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。]

### BPA模型上传/更新



### 可靠性参数编辑



### 可靠性计算序列图



## 部署视图



人机交互节点部署采用单节点，分布式计算采用多节点部署。Redis总线节点集群实现DF的功能。遵循ADS技术的典型部署图

## 内容控制码设计

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **应用名称(APL)** | **功能描述** | **所属节点(ACP)** | **数据流向** | **内容代码(CC)** | **数据格式** |
| BPAModelUploader | BPA模型生成 | 人机交互 | 输出 | created\_BPAModel | RedisDB |
| BPA2PR | BPA网络模型转可靠性网络模型 | 人机交互 | 输入 | created\_BPAModel | RedisDB |
|  |  |  | 输出 | created\_PRModel | RedisDB |
| PRParamEditor | 可靠性参数编辑 | 人机交互 | 输入 | created\_PRModel | RedisDB |
|  |  |  | 输出 | created\_PRSettings | Xml File |
| PIBrowser | 可靠性指标浏览 | 人机交互 | 输入 | created\_RIResult | ControlMap |
| StateSample | 状态抽样 | 计算控制 | 输入 | created\_StateSampleSetting | ControlMap |
|  |  |  | 输出 | created\_StateSampleResult | ControlMap |
|  |  |  | 输出 | created\_StateEstimateTask | ControlList |
| StateEstimate | 状态后评估 | 计算控制 | 输入 | created\_StateEstimateTask | ControlList |
|  |  |  | 输入 | created\_StateSampleResult | ControlMap |
|  |  |  | 输入 | created\_StateEstimateSetting | ControlMap |
|  |  |  | 输出 | created\_StateEstimateResult | ControlMap |
|  |  |  | 输出 | created\_ ReliabilityIndexTask | ControlList |
| ReliabilityIndex | 可靠性指标计算 | 计算控制 | 输入 | created\_ ReliabilityIndexSetting | ControlMap |
|  |  |  | 输入 | created\_StateEstimateResult | ControlMap |
|  |  |  | 输入 | created\_ ReliabilityIndexTask | ControlList |
|  |  |  | 输出 | created\_ReliabilityIndexResult | ControlMap |

\*流向总线为输出，否则为输入

为了明确表达内容代码的含义,怎家内容代码的可读性，内容代码的定义参考IEC61968消息体的设计，由动词+名词表达。IEC61968动词列表如下

|  |  |  |
| --- | --- | --- |
| 请求 | 应答 | 事件 |
| get | reply | none |
| create | reply | created |
| update | reply | updated |
| cancel | reply | canceled |
| close | reply | closed |
| delete | reply | deleted |

# 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

分布式计算框架，底层通信和数据缓存由基于redis集群，redisson客户端，数据总线模块（ZhongNDRedisDF，ZhongNDRedisADF），内存数据库(ZhongNDMemDB)，对上为业务应用提供自律分算计算节点控制模块(ei-ads-core),其中包括三个包：自律控制包（acp），自律数据域包（adf），应用容器包(apl)。redis和redisson由第三方提供。数据总线和内存数据库的设计见自律分散体系架构设计文件等文档。本部分主要描述支撑业务应用的分布式计算相关模块的设计。

## 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。]

### 自律控制包（acp）

### 自律数据域包（adf）

### 应用容器包(apl)

# 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。]

# 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 |

## 与算法程序接口

分布式计算框架一个重要功能是为业务应用提供处理环境，包括消息接收，数据下载，数据上传，结算任务的分解，结算结果合并。本部分以实现可靠性计算为例，对业务应用需要提供的数据输入和输出进行约定。可靠性计算主要划分如下表所示。

|  |  |  |  |
| --- | --- | --- | --- |
| 模块名称 | 模型描述 | 模块功能 | 说明 |
| BPALoader | BPA模型加载 | 将BPA格式的文件转换为BPA内存模型。 | BPA模型文件RTS79.dat，RTS79.swi |
| BPA2PR | PR模型加载 | BPA内存模型转换为PR内存模型 | 模型转换依赖于可靠性计算参数文件，此文件为xml格式 |
| StateSample | 状态抽样软件 | 通过模特卡罗等方法计算出状态抽样集合 |  |
| StateEstimate | 状态后评估软件 |  |  |
| ReliabilityIndex | 可靠性指标计算软件 |  |  |

### BPA模型加载(GCBPALoader)

|  |  |  |  |
| --- | --- | --- | --- |
| 模块名称 | 输入参数 | 输出参数 | 调用方式 |
| BPALoader | swi和dat文件 | 内存数据库，数据库dbentry为MemDB.Bpa | 命令行方式，swi和data文件路径通过参数指定,文件路径为绝对路径  BPALoader RunPath BpaDatFile BpaSwiFile。  例：BPALoader E:\Bin E:/data/RTS79/RTS79.dat E:/data/RTS79/RTS79.swi |

