本文件适用于上海欣方智能系统有限公司*在公司内部使用*，

上海欣方智能系统有限公司保留对此文件的解释权和修改权。

非经许可，不可转发及翻印

CIN-CDM概要设计

上海欣方智能系统有限公司

|  |  |
| --- | --- |
| 地址：上海市浦东新区碧波路690号张江微电子港8号4楼  邮政编码：201203 | 电话(Tel)：86-21-50495688  传真(Fax)：86-21-50495586 |
| 地址：北京市海淀区上地三街1号中黎科技园一号楼5层C段  邮政编码：100085 | 电话：010-62723600  传真：010-62723611 |

目 录

[第1章 概述 4](#_Toc427744446)

[1.1 目的 4](#_Toc427744447)

[1.2 文档历史 4](#_Toc427744448)

[1.3 术语 4](#_Toc427744449)

[第2章 系统简介 5](#_Toc427744450)

[2.1 系统功能 5](#_Toc427744451)

[2.1.1 同构/异构关系型数据库集群 5](#_Toc427744452)

[2.1.2 多种数据访问接口 5](#_Toc427744453)

[2.1.3 动态增删数据库节点 5](#_Toc427744454)

[2.1.4 事务二段提交 5](#_Toc427744455)

[2.1.5 数据库连接池 5](#_Toc427744456)

[2.1.6 连接数限制 5](#_Toc427744457)

[2.1.7 基于SQL的数据全量同步 5](#_Toc427744458)

[2.1.8 数据库集群节点的故障处理和恢复 6](#_Toc427744459)

[2.1.9 负载均衡 6](#_Toc427744460)

[2.1.10 安全 6](#_Toc427744461)

[2.1.11 维护命令 6](#_Toc427744462)

[2.1.12 管理接入 6](#_Toc427744463)

[2.2 系统环境 6](#_Toc427744464)

[第3章 系统用例 7](#_Toc427744465)

[3.1 用例参与者 7](#_Toc427744466)

[3.2 用例简介 7](#_Toc427744467)

[第4章 关键问题和解决方案 8](#_Toc427744468)

[4.1 多类型数据库集群 8](#_Toc427744469)

[4.2 数据同步方式 8](#_Toc427744470)

[4.3 数据库节点动态增删 9](#_Toc427744471)

[4.4 数据库故障处理 9](#_Toc427744472)

[4.5 数据恢复方式 9](#_Toc427744473)

[4.6 负载均衡方式 9](#_Toc427744474)

[4.7 数据缓存 9](#_Toc427744475)

[4.8 支持tuxedo 9](#_Toc427744476)

[4.8.1 XA接口 10](#_Toc427744477)

[4.8.2 非XA（esql）接口 11](#_Toc427744478)

[4.9 异构关系数据库集群 12](#_Toc427744479)

[4.9.1 问题描述 12](#_Toc427744480)

[4.9.2 解决方案 12](#_Toc427744481)

[4.10 异构数据集群 12](#_Toc427744482)

[4.10.1 问题描述 12](#_Toc427744483)

[4.10.2 解决方案 12](#_Toc427744484)

[4.11 管理系统接入 16](#_Toc427744485)

[4.11.1 问题描述 16](#_Toc427744486)

[4.11.2 解决方案 17](#_Toc427744487)

[第5章 系统逻辑结构 18](#_Toc427744488)

[5.1 静态结构 18](#_Toc427744489)

[5.1.1 概述 18](#_Toc427744490)

[5.2 客户端接口 19](#_Toc427744491)

[5.2.1 客户端连接管理 19](#_Toc427744492)

[5.2.2 PG编解码器 20](#_Toc427744493)

[5.2.3 数据加密模块 20](#_Toc427744494)

[5.2.4 鉴权模块 20](#_Toc427744495)

[5.3 开放接口 20](#_Toc427744496)

[5.4 任务调度模块 21](#_Toc427744497)

[5.4.1 调度逻辑 21](#_Toc427744498)

[5.4.2 负载均衡 21](#_Toc427744499)

[5.4.3 同步模块 21](#_Toc427744500)

[5.1 故障检测 21](#_Toc427744501)

[5.2 集群管理 21](#_Toc427744502)

[5.3 异构处理 21](#_Toc427744503)

[5.3.1 SQL转换模块 21](#_Toc427744504)

[5.3.2 错误码转换模块 21](#_Toc427744505)

[5.1 SQL队列 22](#_Toc427744506)

[5.2 DB线程池 22](#_Toc427744507)

[5.3 DB连接池 22](#_Toc427744508)

[5.4 动态结构 22](#_Toc427744509)

[5.4.1 数据访问 22](#_Toc427744510)

[5.4.2 动态增删节点 23](#_Toc427744511)

[5.4.3 事务二段提交 24](#_Toc427744512)

[第6章 系统物理结构 25](#_Toc427744513)

[6.1 静态结构 25](#_Toc427744514)

[6.2 动态结构 25](#_Toc427744515)

[第7章 数据结构 26](#_Toc427744516)

[7.1 数据模型 26](#_Toc427744517)

[7.2 数据库表 26](#_Toc427744518)

[7.3 配置文件 26](#_Toc427744519)

[第8章 系统性能 26](#_Toc427744520)

[8.1 性能需求 26](#_Toc427744521)

[8.2 性能设计 27](#_Toc427744522)

[第9章 参考资料 27](#_Toc427744523)

# 概述

## 目的

在去IOE环境下，为国产数据库提供全量异构数据库集群；对JAVA应用提供标准的JDBC接口，使数据库的替换对应用层的影响降到最低，同时提供高效的数据逻辑处理。本文档供产品经理，开发人员、测试人员参考。

## 文档历史

|  |  |  |  |
| --- | --- | --- | --- |
| 文档历史 | | | |
| 日期 | 参与者 | 版本 | 动作 |
| 2015-3-10 | 高志军 | PA1 | 初始版本 |
| 2015-5-15 | 高志军 | PA3 | 增加数据库集群节点故障检测处理和恢复 |
| 2015-6-16 | 程钢 | PA4 | 1. 格式修订，符合概要设计模版 2. 增加esql/XA接口和事务二段提交 3. 更新用例图和系统逻辑结构 4. 第5章模块细分和增加动态结构（序列图） 5. 增加关键问题4.8-4.10 |
| 2015-8-18 | 高志军 | PA5 | 1. 增加管理系统接入功能支持 |
| 2015-8-24 | 程钢 | PA6 | 1. 增加系统用例-安全和管理 2. 增加“接口定义”章节 3. 增加“数据库表” |
| 2015-9-9 | 程钢 | PA7 | 1. 接口定义章节增加集群管理和配置管理接口 2. 重新定义用户管理接口 3. 定义dbc\_user\_list、dbc\_table\_list、dbc\_view\_list、dbc\_table\_schema、dbc\_table\_index、dbc\_db\_list、dbc\_cdm\_list、dbc\_cdm\_cfg、dbc\_lic表 |

## 术语

| 术语 | 说明 |
| --- | --- |
| CDM | CINTel Data Midware，欣方数据中间件 |
| ODBC | Open Database Connectivity，开放数据库互连 |
| JDBC | Java Data Base Connectivity，java[数据库](http://baike.baidu.com/view/1088.htm)连接 |
| Postgres | 加州大学伯克利分校计算机系开发的 对象关系型数据库管理系统（ORDBMS） |
| Tuxedo | 交易中间件 |

