面向工业应用的柔性可扩展的边缘计算软件系统

软件需求规约

版本 <1.0>

[注：用方括号括起来并以蓝色斜体（样式=InfoBlue）显示的文本，它们用于向作者提供指导，在发布此文档之前应该将其删除。按此样式输入的段落将被自动设置为普通样式（样式=Body Text）。]

[要定制 Microsoft Word 中的自动字段（选中时显示灰色背景），请选择 File>Properties，然后将 Title、Subject 和 Company 等字段替换为此文档的相应信息。关闭该对话框后，通过选择 Edit>Select All（或 Ctrl-A）并按 F9，或只是在字段上单击并按 F9，可以在整个文档中更新自动字段。对于页眉和页脚，这一操作必须单独进行。按 Alt-F9，将在显示字段名称和字段内容之间切换。有关字段处理的详细信息，请参见 Word 帮助。]

修订历史记录

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** |
| <13/10/2021> | 0.1 | 需求调研 | 张俸铭 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

目录

1. 简介 4

1.1 目的 4

1.2 定义、首字母缩写词和缩略语 4

1.3 参考资料 4

2. 整体说明 4

3. 功能需求 4

3.1 <Use case 图> 5

3.2 <Use case1 规约> 5

3.3 <Use case2 规约> 5

4. 非功能需求 5

4.1 易用性 5

4.2 可靠性 5

4.3 性能 5

4.4 可支持性 6

4.5 设计约束 6

5. 其它产品需求 6

5.1 联机用户文档和联机帮助的需求 6

5.2 接口需求 6

5.2.1 用户界面 6

5.2.2 硬件接口 6

5.2.3 软件接口 6

5.2.4 通信接口 7

5.3 适用的标准 7

软件需求规约 (简化版)

# 简介

## 目的

此软件需求规约文档用于描述面向工业应用的柔性可扩展的边缘计算软件系统的功能需求。本文档将详细地说明项目的整体效果、具体的功能和非功能需求以及设计约束，同时提供软件需求说明所需的其他相关要素。

## 定义、首字母缩写词和缩略语

|  |  |  |
| --- | --- | --- |
| **定义** | **缩略语** | **含义** |
| 边缘计算软件系统 | ECSS | 在靠近数据源（设备端）采用网络、计算、存储、应用核心能力为一体的开放平台，就近提供有限的低时延服务，并接入云平台进行管理和计算的软件系统。 |
| 业务流程管理标记 | BPMN | 一套图形化的流程建模标准，用于以业务流程模型详细说明各种业务流程。 |
| 协议集成系统 | PIS | 将各类协议或规则集成到相互关联的、统一和协调的系统之中，使资源达到充分共享，实现集中、高效、便利的管理。 |
| 工业异构设备 | IHD | 工业生产场景下应用不同通信协议、使用不同数据解析方法的高度异构化设备。 |

## 参考资料

[1] 《面向工业应用的柔性可扩展的边缘计算软件系统—立项建议书》，高级软件开发与管理课程第1小组。

[2] 《边缘计算参考架构3.0》，边缘计算产业联盟、工业互联网产业联盟。

[3] 《5G 与工业互联网融合应用发展白皮书》，工业互联网产业联盟、5G 应用产业方阵。

# 整体说明

## 产品总体效果

本项目基于工业生产场景中的实际需求，实现一个面向工业应用的边缘计算软件系统，在设备端进行数据采集，在边缘端实现对设备的管理和控制、对设备数据的筛选及处理，在云端进行数据的统一存储、边缘节点的统一管理、流程的调配调控、机器学习模型的训练和下发等。在项目中，通过实现灵活可配置的协议语义化管理和生产流程管理模块，我们的软件系统能够适配复杂多样的工业设备通信协议，适应多变的工业生产流程，从而实现高柔性、高可扩展性。

## 产品功能

本项目基于云—边—端架构，设计并实现面向工业应用的柔性可扩展的边缘计算软件系统。该系统通过云边协同，实现低时延的指令下发、数据处理和存储，并提供协议语义扩展和流程配置的服务接口。主要功能包括：

1. 数据采集和处理：对异构设备采集的数据，按照数据转换规则转换成统一的格式，以支持数据源直采。边缘端定期将数据汇总并上传到云端存储，同时对数据进行筛选，降低与云端的通信带宽。
2. 设备接入与管理：支持对多种工业通信协议的设备的接入，并能够在边缘端进行注册。边缘端能够通过相应的协议与设备交互。
3. 流程控制：云端支持用户的流程配置，并能够将流程下发到边缘端；边缘端依据所管理的设备进行相应的指令下发。
4. 机器学习推理：能够依据采集的设备数据对流程中的信息进行知识提取和知识发现，并应用于生产线上的质量检测等模块。

## 用户特征

边缘计算软件系统甲方的边缘端业务人员和云端业务人员。

## 约束

前端开发语言：HTML、JavaScript、CSS。

服务端编程语言：Java。

Web前端框架：Vue.js。

Web UI库：iView, ElementUI。

开发工具：Intellij IDEA、Visual Studio Code。

## 假设与依赖关系

1. 进度约束
2. 架构约束
3. 性能约束
4. 资源约束

项目要求自立项起13周内完成；

云—边—端架构的理念契合本系统的应用场景；

Java具有优秀的生态体系。

## 需求子集

系统安全稳定，操作简单方便，时延满足工业应用要求，界面用户友好。

[**SRS** 的这一节应说明影响产品及其需求的一般因素。本节并不列出具体的需求，而只是提供在第 3 节中详述的各种需求的背景，以使这些需求便于理解。所包括的内容有：

