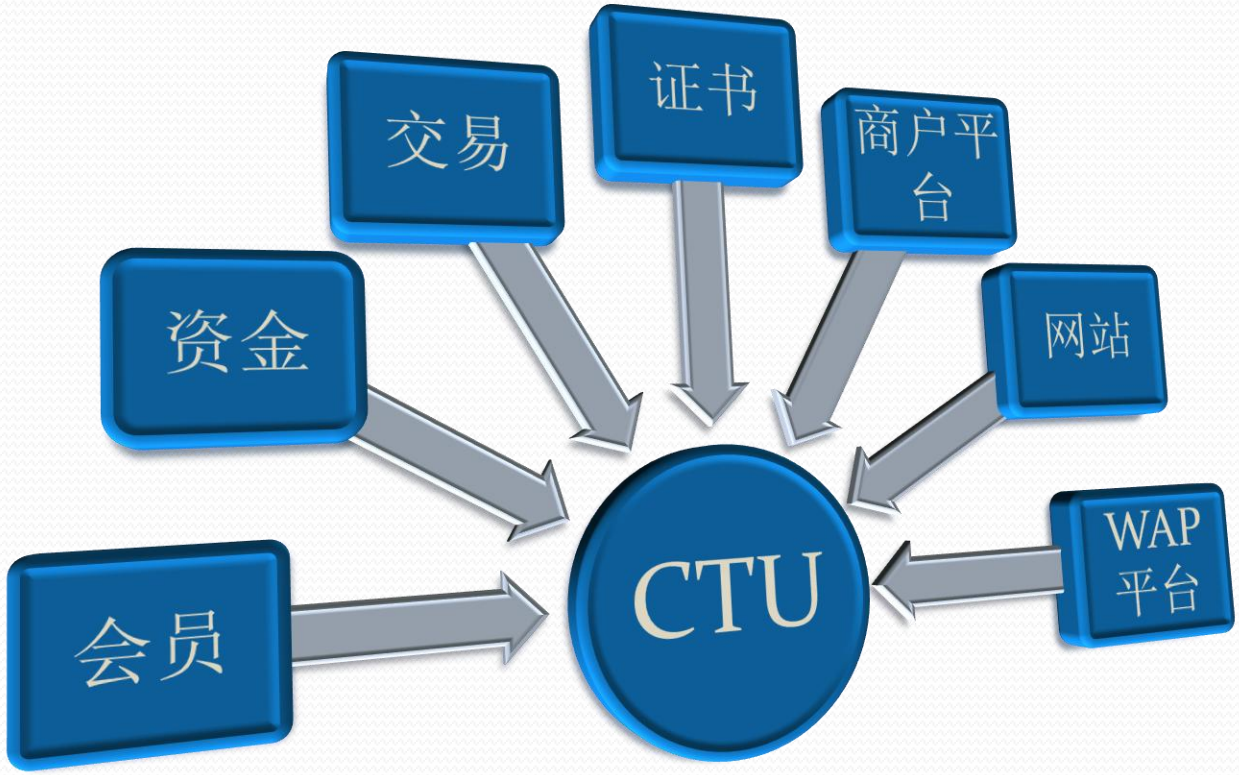
CTU系统介绍

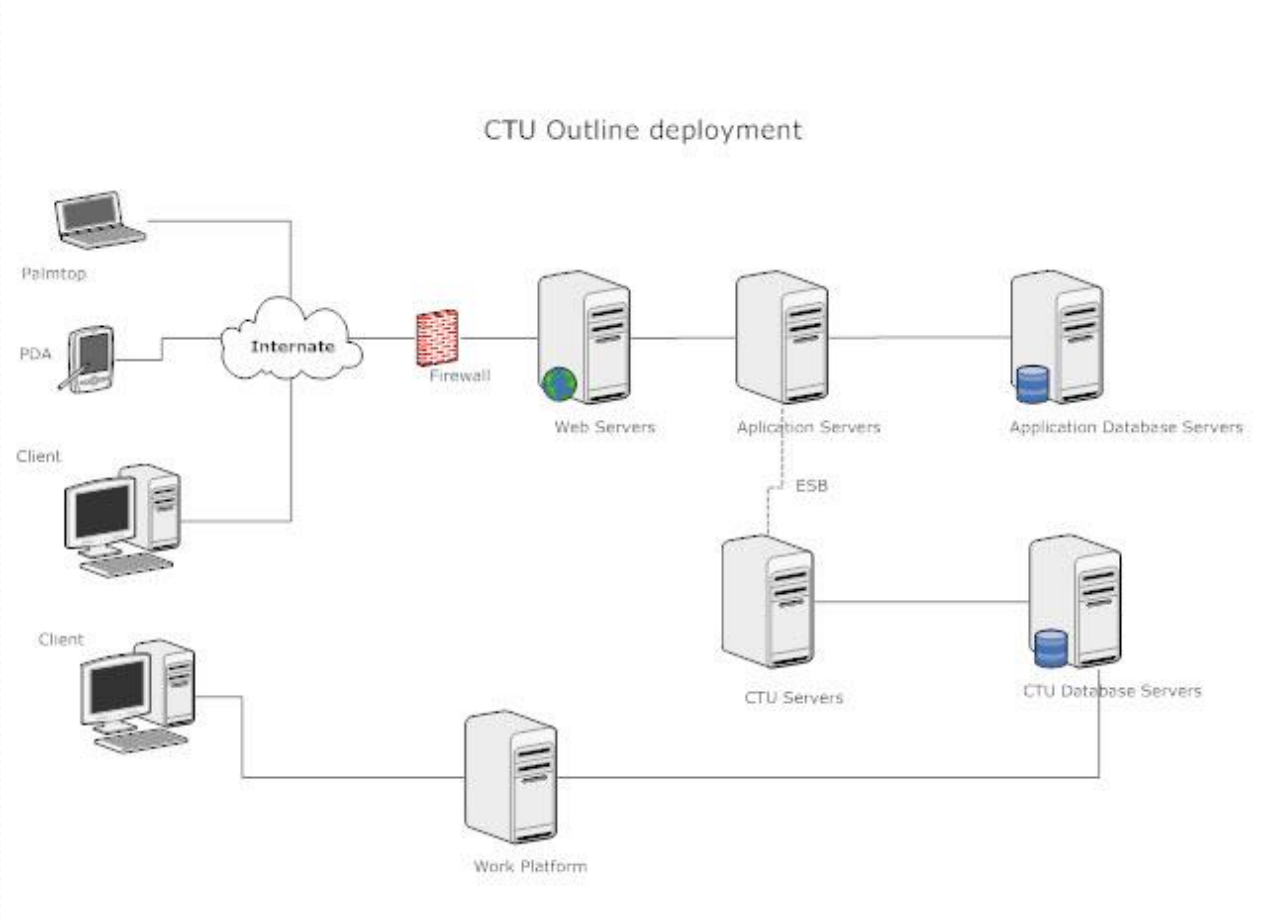
目录

- 别人眼中的CTU
- 我们眼中的CTU
- CTU系统范围
- CTU组成部分
  - 事件
  - 风险
  - 任务
  - 数据
    - Online数据
    - 历史数据
    - 汇总数据
  - 规则免疫
  - 规则
  - 规则监控