# 系统简介

## 系统功能

### 同构关系型数据库集群

CDM采用标准ODBC数据库接口，可连接多种类型数据库，例如国产的达梦、金仓，国外的Oracle、Informix、MySQL等。

### 多种数据访问接口

CDM对外提供基于PostgreSQL的ODBC、JDBC、esql、XA接口，采用PGSQL协议，支持可灵活扩展的REST方式数据访问接口。

XA接口和esql接口用于实现事务二段提交（下文介绍）。

### 动态增删数据库节点

* + 增加数据库节点时，数据同步完成后加入集群
  + 删除数据库节点时，负荷自动转移到其它节点

### 事务二段提交

对于交易类应用，CDM需要支持事务的二段提交，支持对接tuxedo，事务二段提交是指事务处理分为预提交、提交、回滚三种操作，因此CDM与数据库之间的接口为esql。

### 数据库连接池

CDM与数据库之间创建可动态增长的连接池，复用相同属性连接以提高效率。

### 连接数限制

CDM可控制客户端连接数，达到上限时拒绝连接

### 基于SQL的数据全量同步

CDM基于SQL方式实现集群数据库的全量数据同步。每个数据库同步采用SQL队列+表级锁方式，确保数据更新正确。

### 数据库集群节点的故障处理和恢复

CDM执行SQL，发生严重错误（认为该数据库已经无法正常处理事务）时，将对应数据库节点进行逻辑删除，并产生告警。CDM同时保留后续对该节点的同步SQL。

针对发生故障的数据库节点，CDM能够定时检测数据库服务是否恢复，当数据库恢复时，将该节点未同步的SQL进行同步，重新加入到数据库集群中。

对于SQL执行失败的情况，CDM以SQL日志的方式记录，供后续人工恢复数据。另外，数据恢复也可以参考集群内其它正常节点的数据。

### 负载均衡

CDM提供可配置的负载均衡方式，包括权值轮询、最小时延、最短队列。

### 安全

#### 信任关系鉴权

CDM可提供基于IP网段、IP地址级、用户名、数据库名的信任关系鉴权，鉴权行为包括完全信任、密码鉴权和拒绝服务。

#### 密码验证

CDM可提供MD5、明文方式的密码验证，可以灵活配置客户端访问CDM的密码鉴权方式。

#### SSL加密访问

CDM支持与客户端的数据通信通过SSL加密，防止数据内容被窃取，保证数据的安全性

#### 故障检测和恢复

CDM提供对集群内数据节点的故障检测和恢复机制。当集群内某一数据节点发生故障时，CDM发出告警，更改节点状态为故障（等同于离线），此时允许在该节点上缓存数据的更新任务，待故障解除后自行同步，若故障不能自动恢复，需要维护人员通过web进行数据恢复和节点状态的更新。

### 管理

#### 用户管理

管理员可以对CDM的用户进行管理，普通用户仅具有修改本账户的部分信息和查询的权限，包括以下功能：

* 创建用户（管理员）
* 修改密码（管理员/普通用户）
* 删除用户（管理员）
* 查看用户列表（管理员/普通用户）

#### 数据管理

数据管理功能是对CDM的表、索引、视图和数据的管理，操作可概括为增删改查，包括以下功能：

* 获取table列表
* 获取指定表的索引
* 获取view列表
* 表的创建/修改/查询/删除
* 索引的创建/删除
* 数据的插入/更新/查询/删除

#### 集群管理

集群管理功能是对数据库集群节点的管理和CDM集群节点的管理，数据库集群管理包括以下功能：

* 新增集群节点

数据库名、数据库类型、数据库字符集、地址和端口信息、连接方式（ODBC、ESQL）

* 删除集群节点
* 获取集群节点列表
* 查询集群节点信息
* 更新集群节点状态

CDM集群节点管理包括以下功能：

* 查询CDM节点信息

主机名、IP、启动进程数、异常进程数

#### 配置管理

系统管理员可以通过页面对CDM集群内指定数据节点进行配置，包括以下配置信息：

* 连接配置参数

包括连接池大小、线程池大小

* 故障检测及恢复

包括可自动恢复类故障码和不可自动恢复类故障码配置

#### 自动化部署

系统管理员可以在本地配置CDM的逻辑环境，包括数据库集群节点信息，用户访问鉴权信息，对外提供服务信息。逻辑环境部署完成后，可导入到指定服务器的账号和路径下，通过自动化部署工具完成CDM部署。

此功能由安装脚本实现。

#### License管理

CDM允许系统管理员通过管理页面导入CDM运行所需要的license内容，包括以下功能：

* license查看
* license导入
* 制作license所需信息采集
* License告警

license包括以下内容：

* 有效期
* 最大连接数
* 连接数告警阀值

系统首次安装自带试用license。

### 维护命令

CDM通过命令行方式进行维护，包括以下命令：

* 查看数据库状态、队列长度、总任务数等信息
* 在线更新配置
* 增加数据库节点
* 删除数据库节点
* 激活数据库节点
* 去激活数据库节点
* 统计SQL任务执行时长