### PR模型加载(GCBPA2PR)

|  |  |  |  |
| --- | --- | --- | --- |
| 模块名称 | 输入参数 | 输出参数 | 调用方式 |
| BPA2PR | MemDB.Bpa内存数据库，  可靠性参数文件， | 内存数据库，数据库dbentry为MemDB.PR | 命令行方式，可靠性参数文件由输如参数指定,文件路径为绝对路径  BPALoader RunPath BpaDatFile BpaSwiFile BpaRParamFile  例：BPA2PR E:\Bin E:/data/RTS79/RTS79.dat E:/data/RTS79/RTS79.swi E:\data\RTS79\RTS79.xml |

### 状态抽样软件(GCStateSample)

|  |  |  |  |
| --- | --- | --- | --- |
| 模块名称 | 输入参数 | 输出参数 | 调用方式 |
| StateSample | MemDB.Bpa内存数据库，MemDB.PR内存数据库，  抽样用户控制参数 | 更新内存数据库MemDB.Bpa中的FState（抽样状态），FStateFDev（抽样状态下故障设备） | 命令行方式，抽样控制参数通过命令行传入  StateSample RunPath param1 param2  参数按顺序如下。其中nPRSampleObject和nPRSampleMethod是必须的。其他根据方法填写，不用的给0。  nPRSampleObject;//抽样对象类型，全部；支路；发电机  nPRSampleMethod;//抽样类型  nMaxGenFault;// 抽样最大发电机故障重数  nMaxBranFault;//抽样最大支路故障重数  nMCSSimulateTime;//MCS最大抽样仿真时长  fMCSMinStateProb;//MCS[蒙特卡罗]设备故障概率门槛值  fFSTMaxCumuProb;//FST[快速排序]累积概率  fFSTMinStateProb;//FST[快速排序]设备故障概率门槛值  nFSTMaxStateNum;//FST[快速排序]最大状态数  nSTSMaxStateNum;//STS[状态抽样]最大状态数  fANAMinStateProb;//ANA[解析法]设备故障概率门槛值 |

抽样用户控制参数枚举如下：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 名称 | 类型 | 是否必填 | 描述 | 说明 |
| 1 | nPRSampleObject | int | 是 | 抽样对象类型，全部；支路；发电机 |  |
| 2 | nPRSampleMethod | int | 是 | 抽样类型 |  |
| 3 | nMaxGenFault | int |  | 抽样最大发电机故障重数 |  |
| 4 | nMaxBranFault | int |  | 抽样最大支路故障重数 |  |
| 5 | nMCSSimulateTime | double |  | MCS最大抽样仿真时长 |  |
| 6 | fMCSMinStateProb | double |  | MCS[蒙特卡罗]设备故障概率门槛值 |  |
| 7 | fFSTMaxCumuProb | double |  | FST[快速排序]累积概率 |  |
| 8 | fFSTMinStateProb | double |  | FST[快速排序]设备故障概率门槛值 |  |
| 9 | nFSTMaxStateNum | int |  | FST[快速排序]最大状态数 |  |
| 10 | nSTSMaxStateNum | int |  | STS[状态抽样]最大状态数 |  |
| 11 | fANAMinStateProb | double |  | ANA[解析法]设备故障概率门槛值 |  |

### 状态后评估软件(GCStateEstimate)

|  |  |  |  |
| --- | --- | --- | --- |
| 模块名称 | 输入参数 | 输出参数 | 调用方式 |
| StateEstimate | MemDB.Bpa内存数据库，MemDB.PR内存数据库，  状态评估用户控制参数 | 更新内存数据库MemDB.Bpa中的如下表：  FState/抽样状态  FStateFDev/抽样状态下故障设备  FStateMIsland/抽样状态下孤岛损失  FStateOvlDev/抽样状态下越限设备  FStateOvlAd/抽样状态下越限调整 | 命令行方式，状态评估用户控制参数通过命令行传入StateEstimate param1 param2 … |