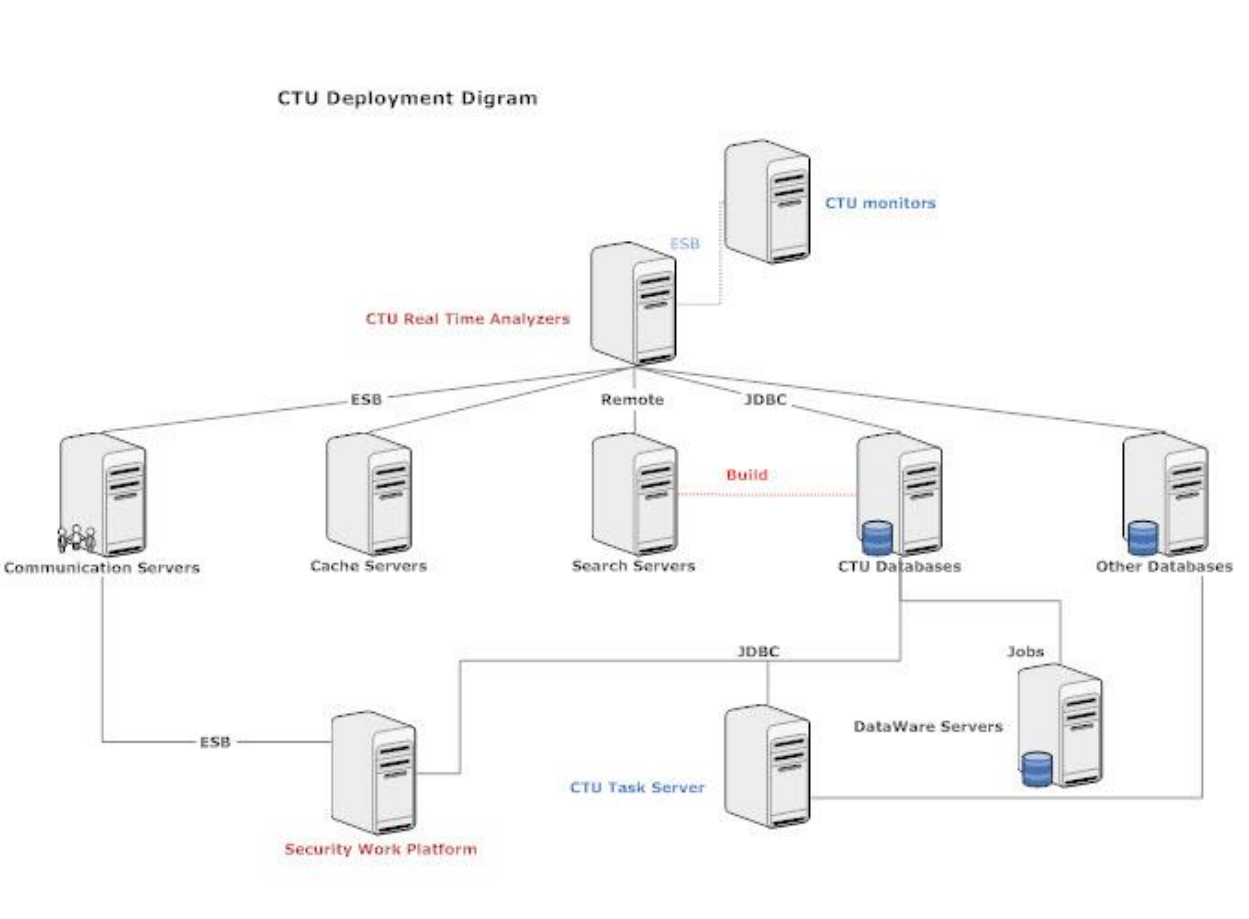
别人眼中的CTU



CTU全景


CTU Outline deployment
Palmtop
PDA
Client
Client
Internate
Firewall
Web Servers
Aplication Servers
Application Database Servers
ESB
CTU Servers
CTU Database Servers
Work Platform

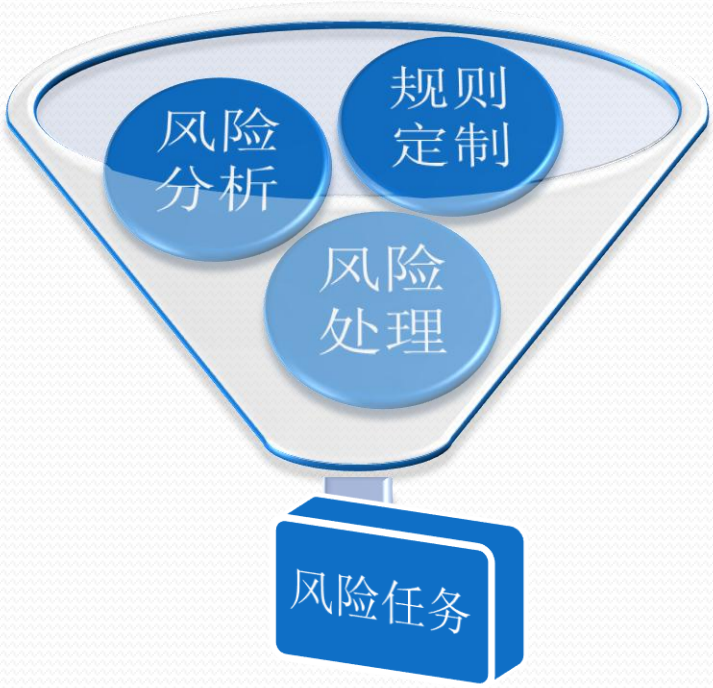
CTU全景


CTU Deployment Digram
CTU monitors
ESB
CTU Real Time Analyzers
ESB
Remote
JDBC
Build
Communication Servers
Cache Servers
Search Servers
CTU Databases
Other Databases
JDBC
Jobs
DataWare Servers
ESB
CTU Task Server
Security Work Platform

我们眼中的CTU

- 一套风险和任务处理系统
- 一套规则分析系统
- 一套风险数据提供系统

我们眼中的CTU

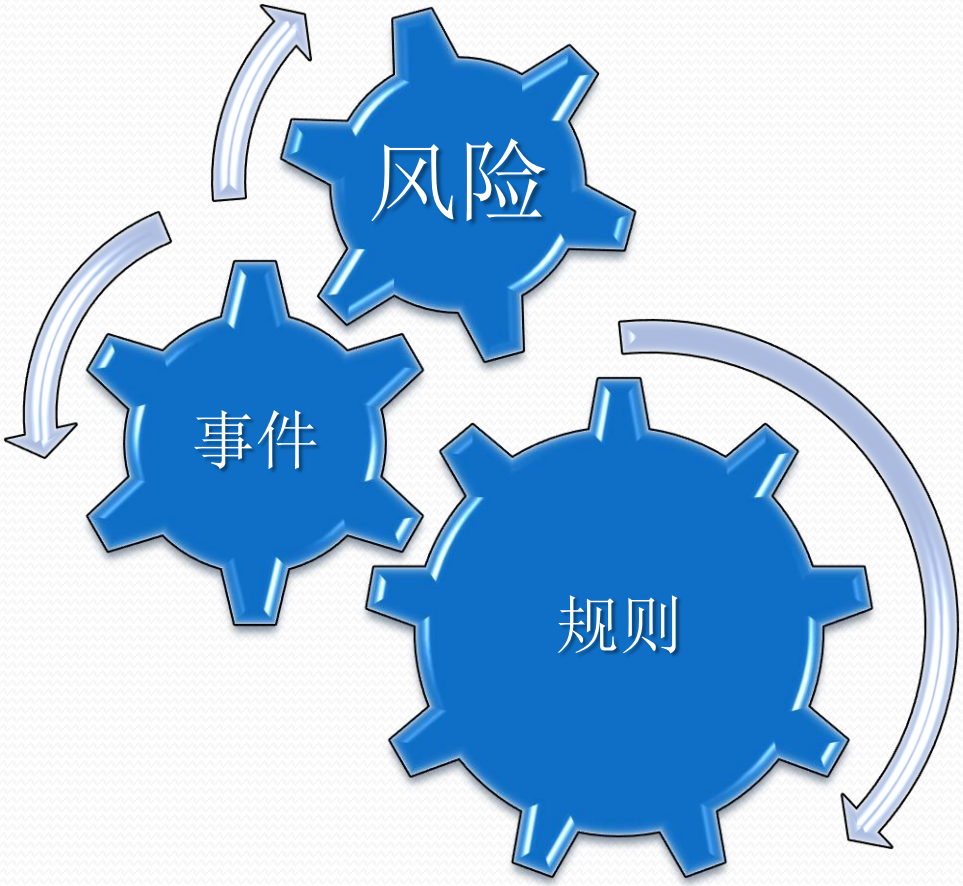


我们眼中的CTU

CTU系统范围

- 账户安全
  - 支付宝账户盗用
- 金融风险
  - 银行卡盗用
  - 套现
  - 洗钱
- 欺诈控制
  - 交易欺诈
  - 批量帐户控制

CTU运行时



•线下代付金额可以体现 已经在8月28号发布

CTU组成—组件

事件 | 事件转换 | 事件存储 | 场景生成 | 因子生成

事件查询 | 风险处理 | 风险备注 | 邮件模块 | 动作备注

规则免疫 | 任务分配 | 条件解析 | 监控报警 | 规则监控

风险分析接口 | 处理动作 | 数据缓存

事件

- CTU是事件驱动型的。事件是CTU分析的输入。
  事件由属性组成，不同事件有不同的属性。
- 事件有历史事件和在线事件。CTU数据保存情况见附件
- 事件属性见附件
- 事件属性文档需要持续更新

事件属性说明

- 用户ID:事件发起方ID（userId）
- 用户帐户：触发事件的帐户号（accountNo)
- 操作者:触发事件的操作者(operatorId)参考 com.iwallet.biz.core.enums. OperatorEnum。
- 外部ID： 比如旺旺ID(externalId)。
- 事件IP：触发者IP(clientIp)。
- 事件MAC:触发者MAC(clientMac）。
- 客户端ID:触发者永久cookieId（ clientId)

事件属性说明

- 服务器ID:事件发生的原始服务器ID(serverId)。
- 模块ID：事件发生的业务模块，比如会员，交易（moduleId）参考com.beyond.biz.event.enums. EventModuleEnum。(需要迁移到CTU，并用标准enum）
- 事件发生时间：事件发生时服务器时间（gmtOccure）
- 事件类型：事件分发起中用到，类型为N的不会被规则引擎分析。（eventType）参考com.beyond.biz.event.enums.EventTypeEnum。（需要迁移到CTU，并用标准enum）
- 事件名称：表示操作的业务（eventName）参考com.beyond.biz.event.enums. EventNameEnum。（需要迁移到CTU，并用标准enum和后台共用)