## 系统环境

CDM位于第三方应用与数据库之间，提供数据库集群、同步和数据访问功能，对应用而言CDM相当于数据库。



# 系统用例

## 用例参与者

* 第三方应用

通过ODBC/JDBC/esql/XA/REST接口方式，接入CDM的数据库应用

* 数据库

支持标准ODBC、esql接口的数据库，可接入CDM的数据库集群。

* 维护人员

CDM日常维护

## 用例简介



## 安全

### 信任关系鉴权

#### 完全信任

* 前置条件

CDM配置指定IP、数据库名、用户名的信任关系为完全信任

* 事件流

1. WEB在指定IP上以指定用户登录CDM，不指定密码
2. CDM检查信任关系通过
3. CDM向web返回登录成功的响应

* 后置条件

Web登录CDM成功

#### 完全信任-鉴权失败

* 前置条件

CDM配置指定IP、数据库名、用户名的信任关系为完全信任

* 事件流

1. WEB在非指定IP上以指定用户登录CDM，不指定密码

或

WEB在指定IP上以非指定用户登录CDM，不指定密码

1. CDM检查信任关系失败
2. CDM向web返回登录失败的响应，原因为不信任的主机/用户

* 后置条件

Web登录CDM成功

#### 密码验证

* 前置条件

CDM配置指定IP、数据库名、用户名的信任关系为密码验证

* 事件流

1. WEB在指定IP上以指定用户登录CDM，携带密码
2. CDM检查信任关系通过，密码验证成功
3. CDM向web返回登录成功的响应

* 后置条件

Web登录CDM成功

#### 密码验证-密码错误

* 前置条件

CDM配置指定IP、数据库名、用户名的信任关系为密码验证

* 事件流

1. WEB在指定IP上以指定用户登录CDM，携带错误密码
2. CDM检查信任关系通过，密码验证失败
3. CDM向web返回登录失败的响应，原因为密码错误

* 后置条件

Web登录CDM失败

#### 不信任

* 前置条件

CDM配置指定IP、数据库名、用户名的信任关系为拒绝

* 事件流

1. WEB在指定IP上以指定用户登录CDM
2. CDM检查信任关系不通过
3. CDM向web返回登录失败的响应，原因为不信任的主机或用户

* 后置条件

Web登录CDM失败

### 密码鉴权

#### Md5加密

* 前置条件

CDM配置指定IP、数据库名、用户名的信任关系为密码验证，加密方式为md5

* 事件流

1. WEB在指定IP上以指定用户登录CDM，携带密码的md5加密串
2. CDM检查信任关系通过，密码验证成功
3. CDM向web返回登录成功的响应

* 后置条件

Web登录CDM成功

#### 明文密码

* 前置条件

CDM配置指定IP、数据库名、用户名的信任关系为密码验证，加密方式为明文

* 事件流

1. WEB在指定IP上以指定用户登录CDM，携带明文密码
2. CDM检查信任关系通过，密码验证成功
3. CDM向web返回登录成功的响应

* 后置条件

Web登录CDM成功

### SSL数据加密

* 前置条件

Web指定已SSL方式访问数据库

* 事件流

1. WEB在登录前协商SSL加密方式及获取密钥
2. WEB登录CDM，携带用户名、密码
3. CDM检查信任关系通过，密码验证成功
4. CDM向web返回登录成功的响应
5. WEB进行SQL查询，并获取到数据。

* 后置条件

通过wireshark抓包无法解析到SQL和数据

## web管理

### 用户管理

#### 创建用户

* 前置条件

WEB以管理员角色成功登录CDM

* 事件流

1. WEB发起创建用户请求，携带用户名、密码、角色信息
2. CDM检查参数有效性通过，保存新用户信息
3. CDM向web返回创建用户成功的响应

* 后置条件

Web以新用户登录CDM成功

#### 创建用户-无权限

* 前置条件

WEB以普通用户角色成功登录CDM

* 事件流

1. WEB发起创建用户请求，携带用户名、密码、角色信息
2. CDM检查当前用户权限失败
3. CDM向web返回创建用户失败的响应，原因为无权限

* 后置条件

无

#### 创建用户-用户名已存在

* 前置条件

WEB以管理员角色成功登录CDM

* 事件流

1. WEB发起创建用户请求，携带用户名、密码、角色信息
2. CDM检查用户名已存在
3. CDM向web返回创建用户失败的响应，原因为用户名已存在

* 后置条件

无

#### 创建用户-用户名格式不正确

* 前置条件

WEB以管理员角色成功登录CDM

* 事件流

1. WEB发起创建用户请求，携带用户名、密码、角色信息
2. CDM检查用户名长度（2-16字节）以及是否具有无效字符（0-9A-Za-z\_之外的字符无效）
3. CDM向web返回创建用户失败的响应，原因为格式不正确

* 后置条件

无

#### 修改用户-改密码

* 前置条件

WEB以管理员角色成功登录CDM

* 事件流

1. WEB发起修改用户密码请求，携带用户名、新密码
2. CDM检查参数有效性和权限通过，更新用户密码
3. CDM向web返回修改密码成功的响应

* 后置条件

Web以此用户的新密码登录CDM成功，旧密码登录失败

#### 删除用户

* 前置条件

WEB以管理员角色成功登录CDM

* 事件流

1. WEB发起删除用户请求，携带用户名
2. CDM检查参数有效性和权限通过，删除用户信息
3. CDM向web返回删除用户成功的响应

* 后置条件

Web以被删除用户登录CDM失败

#### 删除用户-无权限

* 前置条件

WEB以普通用户角色成功登录CDM

* 事件流

1. WEB发起删除用户请求，携带用户名
2. CDM检查当前用户权限失败
3. CDM向web返回删除用户失败的响应，原因为无权限

* 后置条件

Web以待删除用户登录CDM成功

#### 删除用户-用户名不存在

* 前置条件

WEB以管理员角色成功登录CDM

* 事件流

1. WEB发起删除用户请求，携带用户名
2. CDM检查用户名不存在
3. CDM向web返回删除用户失败的响应，原因为用户名不存在

* 后置条件

无

#### 删除用户-资源未回收

* 前置条件

WEB以管理员角色成功登录CDM

* 事件流

1. WEB发起删除用户请求，携带用户名
2. CDM检查该用户存在未删除的数据库表或者具有未回收的权限
3. CDM向web返回删除用户失败的响应，原因为存在未删除的表或未回收的权限

* 后置条件

无

#### 查询用户列表

* 前置条件

WEB以管理员成功登录CDM

* 事件流

1. WEB发起查询用户列表请求
2. CDM查询用户列表
3. CDM向web返回用户列表

* 后置条件

Web在登录界面能够显示用户列表

### 数据管理

#### 获取table列表

* 前置条件

WEB以管理员/普通用户角色成功登录CDM

* 事件流

1. WEB发起查询table列表请求
2. CDM查询用户可见的table列表
3. CDM向web返回table列表

* 后置条件

Web在界面中能够显示当前用户的table列表

#### 获取view列表

* 前置条件

WEB以管理员/普通用户角色成功登录CDM

* 事件流

1. WEB发起查询view列表请求
2. CDM查询用户可见的view列表
3. CDM向web返回view列表

* 后置条件

Web在界面中能够显示当前用户的view列表

#### Table增删改查

* 前置条件

WEB以管理员/普通用户角色成功登录CDM

* 事件流

1. WEB通过SQL发起table的创建、删除、表结构修改、表结构查询
2. CDM执行SQL
3. CDM向web返回执行结果

* 后置条件

Web在界面中能够显示SQL执行结果

#### 查询表索引

* 前置条件

WEB以管理员/普通用户角色成功登录CDM

* 事件流

1. WEB通过SQL查询指定表的索引
2. CDM执行SQL
3. CDM向web返回执行结果

* 后置条件

Web在界面中能够显示表的索引

#### 创建/删除表索引

* 前置条件

WEB以管理员/普通用户角色成功登录CDM

* 事件流

1. WEB通过SQL创建/删除指定表的索引
2. CDM执行SQL
3. CDM向web返回执行结果

* 后置条件

Web在界面中能够显示执行结果

#### 数据增删改查

* 前置条件

WEB以管理员/普通用户角色成功登录CDM

* 事件流

1. WEB通过SQL发起数据的插入、删除、更新和查询操作
2. CDM执行SQL
3. CDM向web返回执行结果

* 后置条件

Web在界面中能够显示SQL执行结果

#### SQL执行-无权限

* 前置条件

WEB以普通用户角色成功登录CDM

* 事件流