抽样用户控制参数枚举如下：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 名称 | 类型 | 描述 | 说明 |
| 1 | fTinyGenThreshold | double | 环辐网分解发电机容量门槛值（容量低于该门槛值的发电机认为不是发电机） |  |
| 2 | fLowVoltThreshold | double | 环辐网分解发电机低电压门槛值（电压低于该门槛值的发电机认为是负荷而不是发电） |  |
| 3 | fZIL | double | 小阻抗清除小阻抗门槛值 |  |
| 4 | nPRSampleObject | int | 抽样算法类型 | (给出枚举类型) |
| 5 | nPRSampleMethod | int | 抽样类型 | (给出枚举类型) |
| 6 | nMaxGenFault | int | MCS, FST, STS, ANA抽样最大发电机故障重数 |  |
| 7 | nMaxBranFault | int | MCS, FST, STS, ANA 抽样最大支路故障重数 |  |
| 8 | bLineELimit | int | 线路消限开关 |  |
| 9 | bTranELimit | int | 主变消限开关 |  |
| 10 | bGenPELimit | int |  |  |
| 11 | bUPFCELimit | int |  |  |
| 12 | bAuxLoadAdjust | int | 厂用电负荷参与调整 |  |
| 13 | bEQGenAdjust | int |  |  |
| 14 | bEQLoadAdjust | int |  |  |
| 15 | nMCSSimulateTime | int | MCS最大抽样仿真时长 |  |
| 16 | fMCSMinStateProb | double | MCS[蒙特卡罗]设备故障概率门槛值 |  |
| 17 | fFSTMaxCumuProb | double | FST[快速排序]累积概率 |  |
| 18 | fFSTMinStateProb | double | FST[快速排序]设备故障概率门槛值 |  |
| 19 | nFSTMaxStateNum | int | FST[快速排序]最大状态数 |  |
| 20 | nSTSMaxStateNum | int | STS[状态抽样]最大状态数 |  |
| 21 | fANAMinStateProb | double | ANA[ 解析法 ]设备故障概率门槛值 |  |
| 22 | fDc2AcFactor | double | 直流潮流 2 交流潮流系数 |  |
| 23 | fMinIslandGLRatio | double | 孤岛的最小容载比 |  |
| 24 | bMultiThread | int |  |  |
| 25 | bUPFCAdjustRC | int |  |  |
| 26 | bGenBusLoadAsAux | int | 发电机母线负荷按厂用电处理 |  |

### 可靠性指标计算软件(GCReliabilityIndex)

|  |  |  |  |
| --- | --- | --- | --- |
| 模块名称 | 输入参数 | 输出参数 | 调用方式 |
| ReliabilityIndex | MemDB.Bpa内存数据库，MemDB.PR内存数据库，  可靠性指标用户控制参数 | 更新内存数据库MemDB.Bpa中的表：更新System | 命令行方式，可靠性指标用户控制参数通过命令行传入ReliabilityIndex param1 param2 … |

抽样用户控制参数枚举如下：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 名称 | 类型 | 描述 | 说明 |
| 1 | fTinyGenThreshold | double | 环辐网分解发电机容量门槛值（容量低于该门槛值的发电机认为不是发电机） |  |
| 2 | fLowVoltThreshold | double | 环辐网分解发电机低电压门槛值（电压低于该门槛值的发电机认为是负荷而不是发电） |  |
| 3 | fZIL | double | 小阻抗清除小阻抗门槛值 |  |
| 4 | nPRSampleObject | int | 抽样算法类型 | (给出枚举类型) |
| 5 | nPRSampleMethod | int | 抽样类型 | (给出枚举类型) |
| 6 | nMaxGenFault | int | MCS, FST, STS, ANA抽样最大发电机故障重数 |  |
| 7 | nMaxBranFault | int | MCS, FST, STS, ANA 抽样最大支路故障重数 |  |
| 8 | bLineELimit | int | 线路消限开关 |  |
| 9 | bTranELimit | int | 主变消限开关 |  |
| 10 | bGenPELimit | int |  |  |
| 11 | bUPFCELimit | int |  |  |
| 12 | bAuxLoadAdjust | int | 厂用电负荷参与调整 |  |
| 13 | bEQGenAdjust | int |  |  |
| 14 | bEQLoadAdjust | int |  |  |
| 15 | nMCSSimulateTime | int | MCS最大抽样仿真时长 |  |
| 16 | fMCSMinStateProb | double | MCS[蒙特卡罗]设备故障概率门槛值 |  |
| 17 | fFSTMaxCumuProb | double | FST[快速排序]累积概率 |  |
| 18 | fFSTMinStateProb | double | FST[快速排序]设备故障概率门槛值 |  |
| 19 | nFSTMaxStateNum | int | FST[快速排序]最大状态数 |  |
| 20 | nSTSMaxStateNum | int | STS[状态抽样]最大状态数 |  |
| 21 | fANAMinStateProb | double | ANA[ 解析法 ]设备故障概率门槛值 |  |
| 22 | fDc2AcFactor | double | 直流潮流 2 交流潮流系数 |  |
| 23 | fMinIslandGLRatio | double | 孤岛的最小容载比 |  |
| 24 | bMultiThread | int |  |  |
| 25 | bUPFCAdjustRC | int |  |  |
| 26 | bGenBusLoadAsAux | int | 发电机母线负荷按厂用电处理 |  |