事件属性说明

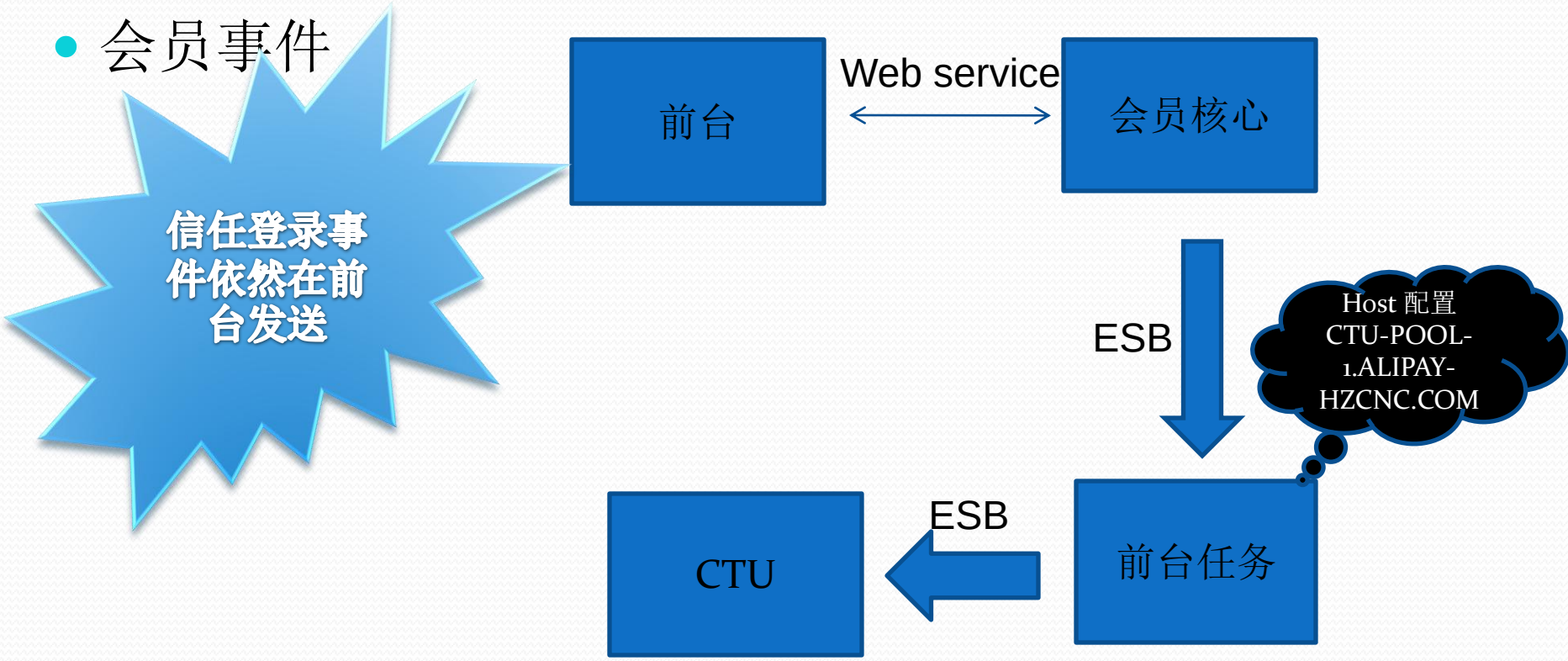
- 事件主对象：用于抽象一个事件描述的主体，如交易为交易号，充值为流水号，提现为提现号等（eventMainTarget）。（有待标准化）
- 事件主对象类型：用于说明主对象的类型，比说交易号，帐户号等（eventMainTargetType）。参考 com.beyond.biz.event.enums. EventTargetTypeEnum（需要移植到CTU，并使用标准enum）
- 事件辅助对象：用于抽象描述事件处理的对方，如交易对方帐户好，代充对方帐户号等（eventSuppTarget）。
- 事件辅助对象类型：用户描述辅助对象的类型。比如交易好，帐户号等（eventSuppTargetType）。参考 com.beyond.biz.event.enums. EventTargetTypeEnum(需要移植到CTU，并使用标准enum）（有待标准化）

事件属性说明

- 会话ID：此会话ID并不和前台的SESSIONID一致。但是用于说明一个会话期（sessionId）
- 属性：此为扩展属性，可以提供事件本身不具备的表述信息，如事件金额，手机号，是否证书等信息。此扩张属性给业务提供更大支持。（有待标准化）