1. WEB通过SQL对其他用户表进行数据的插入、删除、更新和查询操作
2. CDM执行SQL失败
3. CDM向web返回错误，原因为无权限

* 后置条件

Web在界面中能够显示错误提示

#### SQL执行-语法错误

* 前置条件

WEB以管理员/普通用户角色成功登录CDM

* 事件流

1. WEB通过SQL对本用户表进行数据的插入、删除、更新和查询操作，SQL具有语法错误
2. CDM执行SQL失败
3. CDM向web返回错误，原因为语法错误

* 后置条件

Web在界面中能够显示错误提示

#### SQL执行-无可用数据库节点

* 前置条件

1. WEB以管理员/普通用户角色成功登录CDM
2. 宕掉CDM数据库节点

* 事件流

1. WEB通过SQL对本用户表进行数据的插入、删除、更新和查询操作
2. CDM执行SQL失败
3. CDM向web返回错误，原因为无可用数据节点

* 后置条件

Web在界面中能够显示错误提示

#### SQL执行-其他错误

* 前置条件

WEB以管理员/普通用户角色成功登录CDM

* 事件流

1. WEB通过SQL对本用户表进行数据的插入、删除、更新和查询操作，SQL可导致报错，例如主键冲突、表不存在、连接已断开等
2. CDM执行SQL失败
3. CDM向web返回错误，原因为数据库返回的错误

* 后置条件

Web在界面中能够显示错误提示

### 集群管理

#### 获取集群节点

* 前置条件

WEB以管理员角色成功登录CDM

* 事件流

1. WEB发起获取集群节点请求
2. CDM查询集群节点列表
3. CDM向web返回集群节点列表

* 后置条件

Web在界面中能够显示所有集群节点

#### 增加集群节点

* 前置条件

WEB以管理员角色成功登录CDM

* 事件流

1. WEB发起增加集群节点请求
2. CDM在节点列表中增加新节点，开始数据同步，新增节点的状态为同步状态
3. CDM向web返回增加集群节点结果

* 后置条件

Web在界面中能够显示新增节点的状态并定时刷新

#### 删除集群节点

* 前置条件

WEB以管理员角色成功登录CDM

* 事件流

1. WEB发起集群节点离线请求
2. WEB发起删除集群节点请求
3. CDM在节点列表中删除指定节点
4. CDM向web返回删除集群节点结果

* 后置条件

Web在界面中能够显示集群节点状态

#### 查询节点状态

* 前置条件

WEB以管理员角色成功登录CDM

* 事件流

1. WEB发起查询集群节点状态请求
2. CDM在节点列表中查询指定节点的状态，包括在线（离线）状态、是否存在数据更新失败，另外同步状态包括同步百分比。
3. CDM向web返回集群节点状态

* 后置条件

Web在界面中能够显示集群节点状态，定时刷新

#### 更改节点状态-在线

* 前置条件

WEB以管理员角色成功登录CDM

* 事件流

1. WEB更改指定节点状态为在线
2. CDM判断指定节点的状态为离线，开始增量同步数据，状态为同步
3. CDM向web返回更改成功
4. WEB定时刷新查询指定节点的状态
5. CDM同步完成后自动更改此节点状态为在线

* 后置条件

Web在界面中能够显示集群节点状态，定时刷新

#### 更改节点状态-离线

* 前置条件

WEB以管理员角色成功登录CDM

* 事件流

1. WEB更改指定节点状态为离线
2. CDM判断指定节点的状态为在线，更改节点状态为离线，若处于同步状态则不允许更改
3. CDM向web返回更改成功

* 后置条件

Web在界面中能够显示指定集群节点的状态为离线

#### 更改节点状态-非法状态

* 前置条件

WEB以管理员角色成功登录CDM

* 事件流

1. WEB更改指定节点状态为非法状态
2. CDM判断目标状态非法
3. CDM向web返回更改失败，原因为非法状态

* 后置条件

Web在界面中能够显示集群节点的状态，状态未发生变化

#### 查看CDM节点信息

* 前置条件

WEB以管理员/普通用户角色成功登录CDM

* 事件流

1. WEB发起查看CDM节点信息请求
2. CDM向web返回CDM节点信息，包括IP、PORT、进程数和异常进程数

* 后置条件

无

### 配置管理

#### 查看配置

* 前置条件

WEB以管理员角色成功登录CDM

* 事件流

1. WEB发起查询配置请求
2. CDM向WEB返回配置信息，包括连接池大小、线程池大小、故障码配置

* 后置条件

无

#### 修改配置

* 前置条件

WEB以管理员角色成功登录CDM

* 事件流

1. WEB发起修改配置请求，包括连接池大小、线程池大小、故障码配置
2. CDM修改配置，向WEB返回结果，若需重启提示用户重启CDM。

* 后置条件

无

### 自动化部署

#### 正常流程

* 前置条件

以root权限运行自动化部署工具

* 事件流

1. 配置CDM主机、账户信息、启动进程数、服务端口
2. 配置各数据节点的主机地址、账户信息、数据库实例名、初始用户
3. 配置WEB的主机地址、账户信息、初始用户
4. 开始CDM自动化部署
5. 部署完成
6. 通过WEB能够访问CDM

* 后置条件

无

#### 网络断开

* 前置条件

以root权限运行自动化部署工具

* 事件流

1. 配置CDM主机、账户信息、启动进程数、服务端口
2. 配置各数据节点的主机地址、账户信息、数据库实例名、初始用户
3. 配置WEB的主机地址、账户信息、初始用户
4. 开始CDM自动化部署
5. 部署过程中网络断开
6. CDM自动化部署失败

* 后置条件

无

#### 配置冲突

* 前置条件

以root权限运行自动化部署工具

* 事件流

1. 配置CDM主机、账户信息、启动进程数、服务端口（端口已被占用）
2. 配置各数据节点的主机地址、账户信息（账户已存在）、数据库实例名、初始用户
3. 配置WEB的主机地址、账户信息（账户已存在）、初始用户
4. 开始CDM自动化部署
5. CDM自动化部署失败，提示端口占用和账户已存在，精确到主机。

* 后置条件

无

### License管理

#### License查看

* 前置条件

WEB以管理员/普通用户登录CDM

* 事件流

1. WEB发起查看license请求
2. CDM检查权限通过
3. CDM返回license信息、包含序列号、有效期、用户数、连接数
4. WEB展示license信息

* 后置条件

无

#### License导入

* 前置条件

WEB以管理员登录CDM

* 事件流

1. WEB发起导入license请求，携带序列号、用户名、公司名，文件上传至指定目录
2. CDM加载license成功
3. WEB显示license导入成功

* 后置条件

无

#### 制作License信息收集

#### License告警

# 关键问题和解决方案

列出关键问题及相应的解决方案。

## 多类型数据库集群

CDM可以集群各个厂家的数据库，介于各数据库在SQL语法和接入接口规范的不同，CDM需要做接入方式的统一（采用ODBC方式和esql方式）和必要的SQL、错误码转换。

* JDBC驱动

针对JAVA的外部应用，CDM提供标准的JDBC接口，使JAVA应用可以平滑接入CDM访问数据库。CDM采用开源Postgres（8.1版本以上）的PGSQL协议承载，JAVA应用采用其JDBC驱动即可访问CDM。

* 请求SQL语法转换

CDM根据分配的数据库类型做SQL请求的语法校验，对不符合目标数据库SQL语法的SQL做转换

* 请求和结果的字符集转换

CDM默认采用UTF-8编码，对不采用此字符集编码的数据库进行SQL请求和返回结果的编码转换

## 支持数据类型范围

附PostgreSQL同Oracle、达梦、金仓数据类型对照表



### 金仓

|  |  |
| --- | --- |
| 类型名 | 说明 |
| CHAR | 定长字符串 |
| VARCHAR | 变长字符串 |
| TEXT | 最大长度为64K的字符串 |
| NUMERIC | 可选精度的准确数字 |
| BIGINT | 8字节有符号整数 |
| INTEGER, INT | 4字节有符号整数 |
| SMALLINT, INT2 | 2字节位有符号整数 |
| TINYINT | 1字节有符号整数 |
| FLOAT | 可选精度的浮点数 |
| REAL | 单精度的浮点数 |
| DOUBLE | 双精度浮点数字 |
| DATE | 日历日期，包括年、月、日 |
| TIME | 一天里的时间，不包括时区 |
| TIMESTAMP | 包括日期和时间，不带时区的日期和时间 |
| BYTEA | 二进制串 |
| BOOLEA，BOOL | 逻辑布尔量 |
| BIT | 定长位串 |
| BIT VARYING | 变长位串 |

