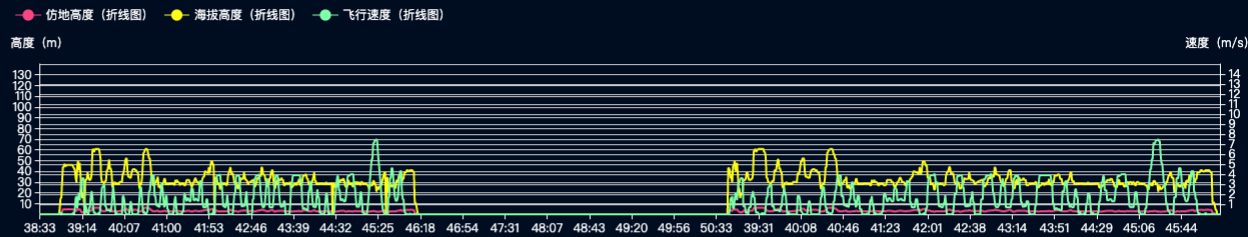
- ☒ 仿地高度 (折线图)
- ☒ 海拔高度 (折线图)
- ☒ 飞行速度 (折线图)
- ☒ 无人机状态 (区块图)
- ☒ 主电池电压 (折线图)
- ☒ 喷洒实时流速 (折线图)
- ☒ 卫星数量 (折线图)
- ☒ 蜂窝信号强度 (折线图)
- ☒ 桶内剩余量 (折线图)
- ☒ diffAge (折线图)
- ☒ 上下线 (折线图)
- ☒ 作业模式 (折线图)
- ☒ 作业-功能状态 (折线图)
- ☒ 作业-飞行准备 (折线图)
- ☒ 作业-飞行状态 (折线图)

取消选择

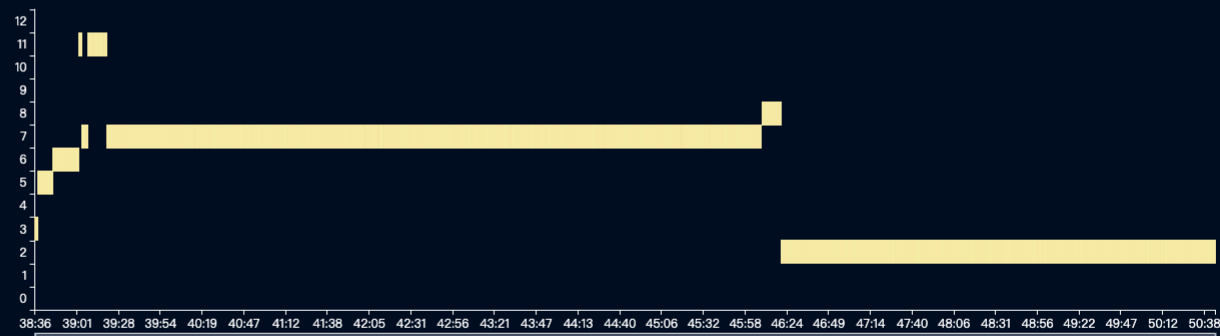
生成视图

## 生成视图

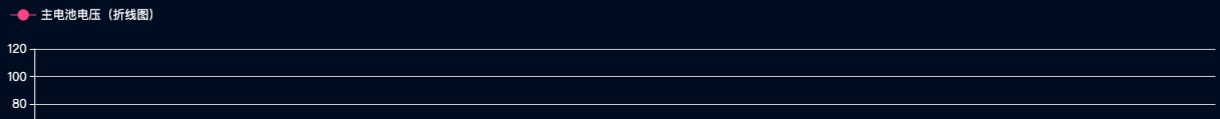
仿地高度 (折线图) - 海拔高度 (折线图) - 飞行速度 (折线图)



无人机状态 (区块图)



主电池电压 (折线图)



## 作业回放

1:43 上个点 下个点  
暂停 继续 结束

作业信息 操作记录 设备信息

## 错误记录

nId:1669621110138  
sponse-BaseStationReport-基站汇报  
命令[连接到QX服务]-连接状态[连接良  
925

nId:1669621110138  
sponse-BaseStationReport-基站汇报  
命令[连接到QX服务]-连接状态[连接良  
916

nId:1669621110138 无人机请求-  
任务-消息号: 11912

nId:1669621110138  
sponse-BaseStationReport-基站汇报  
命令[连接到QX服务]-连接状态[连接良  
906

nId:1669621110138  
sponse-BaseStationReport-基站汇报  
命令[连接到QX服务]-连接状态[连接良  
897



# PART 04

## QA

---

科技的进步不是为了人类的索取  
而是为了万物生长