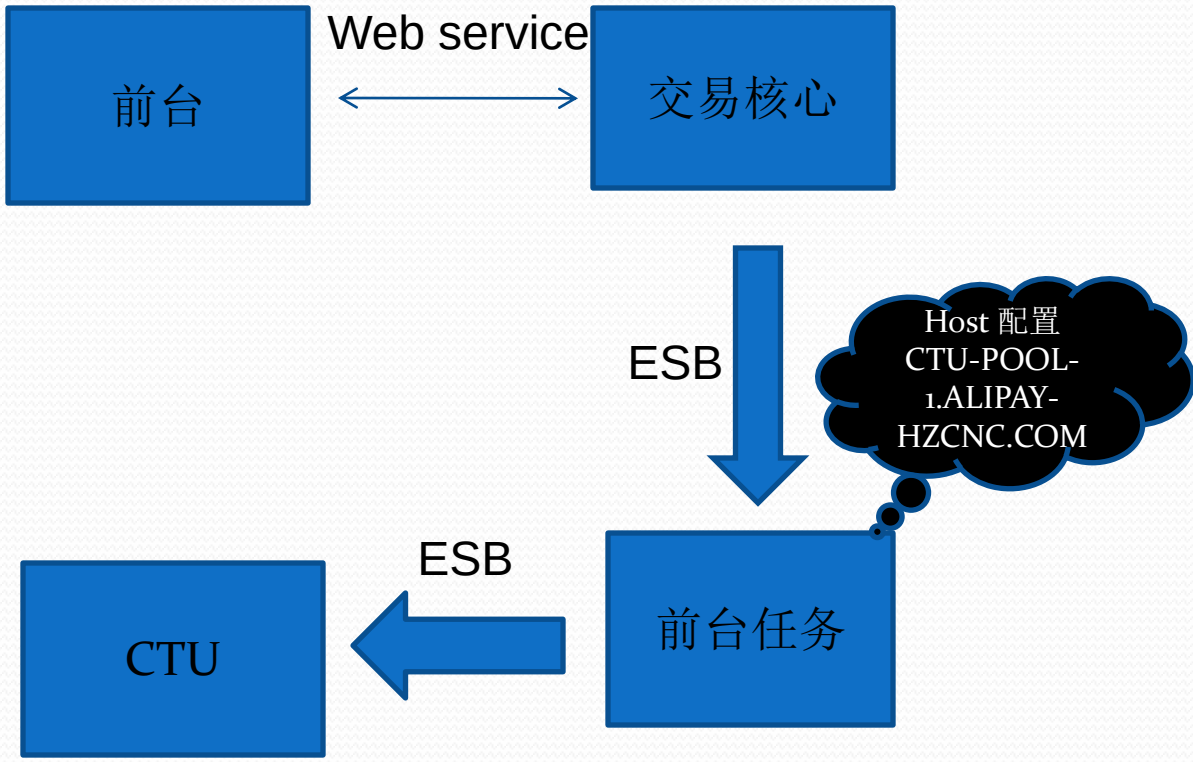
CTU事件—异步

- 会员事件



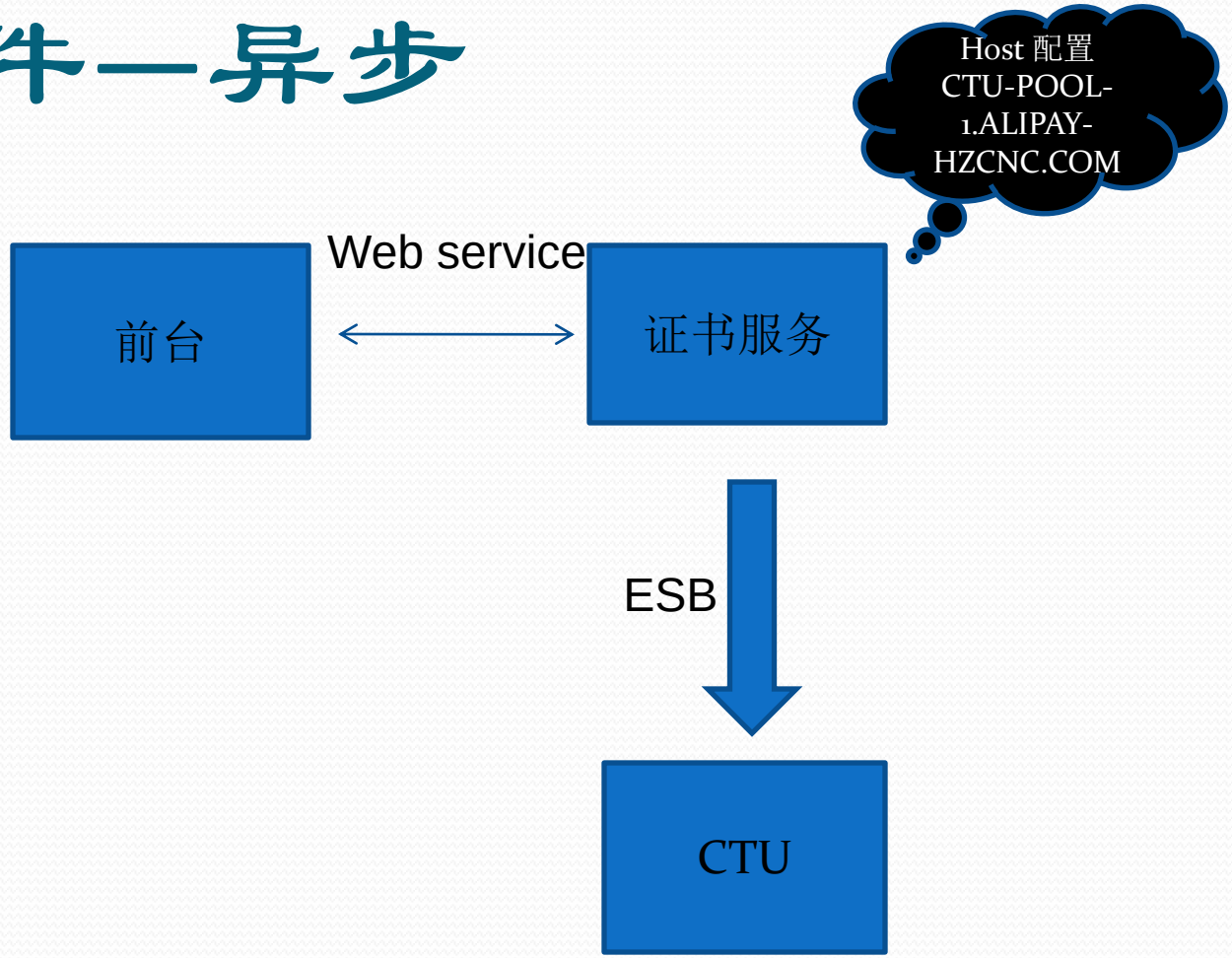
CTU事件—异步

- 交易事件



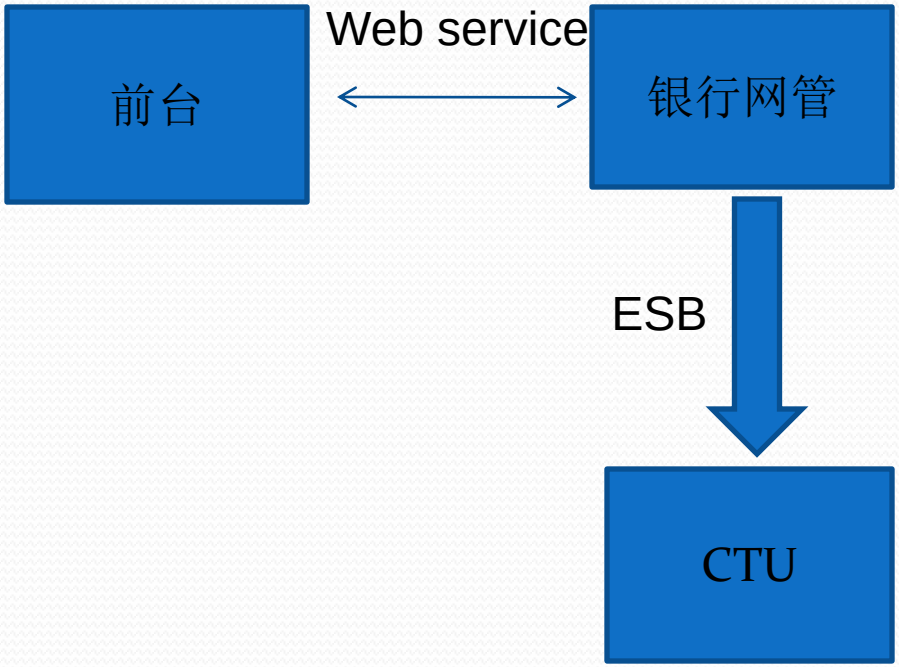
CTU事件—异步

- 证书事件



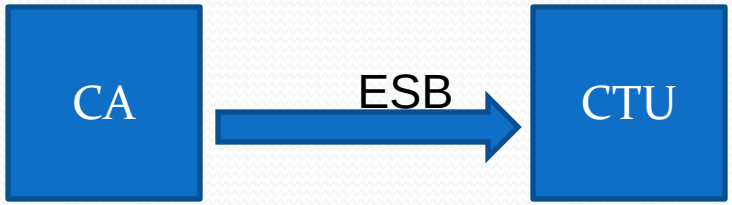
CTU事件—异步

- 资金事件



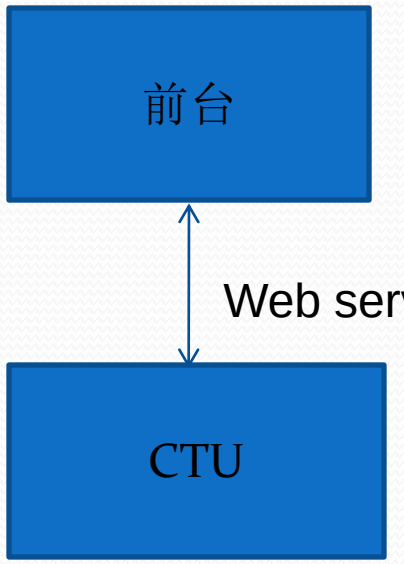
CTU事件—异步

- 商户平台



CTU事件--同步

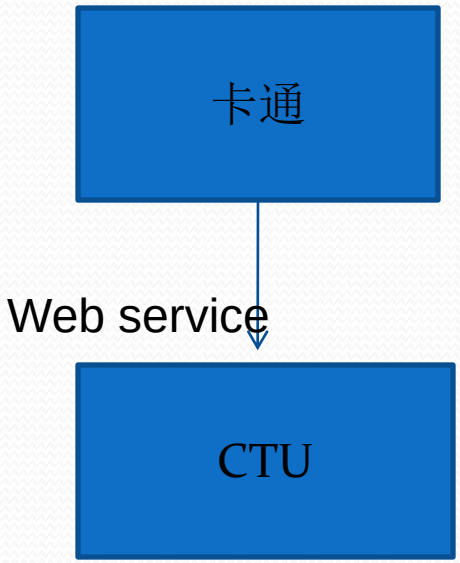
- CTU防火墙



CTU防火墙范围：
1、支付宝登录（暂时关闭）
2、淘宝信任登录（暂时关闭）
3、站内、站外即时到帐付款。
4、淘宝自动发货交易付款。

CTU事件同步--卡通提现

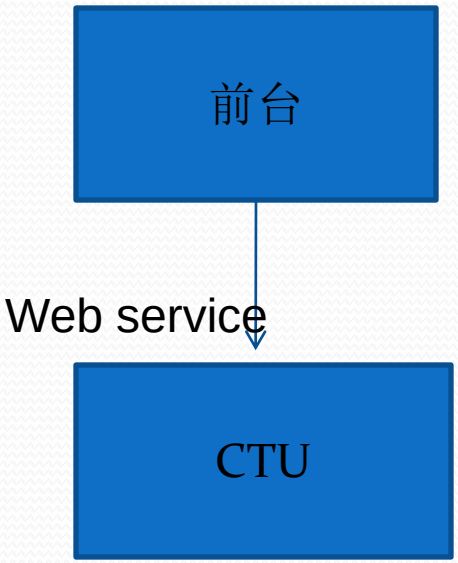
- 如果CTU服务失败，卡通提现失败



需要资金线改造不和CTU有服务上的紧耦合

CTU事件同步—信用卡付款

- 如果CTU服务失败，默认判断无风险。



CTU事件同步—接口改造

- 所有业务对象事件集成同一个父类。
- 提供一个方法，分析所有类型事件。

CTU分析选型

- 异步接口
  - 无页面业务
  - Web服务器并发考虑？
- 同步接口

规则

- 由一以上的条件和一个输出结论组成。
- 每个条件都有一个true或false的结果。
- 规则的输出是一个风险。
- 如果一个规则的所有条件被满足，则输出是一个风险。
- 缺点
  - 多条规则条件重复

规则—条件

- 规则中条件需要标准化。如：所有事件类型的金额为同一扩展名称。
- 符合本地缓存命名方式。条件中需要查询数据或则远程调用服务的名称需要标准化。DAO以*DAO命名方式。远程服务以*ServiceClient方式命名。
- 条件中下拉选项的Enum如果依赖外部系统，则依赖外部系统的Enum，否则由CTU提供。以达到共用。
- 条件不再需要时，需要删除ctu-condition-factor.xml中的配置。需要确认所有运行中的规则都不再使用此条件。

规则引擎

- 规则引擎是CTU的处理器，处理由CTU后台定制界面定制的规则和业务发送的事件。
- 规则引擎分为2级：风险因子和风险场景。

  优点：
  - 可以分级实现数据缓存。
  - 可以实现一个场景多次事件查询。
  - 可以实现场景共用多个因子，减少规则复杂度。

  缺点：
  - 场景对因子依赖，关系变复杂。
  - 场景依赖因子数据，因此并不是解耦。

风险

- 风险是CTU的输出。
  - 风险因子产生的风险
  - 风险场景产生的风险

都有运行和试运行
分属不同的数据表
原则上场景可以引用场景和因子。
因子不引用因子和场景。

风险

- 风险因子产生的风险（因子风险）
  - 因子风险可以被场景引用。
  - 因子风险只记录，不动作。
  - 因子尽量做到业务的最大共用。
  - 可以不保存到数据库。
- 风险场景产生的风险（场景风险）
  - 可以引用运行状态的因子。
  - 可以引用因子的事件抽取。
  - 可以定义动作。
  - 必须保存到数据库。