### Oracle

|  |  |
| --- | --- |
| 数据类型 | 说明 |
| CHAR(n) | 定长字符串 |
| VARCHAR2(n) | 变成字符串 |
| NUMBER(n,m) | Double类型 |
| NCHAR(n) | 具有数据库字符集的定长字符串 |
| NVARCHAR2(n) | 数据库字符集的变长字符串 |
| BINARY\_FLOAT(n) | Double类型 |
| BINARY\_DOUBLE(n) | Double类型 |
| FLOAT(n) | Double类型 |
| DATE | 字符串类型 |
| TIMESTAMP(n) | 字符串类型 |

### 达梦

|  |  |
| --- | --- |
| 类型名 | 说明 |
| CHAR(n) | 定长字符串 |
| VARCHAR2(n) | 变成字符串 |
| NUMERIC(m,n) | Double类型 |
| INTEGER | 整数类型 |
| FLOAT(n) | Double类型 |
| DOUBLE(n) | Double类型 |
| DOUBLE PRECISION(n) | Double类型 |
| DATE | 字符串类型 |
| TIMESTAMP(n) | 字符串类型 |
| LONGVARCHAR | 二进制数据类型 |

## 数据同步方式

CDM负责集群数据库的数据一致性。将外部应用的更新操作分发到集群内的每个数据库执行。

## 数据库节点动态增删

CDM可以动态的增加或删除集群中的数据库节点。在新增数据库节点时，尽量新增空数据库，避免数据库中遗留有冗余数据。新增数据库节点会有数据同步的耗时，在系统数据量比较大时，新增一个空数据库节点需要较长时间。

## 数据库故障处理

当集群内的数据库节点发生严重故障（宕机、网络断开等）时，需要将数据库节点从集群中剔除，待数据库故障解除后重新加入集群中。CDM提供数据库故障判断和检测机制，根据不同的故障级别执行不同的隔离操作。例如，将故障节点进行逻辑隔离、将故障节点进行物理隔离。

对从数据库集群中逻辑隔离处理的数据库节点，CDM能缓存后续的同步SQL，在数据库恢复正常后，将缓存的SQL依次执行，确保数据同步。

对物理隔离的数据库节点，CDM将后续同步SQL保存到文件中，由人工进行数据同步。

## 数据恢复方式

CDM在集群内数据库节点同步失败时，提供异常SQL记录供人为手动恢复数据。

## 负载均衡方式

CDM在处理前端大批量的访问请求时，可以按照一定规则将请求分配给各个数据库执行，规则可以是轮询、最闲、按比例等。

## 数据缓存

CDM是否需要缓存？经过调研，nosql的数据库和面向对象数据库（OQL语言）本身都可以独立运行且具有数据持久化的特性，对于关系型数据库而言通过配置缓存可以实现数据缓存功能，因此CDM无需单独实现缓存功能。

## 支持tuxedo

在交易系统中，一般是通过tuxedo服务操作数据库，tuxedo服务采用c编码，有XA接口和非XA（esql）两种方式，CDM作为中间件必须保证第三方应用以最小的代价集成CDM，因此需要前端支持tuxedo。

为了保证CDM数据访问接口的协议单一性，CDM继续使用PGSQL协议承载。

### XA接口

XA逻辑结构图如下：



* RM文件

增加Postgre\_XA:xapqls:-L/home/tuxedo/lixa/lib -L/usr/lib64 -llixapq -lecpg -L/home/vas/lib/lib -lglib-2.0 -lgmodule-2.0 -ldl -pthread -lgthread-2.0 -lrt -lxml2 -lz -lm

* Ubb配置文件

OPENINFO="PostgreSQL[LIXA]: hostaddr=192.168.2.119 port=5432 dbname=postgre user=postgre password=123456 connect\_timeout=5"

通过测试发现直接集成无法满足XA接口，原因是libxapq.so中定义的xa接口采用libpq.so的API实现，而嵌入式sql通过ecpg编译后会转换为libecpg.so中的接口，两种接口无法混用，libpq接口使用PGConn指针关联连接，ecpg接口使用连接名字符串关联连接，这样会导致sql执行失败（找不到连接）。

解决方案：

基于开源项目lixa重写liblixapq.so

1. libpq接口替换为ecpg接口，由于ecpg接口不能解析ubb配置的OPENINFO，还需要在xa\_open中解析OPENINFO。
2. ecpg接口是基于connection-name标识连接的，使用线程id作为connecttion-name。

### 非XA（esql）接口

非XA（esql）逻辑结构图如下：



* 第三方应用修改

tuxedo的服务需要做少量改动来适配CDM，修改包括以下方面：

1. Makefile中esql预编译器替换为ecpg，例如oracle的proc替换为ecpg。
2. Makefile中链接库替换为-lecpg，例如连oracle时这样链接——-L$ORACLE\_HOME/lib –lclntsh，而连接CDM时这样链接——-L$CDM\_HOME/lib –lecpg
3. 连接数据库代码修改为：EXEC SQL CONNECT TO <db> [AS 连接名] [USER <user>] [USING|IDENTIFIED BY <passwd>];

USER和USING分别指示用户名和密码，可选属性，视信任关系而定。

1. 事务提交和回滚不支持RELEASE关键字（oracle esql关键字，指示在事务提交或回滚后断开连接），支持COMMIT [WORK]和ROLLBACK [WORK]，其中WORK可选。

例如：oracle的esql程序可能会有“EXEC SQL COMMIT WORK RELEASE”

* SCF修改

SCF的esql接口需要支持二段提交，将原来的自动提交改为二段提交，将commit和rollback封装成SLPL供业务调用。

* CDM业务修改

通过参数区分二段提交，业务逻辑需要在执行SQL之后等待commit或者rollback。

## 异构关系数据库集群

### 问题描述

CDM作为关系数据库集群中间件能否将异构（指不同品牌）关系数据库作为一个集群提供数据服务？

### 解决方案

不同的数据库之间的SQL、数据类型、数据库函数和错误码也不尽相同，要做到异构关系数据库集群，CDM对客户端提供一套统一的SQL标准、数据类型、数据库函数以及错误码。

CDM对客户端提供PostgreSQL的接口，对数据库接口使用数据库本身的接口，对于每种数据库需要做SQL转换和错误码转换，这两种转换为了满足对客户端提供统一的PostgreSQL接口。

以数据库函数为例，目前调研了PostgreSQL、oracle和达梦三种数据库的函数，对照表如下：

|  |  |  |  |
| --- | --- | --- | --- |
| 统计项 | PostgreSQL | Oracle | 达梦 |
| 函数总数 | 266 | 102 | 139 |
| 不存在函数 | 141 | 305 | 268 |

由此可见，如果支持数据库函数的全集将不可能，有的函数只在一种数据库中存在。所以，CDM对客户端呈现的函数应该是多种数据库函数集合的交集，对每一种数据库而言，其开放的函数集都被削减了。

## 异构数据集群

### 问题描述

第三方应用会同时访问关系型数据和非结构化数据，对于非结构化数据包括redis、hbase和hdfs，cdm是否能够屏蔽掉数据访问的差异性，提供统一的数据访问接口？

### 解决方案

统一使用SQL，基于ODBC/JDBC，此方案需要将NoSQL DB的接口转换为SQL，NoSQL DB的一些特性无法使用SQL来表达，对整个集群而言，NoSQL DB的功能只能通过SQL部分开放。

