

# 基于物联网的电力综合管理装备及系统

## 1、系统架构图

见附件“日立信物联网系统架构图.pdf”

## 2、技术关键

### 2.1 docker技术

业务软件开发采用docker技术，Docker 是一个开源的应用容器引擎，让开发者可以打包他们的应用以及依赖包到一个可移植的容器中，然后发布到任何流行的 Linux 机器上，也可以实现虚拟化。容器是完全使用沙箱机制，相互之间不会有任何接口。

docker技术可以提高软件发布的效率，并保证开发环境、测试环境和生产环境的一致性，是平台化软件开发的一项关键技术。

### 2.2 系统架构

该平台的系统架构主要分为两大部分，技术平台和业务平台，基础平台主要包括设备接入、设备鉴权、数据预处理、数据库服务等基础业务；业务平台包括所有与应用功能相关的服务及软件，并通过技术平台的API服务、消息订阅发布服务进行联系；

该平台架构中涉及的关键技术包括redis发布订阅系统、API服务系统以及权限管理系统。

### 2.3 设备接入

该平台具备一个设备网关服务器（集群），用于处理设备的接入，对接入设备的合法性、连接可靠性、数据的合法性进行管理，通过该网关服务器，对不同厂家的设备的通信协议进行归一处理，对可扩展的通信协议进行兼容化处理；

设备接入技术包括了socket通信、HTTP通信、鉴权、协议解析、协议扩充等关键技术。

### 2.4 API技术

API技术是保障业务平台与基础平台、业务平台中各个服务之间的通信沟通的重要手段，也是系统软件解耦的关键技术之一。

### 2.5 数据库技术

该设计中包括了数据库存储技术、数据库的数据处理技术、时间线数据的压缩技术等。

数据存储技术主要用于对设备产生的各项数据的长期保存，并提供用户对数据的查看、检索等基本功能；数据库数据处理技术包括数据库触发器、数据的平均算法、特征值提取算法等数据处理技术，利用数据库强大数据管理能力与计算能力，在数据库端对数据进行基础的计算；时间线数据的压缩技术用于提高数据的检索效率，降低数据存储成本。

### 2.6 软件解耦

该设计中主要的软件解耦方法采用了消息订阅发布机制和API技术，针对平台中对实时性要求较高的数据及事物，采用redis消息发布订阅机制，不同的业务服务通过订阅技术平台中的数据通道，达到快速获取数据进行处理的效果，并对处理后的结果进行实时的发布，使其他订阅消息的服务实时获取计算结果；对于非实时性数据及事物，采用API接口技术进行交互；通过上述两种关键技术，达到软件解耦的目的。

## 3、技术路线（准备，启动，进行，运维，取得成果的过程）

### 3.1 准备阶段：

调研物联网平台软件的用户需求，形成软件需求书，根据软件需求书，制定软件功能框图，并根据软件功能框图，对软件中的各个功能进行细化，形成相应功能的软件开发任务书。

### 3.2 启动阶段：

制定平台架构设计方案，依照软件需求书，选择合适的平台架构方案（参见平台架构图），制定架构中各个部分之间的通信交互方案，制定通信交互协议。

根据各个功能模块的软件开发任务书，制定软件设计方案，并对软件设计方案进行审核；软件方案包括功能实现的技术方案，人员、时间安排，以及对所涉及的技术的成熟度进行评估。

### 3.3 开发阶段

对技术成熟度不高的关键技术进行前期预研，确保构建代码开始前，所有使用到的技术处于可控开发状态；

搭建开发、测试、生产环境，对各个环境中的基础软件、插件、数据库进行搭建部署；

根据软件架构及软件开发任务书，进行代码构建；

针对各个功能代码进行单元测试；

功能模块代码的开发，并根据平台架构的设计，对各个功能模块的软件进行联合调试；

对整体平台软件进行系统测试，并与设计开发任务书进行进一步的确认；

平台部署及上线，对经过系统测试的软件进行上线试运行，并进一步确认功能的合理性；

平台发布，交付用户使用；

### 3.4 发布后阶段

对软件平台进行发布后，建立软件运维团队，对软件运行过程中的状态进行监控，对软件运行过程进行维护和调整。

### 3.5 形成成果

对软件开发过程中运用的技术、知识产权进行保护；

对软件成果进行推广。

## 4、量化考核指标（软件方面）