风险处理

- 风险处理是CTU的四肢。处理风险，减低损失，保护资金安全。
- 风险的处理方式也决定事件是否需要同步或则异步判断。
- 风险处理根据人区分，区分受害者和被害者。

规则免疫

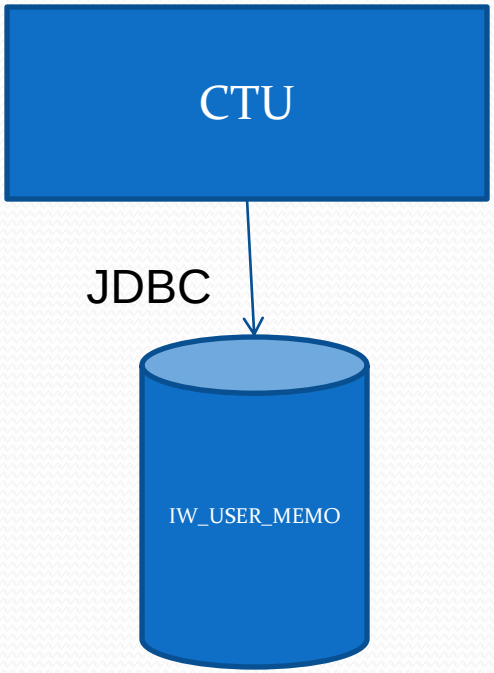
- 根据人和触犯的规则进行免疫。
- 按照触发的风险规则按照高到低免疫。

风险处理

- 沟通执行器
  - 发送邮件
  - 发送短信

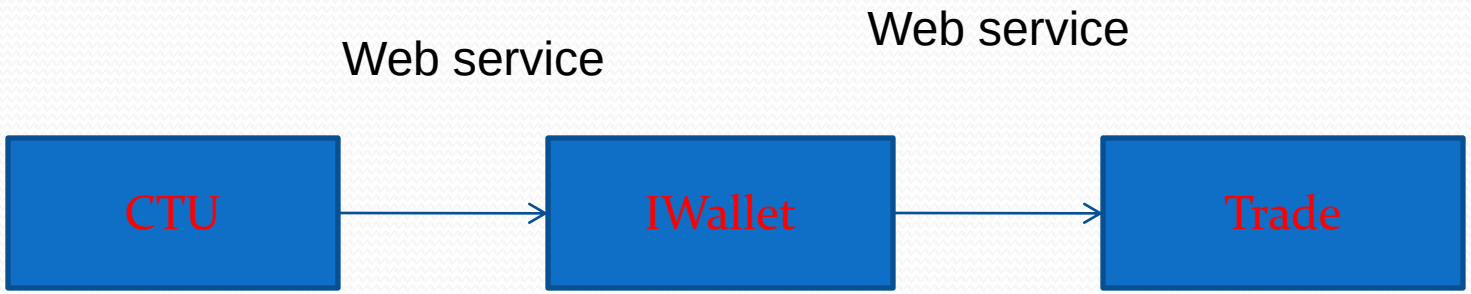
风险动作执行器

- 备注执行器



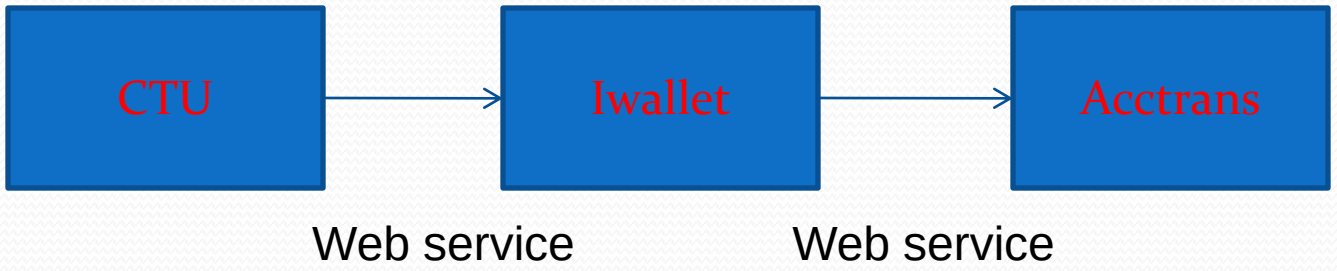
风险动作执行器

- 交易动作执行器
  - 关闭交易



风险动作执行器

- 账户动作执行器
  - 冻结帐户
  - 提现失败
  - 禁止余额支付
  - 禁止余额支付前台开启

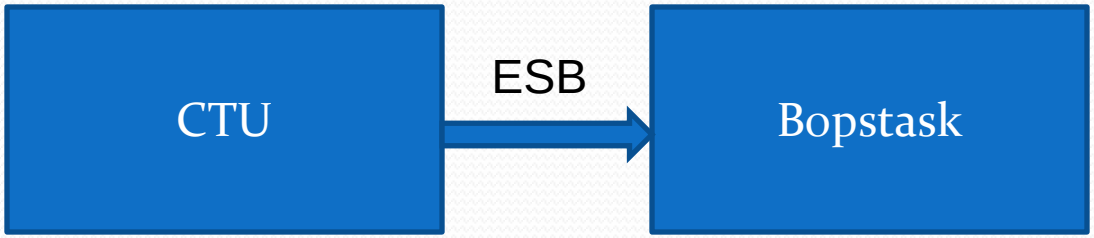


风险动作执行器

- 冻结余额执行器
  - 冻结帐户余额

风险动作执行器

- 冲值退回执行器

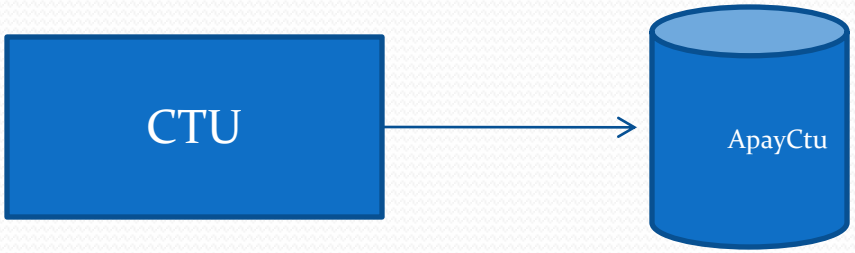


风险动作执行器

- 关闭卖家信贷服务
  - 关闭卖家信贷服务

风险动作执行器

- 记录可信IP、MAC
  - 记录用户的可信ip和mac

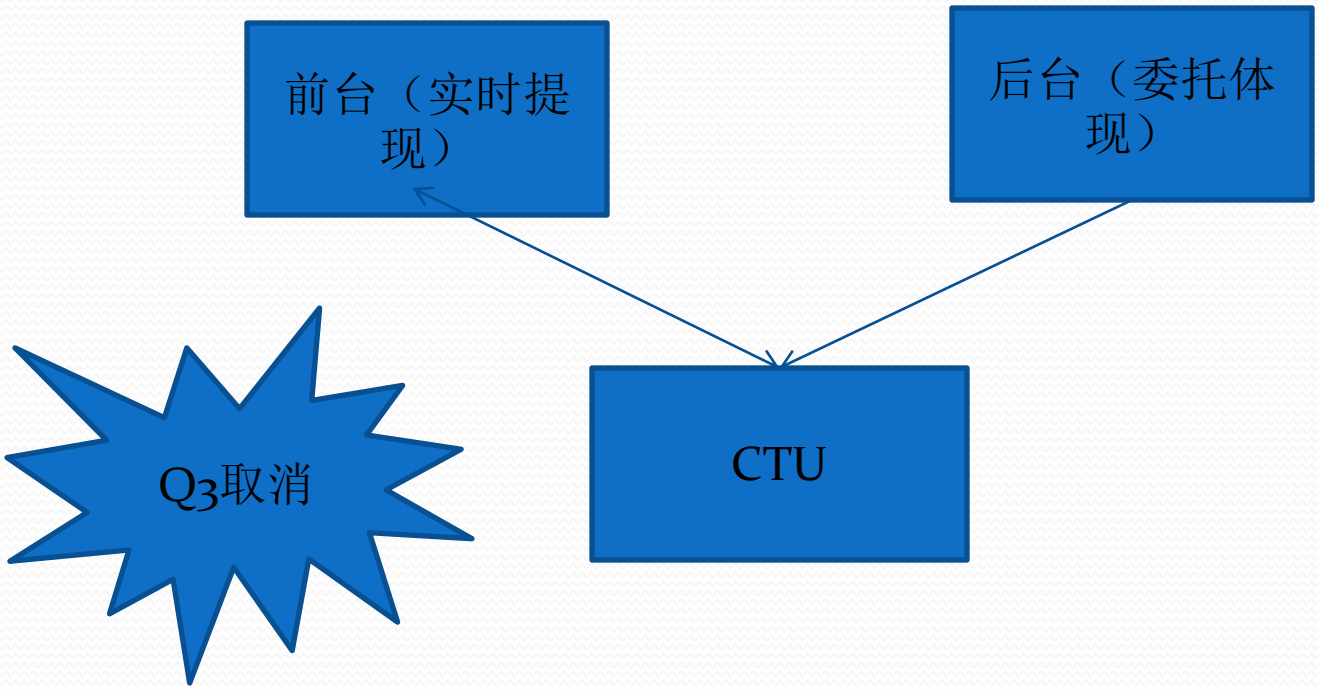


任务

- 任务由风险场景生成
- 任务需要业务人员核查
- 任务由场景风险产生
- 任务根据值班人员进行分派
- 任务平台
  - 待申领
  - 待审核
  - 完结