**Redis和Hbase的对比**

|  |  |  |
| --- | --- | --- |
| 对比项 | Redis | Hbase |
| 数据结构 | strings/set/hash | 逻辑表【行、列族】 |
| 操作数 | 192（hash含14个） | 8 |
| 数据有效期范围 | 单条记录 | 单个表 |
| 主键 | Key | Rowkey |
| 二级索引 | 通过set实现 | 通过索引表实现 |
| 接口 | 命令 | 函数 |
| 新增记录 | 支持 | 支持 |
| 修改记录 | 支持 | 支持 |
| 删除记录 | 支持 | 支持 |
| 查询记录 | 支持 | 支持 |
| 唯一性约束 | 支持 | 支持 |
| 模糊匹配 | 支持 | 指定矩阵范围匹配 |
| 数据排序 | 支持 | 默认字典序 |

#### SQL请求

通过Redis和Hbase的对比，CDM对外提供的统一接口的操作数只能是两种NoSQL DB的交集。使用SQL接口，只能提供基本的增删改查功能，对于存储过程、函数、触发器都不能支持。鉴于NoSQL的数据存取方式和性能考量，CDM对应用层的SQL请求要有限制。

|  |  |
| --- | --- |
| 请求类型 | 限制因素 |
| Create Table | 需要指明表的主键（唯一性约束，存储的key），以对应NoSQL DB类型 |
| Drop Table | 无 |
| Alter Table | 不能修改主键字段 |
| Insert | 写入记录的主键必须有值，需要CDM处理唯一性约束 |
| Update | 需要主键作为更新条件，条件限“=,<=,>=,<,>”【性能考虑】或无修改条件 |
| Delete | 需要主键作为删除条件，条件限“=,<=,>=,<,>” 【性能考虑】或无删除条件 |
| Select | 需要主键作为查询条件，条件限“=,<=,>=,<,>” 【性能考虑】或无查询条件 |
| Join | 需要参与Join的表主键做查询条件，条件限“=,<=,>=,<,>” 【性能考虑】 |
| Call/Execute/Select function/procedure | 拒绝，不支持方法或存储过程 |
| SQL函数 | 仅限统计类函数，count，sum，avg |

#### 数据类型

应用在创建库表时可申明的字段类型仅限整型、浮点型和字符串。

#### SQL转义

|  |  |
| --- | --- |
| 请求类型 | 转义关键 |
| Create Table | 获取请求中的primary key并记录 |
| Drop Table | 无 |
| Alter Table | 校验请求中是否包含库表的primary key |
| Insert | 校验请求中是否包含库表的primary key |
| Update | 校验请求中是否包含库表的primary key |
| Delete | 校验请求中是否包含库表的primary key |
| Select | 校验请求中是否包含库表的primary key |
| Join | 校验请求中是否包含库表的primary key |
| Call/Execute/Select function/procedure | 拒绝 |
| SQL函数 | 校验请求中是否包含库表的primary key和函数名 |

#### 数据存储

库表在创建时，指定了NoSQL DB类型，CDM根据Redis和Hbase各自的特性进行数据存储。

|  |  |  |
| --- | --- | --- |
| 存储项 | Redis | Hbase |
| 库表结构信息 | CDM本地存储（缓存+文件） | |
| 库表 | 无 | 创建（行、列族） |
| 记录Key | 表名+主键值 | 主键值 |
| 记录Value | 字段列对应Hash中的field | 字段列对应到列族中的列 |

#### 结果返回

##### 创建、删除表

CDM本地记录、更新库表结构后，返回成功。对库表结构修改的（新增、删除字段、删除库表）情况，需要同步更新NoSQL DB中的存储数据

##### 新增记录

CDM在NoSQL DB写入成功后，返回成功，影响记录行数为1；失败返回0

##### 修改记录

CDM需要先按对应条件进行查询，获取记录总数，然后执行修改操作；删除成功后，返回影响记录行数

##### 删除记录

CDM需要先按对应条件进行查询，获取记录总数，然后执行修改操作；删除成功后，返回影响记录行数

##### 查询记录

CDM从NoSQL DB中获取数据后，根据本地记录的对应库表结构信息，按Postgre的通信协议，构造结果数据返回；若无数据，则返回空

#### 处理逻辑图



## 管理系统接入

### 问题描述

CDM需要提供一个统一接口，供管理系统查询集群节点数据源信息、管理用户和权限、更新和查询数据记录、管理集群节点。

### 解决方案

CDM新增管理进程，它与普通任务进程一样，只是作为管理角色，加载运行负责对接管理系统的业务逻辑。管理进程额外部署一个仅用作管理使用的Postgres数据库（版本在9.4以上），负责存储数据源信息。如下图所示：



CDM-M：CDM管理系统

Application：第三方应用

manager-process：CDM管理进程

task-process：CDM任务进程

DB-M：CDM数据信息存储数据库（Postgres）

DB-1，DB-2：CDM集群数据库节点（其它类型数据库）

CDM的管理系统直接与CDM的管理进程对接，协议遵循Postgre的PGSQL，管理系统对CDM的操作全部基于SQL语句，像数据库表结构的获取、更新，用户权限的管理，数据记录的更新、查询等。对于集群节点的管理，也可以基于自定义的库表操作，由管理进程识别并转化为相关操作，例如集群节点的查看、新增、删除等。

对于所有数据（包括用户、权限信息）的更新，除记录数据不需要再本地数据库执行外，其它数据都需要先在本地数据库执行，之后将更新请求分发到全部的数据库集群节点，对具体的数据库需要作SQL的适配性转换。包括

* SQL语法适配转换
* 数据类型转换
* 数据库表及字段适配转换（例如：用户权限相关的库表和字段）

对于记录数据的查询，CDM管理进程仅在集群中的某一节点获取即可，对于异常的返回，由CDM完成错误码的转换。

对于数据源非记录数据的查询，CDM管理进程仅在管理数据库中获取，例如数据库表结构、用户、用户权限等。CDM管理进程需要过滤这类操作，选择本地数据库执行并返回。

对于集群节点的更新，CDM管理进程自身逻辑更新完成后，还需要将信息同步到所有任务进程，最后管理进程修正自身节点信息。同步方式可以通过发送UDP命令。集群节点的查询在本地完成。

第三方应用发起的数据库表、表结构变更请求，任务进程需要同步到管理数据库中，保持一致性。

管理系统的接入鉴权由Postgres数据库负责完成。

此方案可支持Postgres数据库（9.4以上）的客户端pgadmin对接CDM。

## 支持多用户

### 问题描述

CDM在设计之初没有计划支持多用户，即一套CDM下只提供一套数据库服务。对于多个应用使用不同的数据库而不同的数据库又在同一台server上的场景不能很好的支持，因此CDM支持数据库多用户是不可或缺的。

### 解决方案

多用户逻辑结构图如下：



* 连接管理

负责接受连接、消息解析、调用安全模块和任务调度，是整个系统的入口。

* 安全模块

负责信任关系鉴权、密码验证和数据加密，多用户方案中需要对密码验证功能进行修改，CDM密码验证通过后，再用同样的用户名和密码登录数据库，连接是在登录过程中建立，不是通过配置建立。

* 任务调度

任务调度与单用户机制相比，需要增加多数据库的支持。多用户必然导致多数据库实例，任务调度模块首先要根据连接访问的数据库名进行一级分发，找到数据库名称所对应的SQL队列组，一组SQL队列中队列的数目与数据库节点数目一致，对于同一组SQL队列的处理与单用户机制相同。