GCStateEstimate程序包括启动和初始化，启动计算，计算结果返回，程序退出几个过程。程序启动过程如上所示，启动时传入对应的配置参数，程序启动之前MemDB.PR创建成功，FState和FstateDev表中已经准备好了抽样算例。GCStateEstimate端口号通过启动参数传入

a)启动计算过程：

1. ads向GCStateEstimate程序通过本地9001端口请求建立TCP连接
2. ads通过TCP连接发送FState的字符串，比如：

{

"memIndex": 15,

"sampleType": 1,

"fDevNum": 1,

"probability": 0.0319848,

"durition": 63969.7,

"stateNum": 2243,

"fLossGen": 400.0,

"fLossGenCap": 400.0,

"fLossLoad": 0.0,

"mIsland": 0,

"mIslandInsGen": 0.0,

"mIslandCutGen": 0.0,

"mIslandOutLoad": 0.0,

"balanceInsGen": 0.0,

"balanceCutGen": 0.0,

"balanceCutLoad": 0.0,

"overLimit": 1,

"eLimit": 0,

"eLimitCutGen": 2.18466,

"eLimitInsGen": 2.18466,

"eLimitCutLoad": 0.0,

"eLimitResult": 1,

"maxFaultZone": 0,

"maxFaultRatio": 0.0,

"faultGrade": 0,

"mSoutIndex": -1,

"estimated": 1

}

1. ads关闭连接。

b)计算结果返回过程：

1．GCStateEstimate与ads通过本地9002端口建立TCP连接

2，GCStateEstimate通过TCP连接发送评估结果，格式为json字符串

json串的格式如下：

public class StateEstimateResult {

private FState state;

private List<FStateFDev> devs = new ArrayList<FStateFDev>();

private List<FStateMIsland> mislands = new ArrayList<FStateMIsland>();

private List<FStateOvlDev> ovlDevs = new ArrayList<FStateOvlDev>();

private List<FStateOvlAd> ovlAds = new ArrayList<FStateOvlAd>();

}

json串样例如下。其中个的memIndex为内存记录索引，FStateFDev，FStateMIsland，FStateOvlAd，FstateOvlDev对象中的fState字段为FState中的memIndex。

{

"state": {

"memIndex": 15,

"sampleType": 1,

"fDevNum": 1,

"probability": 0.0319848,

"durition": 63969.7,

"stateNum": 2243,

"fLossGen": 400.0,

"fLossGenCap": 400.0,

"fLossLoad": 0.0,

"mIsland": 0,

"mIslandInsGen": 0.0,

"mIslandCutGen": 0.0,

"mIslandOutLoad": 0.0,

"balanceInsGen": 0.0,

"balanceCutGen": 0.0,

"balanceCutLoad": 0.0,

"overLimit": 1,

"eLimit": 0,

"eLimitCutGen": 2.18466,

"eLimitInsGen": 2.18466,

"eLimitCutLoad": 0.0,

"eLimitResult": 1,

"maxFaultZone": 0,

"maxFaultRatio": 0.0,

"faultGrade": 0,

"mSoutIndex": -1,

"estimated": 1

},

"devs": [{

"memIndex": 14,

"fState": 15,

"fDevTyp": 6,

"fDevIdx": 14,

"dFltPos": 0,

"dFltTyp": 1

}],

"mislands": [],

"ovlDevs": [{

"memIndex": 0,

"fState": 15,

"devTyp": 4,

"devIdx": 30,

"ovLmtP": 161.038,

"rated": 174.96,

"adLmtP": 158.854

}],

"ovlAds": [{

"memIndex": 0,

"fState": 15,

"ovlDevTyp": 4,

"ovlDevice": 30,

"adjDevTyp": 6,

"adjDevice": 10,

"adjValue": 2.18466

},

{

"memIndex": 1,

"fState": 15,

"ovlDevTyp": 4,

"ovlDevice": 30,

"adjDevTyp": 6,

"adjDevice": 22,

"adjValue": -2.18466

}]

}

## 与用户界面应用接口

用户界面应用负责MemDB.Bpa内存数据库，MemDB.PR内存数据库创建，可靠性计算参数编辑，算法控制参数设置，并把这些数据上传到数据总线的redis缓存中。用户界面应用与分布式计算节点之间不直接交互，而是通过数据总线进行消息和数据的交互。



# 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。]