CTU同步--接口

- 可提现接口



CTU同步--接口

- CTU防火墙
  - 范围
    - 自动发货付款
    - 即时到帐付款
  - 事件
    - PayByAccount --买家尝试付款事件
    - payByBank --买家使用网银付款返回事件

CTU防火墙可以区分用户尝试用余额或则卡通付款。
所有通过网银付款的回调都会发送payByBank事件。

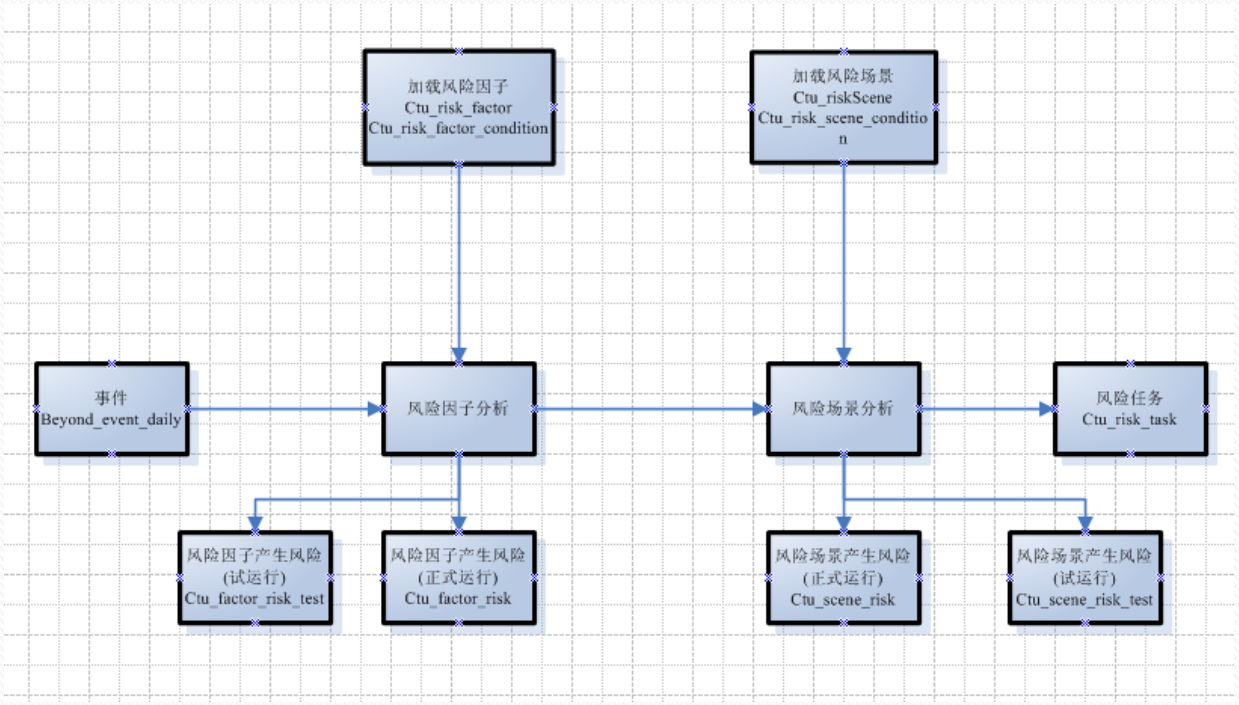
CTU同步--接口

- 性能
  - 合理的方法性能值定义。
  - 条件中减低远程调用
  - 减少对大表的查询，帐务日志表，交易日志表等。
  - 减少规则的事件抽取。

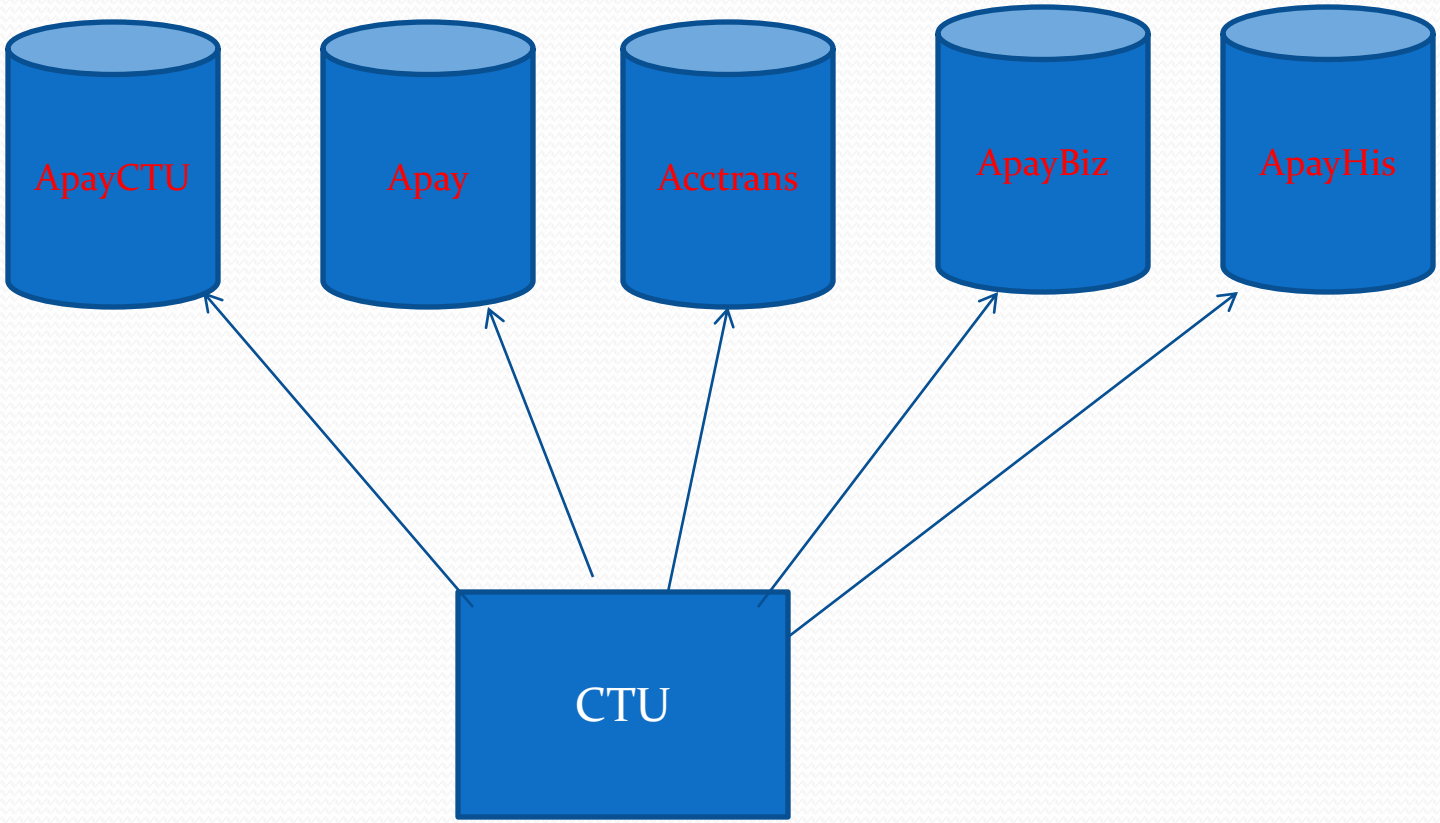
数据

- 分类
  - Online数据
  - 历史数据
    - 事件表（表名按月）
    - 因子风险表（正式运行和试运行）
  - 汇总数据
    - 数据仓库
      - 买卖家列表
      - 买家交易历史表
      - 卖家交易历史表
      - 用户信息表（mac，ip，mobile phone...）