# 系统逻辑结构

## 静态结构

### 概述

CDM基于SCF7.0版本进行开发，由独立的动态库（libdbcluster.so）和业务逻辑实现。系统结构图如下：



模块管理器和数据库线程池是SCF7.0新增的模块，能够对动态库进行统一管理和调用，数据库线程池能够并行执行sql，服务于CDM以外的应用。

CDM动态库libdbcluster在加载的时候会向模块管理器注册自身的通信回调函数、命令回调函数和SLPL回调函数，通信模块、命令接入模块和业务执行模块通过模块管理器调用CDM的API。

CDM业务负责sql的执行并将结果返回给客户端，通过业务调用可以在inmon中清楚的看到tps（原caps）和并发事务数（自动机数）。

Libdbcluster的逻辑结构图如下：



## 客户端接口

客户端接口负责对数据库客户端（第三方应用）提供通信和服务接口。

### 客户端连接管理

此模块用于实现与客户端通信和连接管理，逻辑功能有：

* + 消息收发接口

收消息接口为回调函数，在scf的通信模块初始化时通过模块管理器进行注册，config.server中需要配置协议类型为dbcluster（动态库名称）。

发消息接口通过逻辑名和linkID获取到对应连接并将消息转发出去。

* + 链路状态接口

此接口为回调函数，在scf的通信模块初始化时通过模块管理器进行注册，同上。当链路状态变化时scf自动调用此接口，用户实现初始化连接和客户端断链的处理。

* + 逻辑控制接口

此接口由收消息接口调用，负责调用PG编解码器和任务调度器，进而触发业务。

### PG编解码器

负责PGSQL的编解码，协议遵循PostgreSQL 8.1，参考<http://www.php100.com/manual/PostgreSQL8/>。

### 数据加密模块

数据加密模块支持SSL加密，在客户端发起startup时可以和CDM协商是否启用SSL加密。

若启用数据加密，后续的消息收发由PG编解码器调用此模块完成数据加解密。

### 鉴权模块

鉴权模块处理用户登录请求，包括用户名密码校验、信任关系鉴权两个方面。

## 开放接口

Libdbcluster开放的接口，基于SCF7.0框架定义三类函数列表：SLPL列表、命令列表和RPC列表。

* + SLPL列表
* 同步查询
* 同步更新
* 发送结果
  + 命令列表
* 更新配置
* 查看DB状态
* 增加/删除DB节点
* 同步数据
* 激活/去激活DB节点
  + RPC列表
* 任务队列初始化
* 获取任务
* 收消息
* 链路状态通知

## 任务调度模块

任务调度模块对SQL任务进行调度和同步。

### 调度逻辑

### 负载均衡

### 同步模块

## 故障检测

## 集群管理

## 异构处理

### SQL转换模块

### 错误码转换模块

## SQL队列

## DB线程池

## DB连接池

## 动态结构

对第3章列出的典型用例，把每个用例和每一个路径细化到模块一级。建议用序列图（UML Sequence Diagram）描述，该图的对象（Object）为各个模块。

### 数据访问

1. 第三方应用作为客户端发起SQL查询。
2. 客户端接口解析SQL，通过invokeservice触发CDM业务，将SQL传递给业务。
3. CDM业务解析SQL操作类型，针对select和写操作分别调用不同的接口。对于读操作，调用负载均衡模块选择目标数据库并将SQL发送到目标数据库的SQL队列；对于写操作，调用同步模块将SQL分发到各个数据库的SQL队列。
4. Db线程池从各自的SQL队列中获取任务，根据任务中指定的数据库名称从连接池中选择空闲连接。
5. 线程执行SQL，等待数据库返回结果。
6. 数据库返回结果到线程，线程将结果返回给自动机。
7. CDM业务将结果返回给客户端接口。
8. 客户端接口将结果进行编码并返回给第三方应用。



### 动态增删节点



### 事务二段提交

#### XA流程

暂不支持

#### 非XA（esql）流程



## 接口定义

### 管理接口

管理接口与第三方应用的接口相同，都采用SQL， 可使用odbc/jdbc连接方式，此接口既支持CDM WEB，也支持开发者使用的工具——psql和pgAdmin。

本节只列出与系统表相关的接口，与系统表无关的接口（例如创建表）遵循标准SQL。

#### 用户管理

##### 查询用户列表

GET dbc\_user\_list;

返回结果相当于select结果，表结构参见[dbc\_user\_list](#_dbc_list)。

##### 创建用户

SET op=newuser, name=[usr], pwd=[passwd],db=[dbname],role=[role]

|  |  |
| --- | --- |
| 参数 | 含义 |
| op | 操作类型，newusr表示新增用户 |
| name | 新增用户名 |
| pwd | 密码，格式为”md5”+md5(明文密码+用户名)，例如：  md57b40bcce1b88c4020fb0915e53771d84 |
| db | 数据库名 |
| role | 角色  super：超级用户  common： |

##### 修改用户密码

SET op=chpwd, name=[usr], pwd=[passwd],db=[dbname]

|  |  |
| --- | --- |
| 参数 | 含义 |
| op | 操作类型，chpwd表示修改密码 |
| name | 用户名 |
| pwd | 新密码，格式为”md5”+md5(明文密码+用户名)，例如：  md57b40bcce1b88c4020fb0915e53771d84 |
| db | 数据库名 |

##### 删除用户

SET op=deluser, name=[usr], db=[dbname]

|  |  |
| --- | --- |
| 参数 | 含义 |
| op | 操作类型，deluser表示删除用户 |
| name | 用户名 |
| db | 数据库名 |

#### 数据管理

##### 获取table列表

GET dbc\_table\_list [username];

Username填写登录用户名。

返回结果相当于select结果，表结构参见[dbc\_table\_list](#_dbc_table_list_1)。

##### 获取view列表

GET dbc\_view\_list [username];

Username填写登录用户名。

返回结果相当于select结果，表结构参见[dbc\_view\_list](#_dbc_view_list)。

##### 获取表结构

GET dbc\_table\_schema [tablename];

tablename填写表名。

返回结果相当于select结果，表结构参见[dbc\_table\_schema](#_dbc_table_schema)。

##### 获取表索引

GET dbc\_table\_index [tablename];

tablename填写表名。

返回结果相当于select结果，表结构参见[dbc\_table\_index](#_dbc_table_index)。

#### 集群管理

##### 新增集群节点

SET op=newdb, name=[db1], connection=[user]/[passwd]@[dbname], host=[ip], port=[port], ctype=[odbc/esql], charset=[utf8/gbk]

SET op=newdb, name=[db1], user=[user], pwd=[passwd], db=[dbtest], host=[ip], port=[port], ctype=[odbc/esql], charset=[utf8/gbk]

|  |  |
| --- | --- |
| 参数 | 含义 |
| op | 操作类型，newdb表示新增集群节点 |
| name | DB节点逻辑名称 |
| user | user表示数据库用户名 |
| pwd | pwd表示密码 |
| db | db表示对应数据库名 |
| connection | 连接标识，user表示数据库用户名，passwd表示密码，dbname表示数据库名 |
| host | 数据库服务器地址 |
| port | 数据库服务端口 |
| ctype | cdm与数据库的接口方式，支持odbc和esql两种 |
| charset | 数据库字符集，涉及中文字符时配置，取值utf8或gbk |

##### 删除集群节点

SET op=deldb, name=[db1]

|  |  |
| --- | --- |
| 参数 | 含义 |
| op | 操作类型，deldb表示删除集群节点 |
| name | DB节点逻辑名称 |

