分布式计算

软件构架文档

版本 <0.1>

修订历史记录

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** |
| 2017/08/30 | 0.1 | 草稿 | 王恒 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

目录

1. 简介 3

1.1 目的 3

1.2 范围 3

1.3 定义、首字母缩写词和缩略语 3

1.4 参考资料 3

1.5 概述 3

2. 构架表示方式 3

3. 构架目标和约束 3

4. 用例视图 3

4.1 用例实现 3

5. 逻辑视图 3

5.1 概述 3

5.2 在构架方面具有重要意义的设计包 3

6. 进程视图 3

7. 部署视图 3

8. 实施视图 3

8.1 概述 3

8.2 层 3

9. 数据视图（可选） 3

10. 大小和性能 3

11. 质量 3

软件构架文档

# 简介

[软件构架文档的简介应提供整个软件构架文档的概述。它应包括此软件构架文档的目的、范围、定义、首字母缩写词、缩略语、参考资料和概述。]

## 目的

本文档目的是实现一种基于自律分散技术的分布式计算系统设计。本文档将从构架方面对系统进行综合概述，其中会使用多种不同的构架视图来描述系统的各个方面。它用于记录并表述已对系统的构架方面作出的重要决策。

## 范围

本文档用于系统的设计者，开发者及使用人员了解系统的整体架构，功能模块划分，以及模块之间的接口约定。

## 定义、首字母缩写词和缩略语

自律分散系统（Autonomous Decentralized System，简称ADS）。在这一系统中所有的单元（子系统）都是独立平等的，它们之间不存在任何隶属关系。

自律分散管理系统(Autonomous control processor，简称ACP)

原子节点(Atom)

原子节点数据域(Atom Data Field) ADF

数据域(DF)。

DF使用内容代码（content code，CC）来判断一个数据是否必要的。DF管理负责与数据域的数据收发功能。

应用软件(APL)

原子节点数据域（ADF）

## 参考资料

自律分散系统入门 科学出版社

ZhongNDRedisDF-V2.0说明文档

ZhongNDRedisADF文档说明

## 概述

[本小节应说明此软件构架文档中其他部分所包含的内容，并解释此软件构架文档的组织方式。]

# 构架表示方式

[本节说明当前系统所使用的软件构架及其表示方式。还会从用例视图、逻辑视图、进程视图、部署视图和实施视图中列出必需的那些视图，并分别说明这些视图包含哪些类型的模型元素。]

# 构架目标和约束

描述架构设计最主要的目标就是满足关键系统功能需求和质量约束，这些功能需求和质量要求对软件架构有重大影响，并决定架构的设计。本节同时还列明影响架构设计的其他因素，如软件复用策略、使用商用构建、设计和实施的策略等。

## 目标

本设计的目标是设计分布式计算平台，即实现电网分析计算（如电网可靠性分析）的单节点运行到多节点并行运行的转变，提高电网分析计算的速度。另一个目标是基于自律分散的数据总线中间件，实现人机交互节点，以及各计算节点之间协作，同时又保持各节点相对独立。

## 关键需求说明

需求一：实现电网可靠性评估计算功能。用户可指定BPA模型文件，设置可靠性计算参数，调用可靠性计算算法，查询可靠性指标参数。此需求从单机版可靠性评估计算模块继承过来的功能性需求

需求二：实现电网可靠性评估计算速度优化。基于自律分散的数据总线中间件，实现可靠性评估计算的多节点并行计算，从而解决单机运行的计算能力瓶颈，提升大规模可靠性评估计算的性能优化。

需求三：实现功能模块之间的松耦合协同控制。基于自律分散的数据总线中间件，各计算节点相对独立，又能自律协调，可保证系统的在线扩展、在线维护及容错，此为系统的可维护性需求。

# 用例视图

本系统最终要实现的功能如下面的用例图所示。



## 用例实现

[本节通过几个精选的用例（场景）实现来阐述软件的实际工作方式，并解释不同的设计模型元素如何促成其功能的实现。]

内容控制码设计

|  |  |  |
| --- | --- | --- |
| 控制码 | 内容代码 | 数据根式 |
|  |  |  |

# 逻辑视图

[本节说明设计模型在构架方面具有重要意义的部分，例如设计模型被分解为多个子系统和包。而每个重要的包又被分解为多个类和类实用程序。您应该介绍那些在构架方面具有重要意义的类，并说明它们的职责，以及几项非常重要的关系、操作和属性。]

## 概述

[本节按照设计模型中包的层次结构来说明设计模型的整体分解情况。]

## 在构架方面具有重要意义的设计包

[对于每个重要的包，都用一个小节来加以说明，其中应包括该包的名称、简要说明以及显示该包中所有重要的类和包的图。

对于该包中的每个重要类，应包括其名称、简要说明，还可选择包括对其部分主要职责、操作和属性的说明。]

# 进程视图

[本节说明将系统分解为轻量级进程（单个控制线程）和重量级进程（成组的轻量级进程）的情况。本节的内容按照各个通信或交互的进程组来进行组织。说明进程之间的主要通信模式，例如消息传递、中断和会合。]

# 部署视图



# 接口设计

## 用户接口

用户接口主要再人机交互模块进行说明，但本设计文档不涉及人机交互模块的设计，只列出功能需求，并说明与数据总线以及分布式计算框架间见的消息约定。

## 内部接口

算法调度内部接口如下：

|  |  |
| --- | --- |
| **接口名** | **描述** |
| TaskSubscribe | 计算任务订阅接口 |
| TaskPublisher | 计算任务发布接口 |
| TaskExecuter | 计算任务执行接口 |

# 实施视图

[本节说明实施模型的整体结构、软件分解为实施模型中的层和子系统的情况，以及所有在构架方面具有重要意义的构件。]

## 概述

[本小节指定并定义各个层及其内容、添加到指定层时要遵循的规则以及各层之间的边界。还应包括一个显示层间关系的构件图。 ]

## 层

[对于每个层，都用一个小节来加以说明，其中包括该层的名称和一个构件图，并列举位于该层的子系统。]

# 数据视图（可选）

[从永久性数据存储方面来对系统进行说明。如果几乎或根本没有永久性数据，或者设计模型与数据模型之间的转换并不重要，那么本节就为可选。]