数据

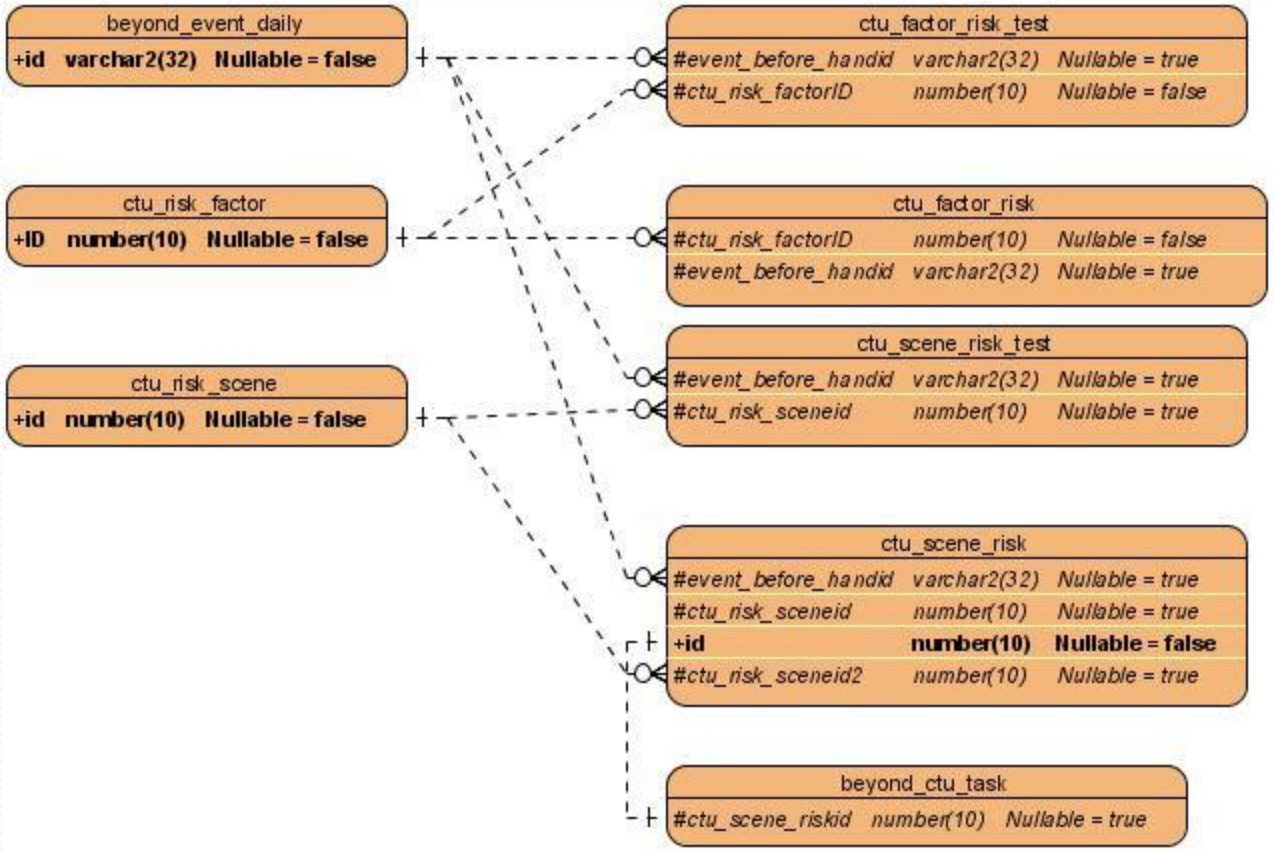

加载风险因子
Ctu_risk_factor
Ctu_risk_factor_condition
加载风险场景
Ctu_riskScene
Ctu_risk_scene_conditio
n
事件
Beyond_event_daily
风险因子分析
风险场景分析
风险任务
Ctu_risk_task
风险因子产生风险
(试运行)
Ctu_factor_risk_test
风险因子产生风险
(正式运行)
Ctu_factor_risk
风险场景产生风险
(正式运行)
Ctu_scene_risk
风险场景产生风险
(试运行)
Ctu_scene_risk_test

数据—范围



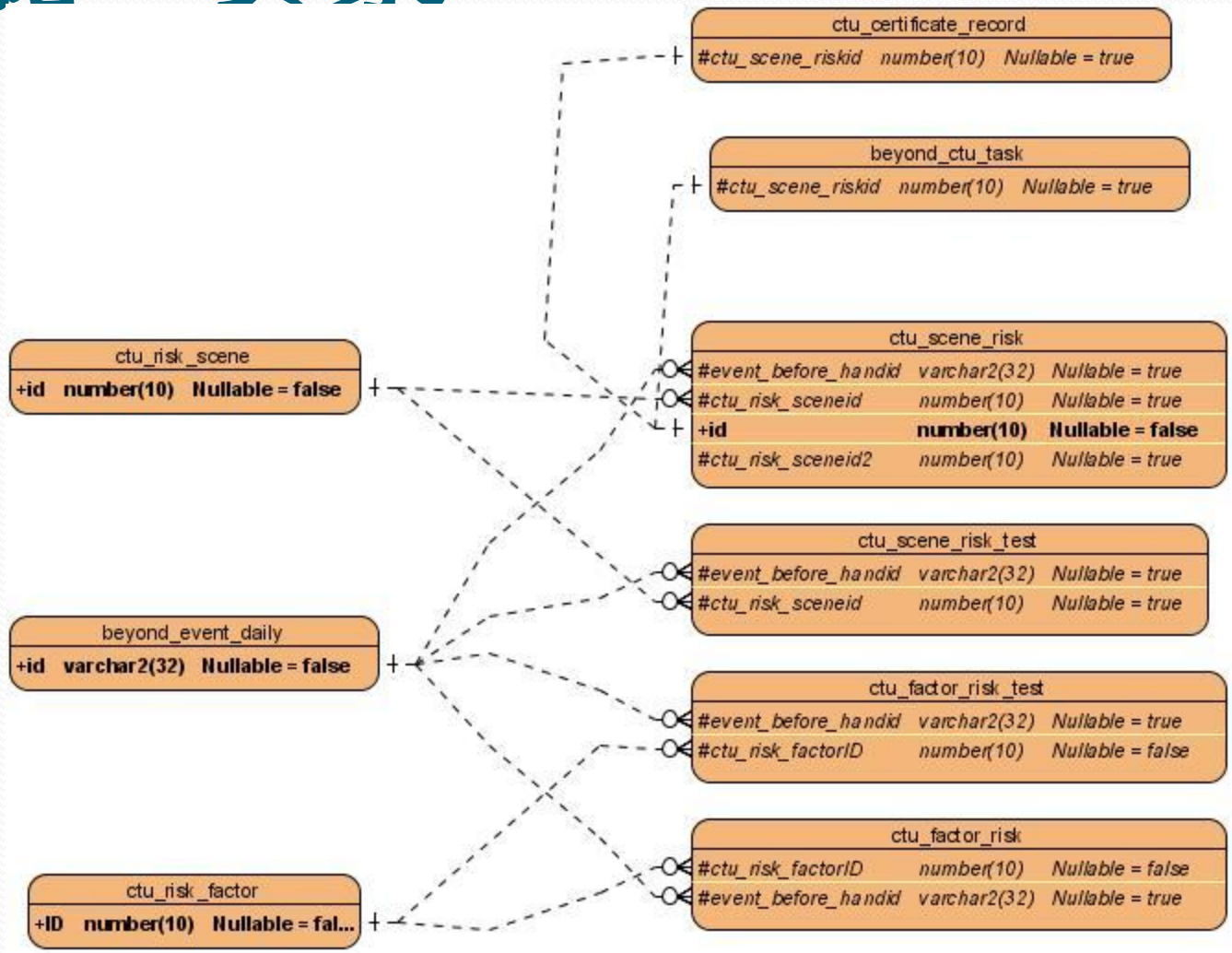
数据—关系

异步风险关系


beyond_event_daily
+id varchar2(32) Nullable = false
ctu_risk_factor
+ID number(10) Nullable = false
ctu_risk_scene
+id number(10) Nullable = false
ctu_factor_risk_test
#event_before_handid varchar2(32) Nullable = true
#ctu_risk_factorID number(10) Nullable = false
ctu_factor_risk
#ctu_risk_factorID number(10) Nullable = false
#event_before_handid varchar2(32) Nullable = true
ctu_scene_risk_test
#event_before_handid varchar2(32) Nullable = true
#ctu_risk_sceneid number(10) Nullable = true
ctu_scene_risk
#event_before_handid varchar2(32) Nullable = true
#ctu_risk_sceneid number(10) Nullable = true
+id number(10) Nullable = false
#ctu_risk_sceneid2 number(10) Nullable = true
beyond_ctu_task
#ctu_scene_riskid number(10) Nullable = true