##### 查询集群节点信息

GET dbc\_db\_list;

返回结果相当于select结果，表结构参见[dbc\_db\_list](#_dbc_db_list)。

##### 更新集群节点状态

SET op=setstate, name=[db1], value=[online/offline];

|  |  |
| --- | --- |
| 参数 | 含义 |
| op | 操作类型，setstate表示更新集群节点状态 |
| name | 节点标识 |
| value | 节点状态，取值：online、offline |

##### 查询CDM节点信息

GET dbc\_cdm\_list;

返回CDM节点信息，相当于select结果，参见章节“[dbc\_cdm\_list](#_dbc_cdm_list)”。

#### 配置管理

##### 查看配置信息

GET dbc\_cdm\_cfg;

返回CDM节点配置信息，相当于select结果，参见章节“ [dbc\_cdm\_cfg](#_dbc_cdm_cfg) ”。

##### 修改配置信息

SET op=config, [name]=[value];

|  |  |
| --- | --- |
| 参数 | 含义 |
| op | 操作类型，config表示修改配置信息 |
| [name] | 参数名称，参见章节“ [dbc\_cdm\_cfg](#_dbc_cdm_cfg) ” |
| [value] | 参数值，参见章节“ [dbc\_cdm\_cfg](#_dbc_cdm_cfg) ” |

#### license管理

##### license查看

GET [dbc\_lic](#_dbc_lic);

|  |  |
| --- | --- |
| 参数 | 含义 |
| User | 用户名，参见章节“[dbc\_lic](#_dbc_lic)” |
| Company | 公司， 参见章节“[dbc\_lic](#_dbc_lic)” |
| Licnumber | 序列号，参见章节“[dbc\_lic](#_dbc_lic)” |
| Expire | 有效期，参见章节“[dbc\_lic](#_dbc_lic)” |
| Session\_max | 连接数，参见章节“[dbc\_lic](#_dbc_lic)” |
| User\_max | 用户数 |

##### license导入

SET op=newlicense, name=[name], company=[company], licenseNum=[licenseNum],content=[content];

|  |  |
| --- | --- |
| 参数 | 含义 |
| op | 操作类型，newlicense表示导入新的license |
| name | 参数名称，用户名 |
| company | 参数名称，公司名 |
| licenseNum | 参数名称，序列号 |
| content | 参数名称，license文件内容 |
| [name] | 用户名 |
| [company] | 公司名 |
| [licenseNum] | 序列号 |
| [content] | license文件内容（ASCii编码） |

# 系统物理结构

## 静态结构



## 动态结构

详细描述系统所有进程、业务FSM的生命周期以及创建关系。



# 数据结构

描述系统的数据模型、数据库表定义、配置文件定义。

## 数据模型

## 数据库表

### dbc\_user\_list

此表结构是虚拟的，不可直接操作，参见章节“[查询用户列表](#_查询用户列表)”。

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 类型 | 描述 | 备注 |
| name | varchar(20) | 用户名 |  |
| super | char(1) | 是否为DBA，Y：是，N：否 |  |

### dbc\_table\_list

此表结构是虚拟的，不可直接操作，参见章节“[获取table列表](#_获取table列表)”。

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 类型 | 描述 | 备注 |
| name | varchar(20) | 数据库表名 |  |
| oid | varchar(20) | 对象ID |  |
| owner | varchar(20) | 所有者 |  |
| sql | varchar(255) | 建表SQL |  |
| comments | varchar(255) | 注释 |  |

### dbc\_view\_list

此表结构是虚拟的，不可直接操作，参见章节“[获取view列表](#_获取view列表)”。

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 类型 | 描述 | 备注 |
| name | varchar(20) | 视图名称 |  |
| oid | varchar(20) | 对象ID |  |
| owner | varchar(20) | 所有者 |  |
| comments | varchar(255) | 注释 |  |

### dbc\_table\_schema

此表结构是虚拟的，不可直接操作，参见章节“[获取表结构](#_获取表结构)”。

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 类型 | 描述 | 备注 |
| name | varchar(40) | 字段名称 |  |
| type | varchar(40) | 字段类型 |  |
| defaultvalue | varchar(64) | 默认值 |  |
| isnull | char(1) | 是否为空，Y表示可以为空，N表示不为空 |  |
| comments | varchar(255) | 字段注释 |  |

### dbc\_table\_index

此表结构是虚拟的，不可直接操作，参见章节“[获取表索引](#_获取表索引)”。

索引名称、索引类型，索引ID，索引字段key，索引属性、是否唯一、是否主键；当索引为多字段联合索引时，key为空格分隔的多个值

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 类型 | 描述 | 备注 |
| name | varchar(20) | 索引名称 |  |
| type | varchar(20) | 索引类型 |  |
| fileds | varchar(128) | 索引字段 |  |
| ispk | char(1) | 是否主键，Y：是，N：否 |  |
| sql | varchar(512) | 索引创建SQL |  |

### dbc\_db\_list

此表结构是虚拟的，不可直接操作，参见章节“[查询集群节点信息](#_查询集群节点信息)”。

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 类型 | 描述 | 备注 |
| id | varchar(10) | 数据库节点ID |  |
| name | varchar(20) | 数据库节点逻辑名 |  |
| host | varchar(20) | 数据库服务器地址 |  |
| port | Integer | 数据库服务端口 |  |
| dbtype | varchar(20) | 数据库类型 |  |
| state | Integer | 数据库节点状态  0：在线  1：同步  2：保留  3：离线  4：故障 |  |

### dbc\_cdm\_list

此表结构是虚拟的，不可直接操作，参见章节“[查询CDM节点信息](#_查询CDM节点信息)”

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 类型 | 描述 | 备注 |
| host | Varchar(20) | 主机名 | Not NULL |
| ip | Varchar(20) | IP地址 | Not NULL |
| procnum | Integer | 启动进程数 | Not NULL |
| errnum | Integer | 异常进程数 | Not NULL |

### dbc\_cdm\_cfg

此表结构是虚拟的，不可直接操作，参见章节“ [配置管理](#_配置管理) ”

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 类型 | 描述 | 备注 |
| name | varchar(32) | 参数名称 | Not NULL |
| value | varchar(512) | 参数取值 | Not NULL |

参数名称见下表：

|  |  |
| --- | --- |
| 参数名称 | 描述 |
| thread\_pool\_size | 线程池大小 |
| connection\_pool\_size | 连接池大小 |
| general\_error | 一般错误码 |
| recoverable\_error | 可自动恢复类错误码 |
| unrecoverable\_error | 不可自动回复类错误码 |

### dbc\_lic

此表结构是虚拟的，不可直接操作，参见““。

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 类型 | 描述 | 备注 |
| Licnumber | Varchar(40) | 序列号 | Not NULL |
| User | Varchar(20) | 授权用户名 | Not NULL |
| Company | Varchar(20) | 授权公司名称 | Not NULL |
| Expire | Timestamp without time zone | 有效期，截止时间。  NULL表示无限制 |  |
| Session\_max | Integer | 最大连接数，0表示无限制 | Not NULL |
| Session\_warn | Integer | 连接数告警阀值 | Not NULL |

## 配置文件

# 系统性能

描述系统的性能需求以及系统的设计是如何满足性能需求的。

## 性能需求

* 关系型数据库支持4节点
* 关系型数据库集群倍增系数1.6

## 性能设计

# 参考资料

参考文献说明一般包括：编号、名称、作者、版本号、出版/发布时间。

－ 完毕 －