• 产品总体效果

• 产品功能

• 用户特征

• 约束

• 假设与依赖关系

• 需求子集]

# 功能需求

[此节为以UseCase模型和自然语言风格表达的需求说明为此设计的系统功能性需求。对于许多应用程序，此节会成为 **SRS** 包的主体部分，所以应仔细考虑此节的组织方式。此节通常按ＵseCase或特性来组织，但也可能会有其他适用的组织方式，例如按用户或子系统组织的方式。功能性需求可能包括特性集和安全性。]

## <Use case 图>

[画出Use case 图，并对每个actor或usecase有简要说明。如果系统比较大，则可按用户或子系统进行组织]

## <Use case1 规约>

[对每个usecase 要有详细规约，说明其事件流等信息。]

## <Use case2 规约>

# 非功能需求

系统的非功能性需求主要从纵切面入手考虑，可以分为可靠性、易用性、实时性、安全性等四个方面，需求描述如下：

|  |  |
| --- | --- |
| 纵切面需求 | 需求描述 |
| 可靠性 | 为了保证软件系统的无故障执行水平、可恢复性及准确性，本项目要求平均无故障运行时间达到95%。 |
| 易用性 | 提供美观、用户友好的前端界面，功能设计易于用户操作，同时为用户提供易于理解的操作文档。 |
| 实时性 | 为保证系统满足生产需要，边到端的指令下发延迟应小于500毫秒，图像采集、识别和反馈调度的延迟应小于800毫秒。 |
| 安全性 | 提供全生命周期的用户权限管理，保证数据、设备控制等生产信息安全。 |

## 易用性

[此节应包括所有影响易用性的需求。例如，

•指出普通用户和高级用户要高效地执行特定操作所需的培训时间

•指出典型任务的可评测任务次数或根据用户已知或喜欢的其他系统确定新系统的易用性需求

•指出在符合公认的易用性标准（如 IBM 的 CUA 标准和 Microsoft 的 GUI 标准）方面的需求]

## 可靠性

### 可用性

为了保证软件系统的无故障执行水平、可恢复性及准确性，本项目要求一年内的平均正常运行时间达到95%。

### 平均故障间隔时间（MTBF）

系统的平均故障间隔时间应大于30×24小时。

### 平均修复时间（MTTR）

数据在处理流程中的故障为严重故障，其修复时间不超过2小时；

由前端界面引发的用户交互故障为普通故障，其修复时间不超过1小时。

### 最高错误或缺陷率

根据能力成熟度模型集成（Capability Maturity Model Integration，CMMI），要求千行代码出错率不超过11.95‰（CMMI 1级）。

[对系统可靠性的需求应在此处说明。以下是一些建议：

• 可用性—指出可用时间百分比 ( xx.xx%)、使用小时数、维护访问权、降级模式操作等。

• 平均故障间隔时间 (MTBF) – 通常表示为小时数，但也可表示为天数、月数或年数。

• 平均修复时间 (MTTR) — 系统在发生故障后可以暂停运行的时间。

• 精确度 — 指出系统输出要求具备的精密度（分辨率）和精确度（按照某一已知的标准）。

• 最高错误或缺陷率—通常表示为每千行代码的错误数目 (bugs/KLOC) 或每个功能点的错误数目 (bugs/function-point)。

• 错误或缺陷率—按照小错误、大错误和严重错误来分类。需求中必须对“严重”错误进行界定，例如：数据完全丢失或完全不能使用系统的某部分功能。]

## 性能

[此节应概述系统的性能特征。其中需包括具体的响应时间。

• 对事务的响应时间（平均、最长）

• 吞吐量，例如每秒处理的事务数

• 容量，例如系统可以容纳的客户或事务数

• 降级模式（当系统以某种形式降级时可接受的运行模式）

• 资源利用情况，如内存、磁盘、通信等]

## 可支持性

[此节应列出将提高所构建系统的可支持性或可维护性的所有需求，其中包括编码标准、命名约定、类库、维护访问权和维护实用程序。]

## 设计约束

[此节应列出所构建系统的所有设计约束。设计约束代表已经批准并必须遵循的设计决定。其中包括软件语言、软件流程需求、开发工具的指定用途、构架及设计约束、购买的构件、类库等。]

# 其它产品需求

## 联机用户文档和联机帮助的需求

[如果存在对联机用户文档、帮助系统、关于声明的帮助等的需求，请在此说明。]

## 接口需求

[此节规定应用程序必须支持的接口/界面。它应非常具体，包含协议、端口和逻辑地址等，以便于按照接口/界面需求开发并检验软件。]

### 用户界面

[说明软件将实现的用户界面。]

### 硬件接口

[此节指出软件所支持的所有硬件接口，其中包括逻辑结构、物理地址、预期行为等。]

### 软件接口

[此节说明软件系统中与其他构件之间的软件接口。这些构件可以是购入的构件、取自其他应用程序重新利用的构件，也可以是为此 **SRS** 范围之外的子系统开发，但该软件应用程序必须与之交互的构件。]

### 通信接口

[说明与其他系统或设备（如局域网、远程串行设备等）的所有通信接口。]

## 适用的标准

[通过引用，此节说明了所有适用的标准以及适用于所述系统的相应标准的具体部分。例如，其中可以包括法律、质量及法规标准；业界在易用性、互操作性、国际化、操作系统相容性等方面的标准。]