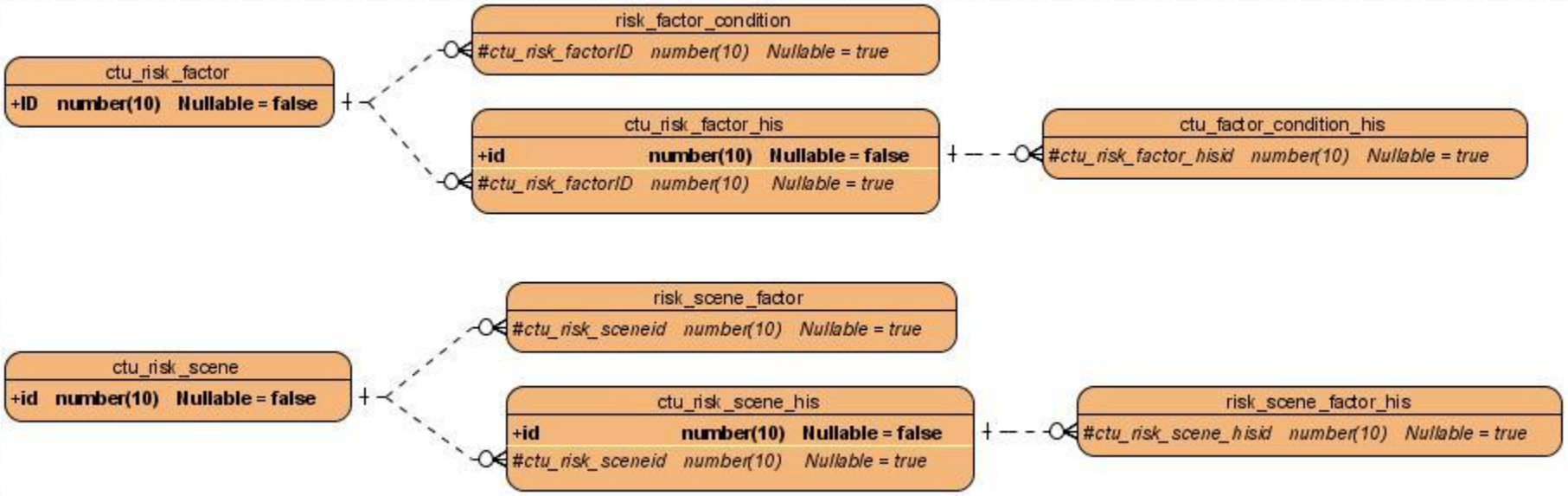
数据—关系

- 同步风险关系


ctu_certificate_record
#ctu_scene_riskid number(10) Nullable = true
beyond_ctu_task
#ctu_scene_riskid number(10) Nullable = true
ctu_risk_scene
+id number(10) Nullable = false
ctu_scene_risk
#event_before_handid varchar2(32) Nullable = true
#ctu_risk_sceneid number(10) Nullable = true
+id number(10) Nullable = false
#ctu_risk_sceneid2 number(10) Nullable = true
ctu_scene_risk_test
#event_before_handid varchar2(32) Nullable = true
#ctu_risk_sceneid number(10) Nullable = true
beyond_event_daily
+id varchar2(32) Nullable = false
ctu_factor_risk_test
#event_before_handid varchar2(32) Nullable = true
#ctu_risk_factorID number(10) Nullable = false
ctu_factor_risk
#ctu_risk_factorID number(10) Nullable = false
#event_before_handid varchar2(32) Nullable = true
ctu_risk_factor
+ID number(10) Nullable = fal...

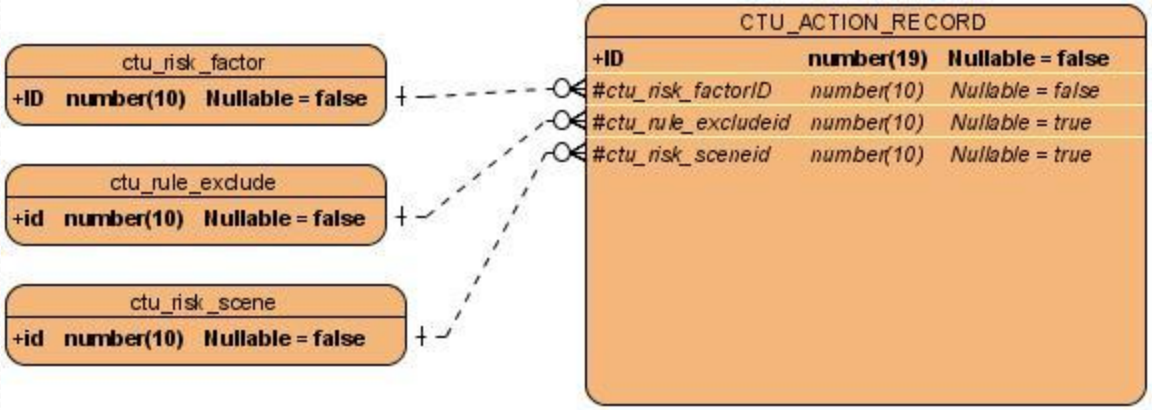
数据—关系

- 规则


ctu_risk_factor
+ID number(10) Nullable = false
risk_factor_condition
#ctu_risk_factorID number(10) Nullable = true
ctu_risk_factor_his
+id number(10) Nullable = false
#ctu_risk_factorID number(10) Nullable = true
ctu_factor_condition_his
#ctu_risk_factor_hisid number(10) Nullable = true
ctu_risk_scene
+id number(10) Nullable = false
risk_scene_factor
#ctu_risk_sceneid number(10) Nullable = true
ctu_risk_scene_his
+id number(10) Nullable = false
#ctu_risk_sceneid number(10) Nullable = true
risk_scene_factor_his
#ctu_risk_scene_hisid number(10) Nullable = true

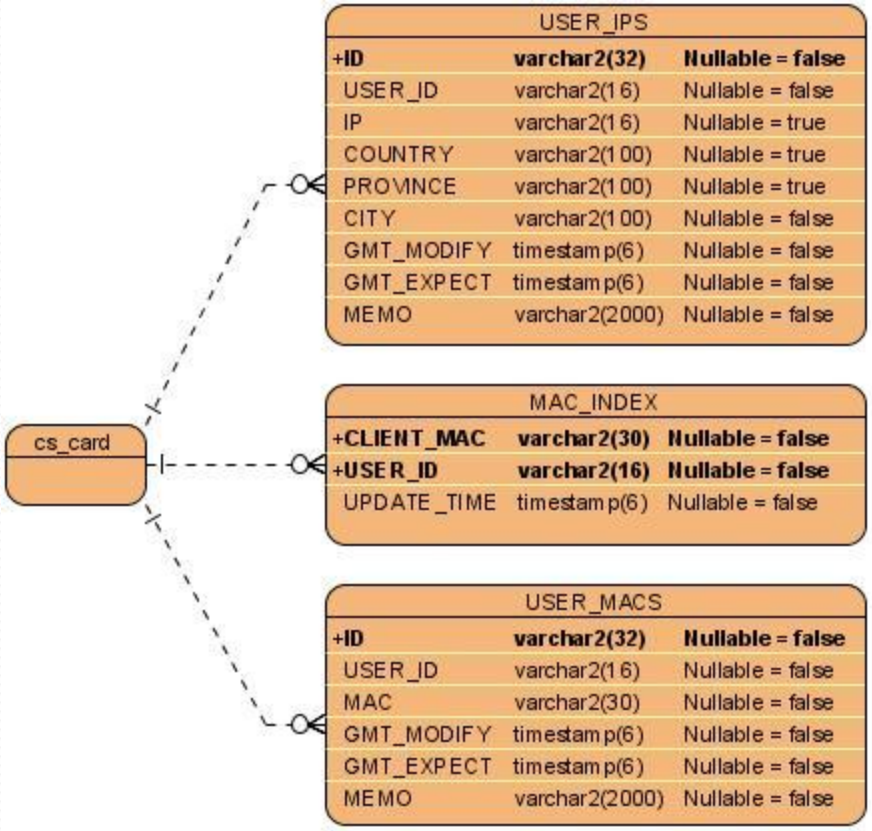
数据—关系

- 操作日志


ctu_risk_factor
+ID number(10) Nullable = false
ctu_rule_exclude
+id number(10) Nullable = false
ctu_risk_scene
+id number(10) Nullable = false
CTU_ACTION_RECORD
+ID number(19) Nullable = false
#ctu_risk_factorID number(10) Nullable = false
#ctu_rule_excludeid number(10) Nullable = true
#ctu_risk_sceneid number(10) Nullable = true

数据—关系

- 应用数据

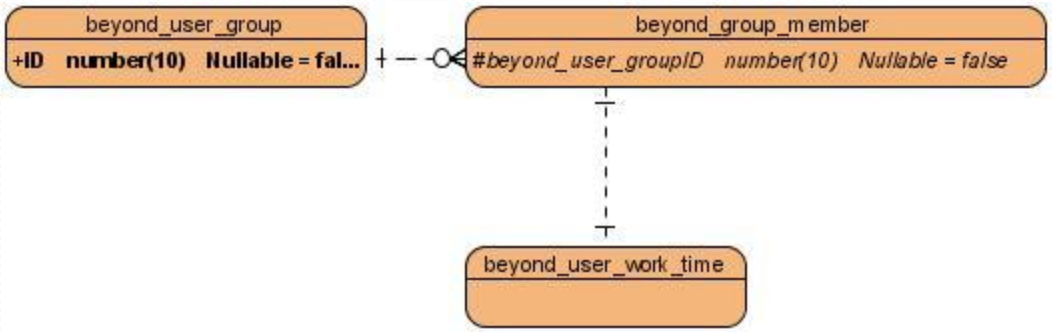

cs_card
USER_IPS
+ID varchar2(32) Nullable = false
USER_ID varchar2(16) Nullable = false
IP varchar2(16) Nullable = true
COUNTRY varchar2(100) Nullable = true
PROVINCE varchar2(100) Nullable = true
CITY varchar2(100) Nullable = false
GMT_MODIFY timestamp(6) Nullable = false
GMT_EXPECT timestamp(6) Nullable = false
MEMO varchar2(2000) Nullable = false
MAC_INDEX
+CLIENT_MAC varchar2(30) Nullable = false
+USER_ID varchar2(16) Nullable = false
UPDATE_TIME timestamp(6) Nullable = false
USER_MACS
+ID varchar2(32) Nullable = false
USER_ID varchar2(16) Nullable = false
MAC varchar2(30) Nullable = false
GMT_MODIFY timestamp(6) Nullable = false
GMT_EXPECT timestamp(6) Nullable = false
MEMO varchar2(2000) Nullable = false

数据—关系

- 数据仓库

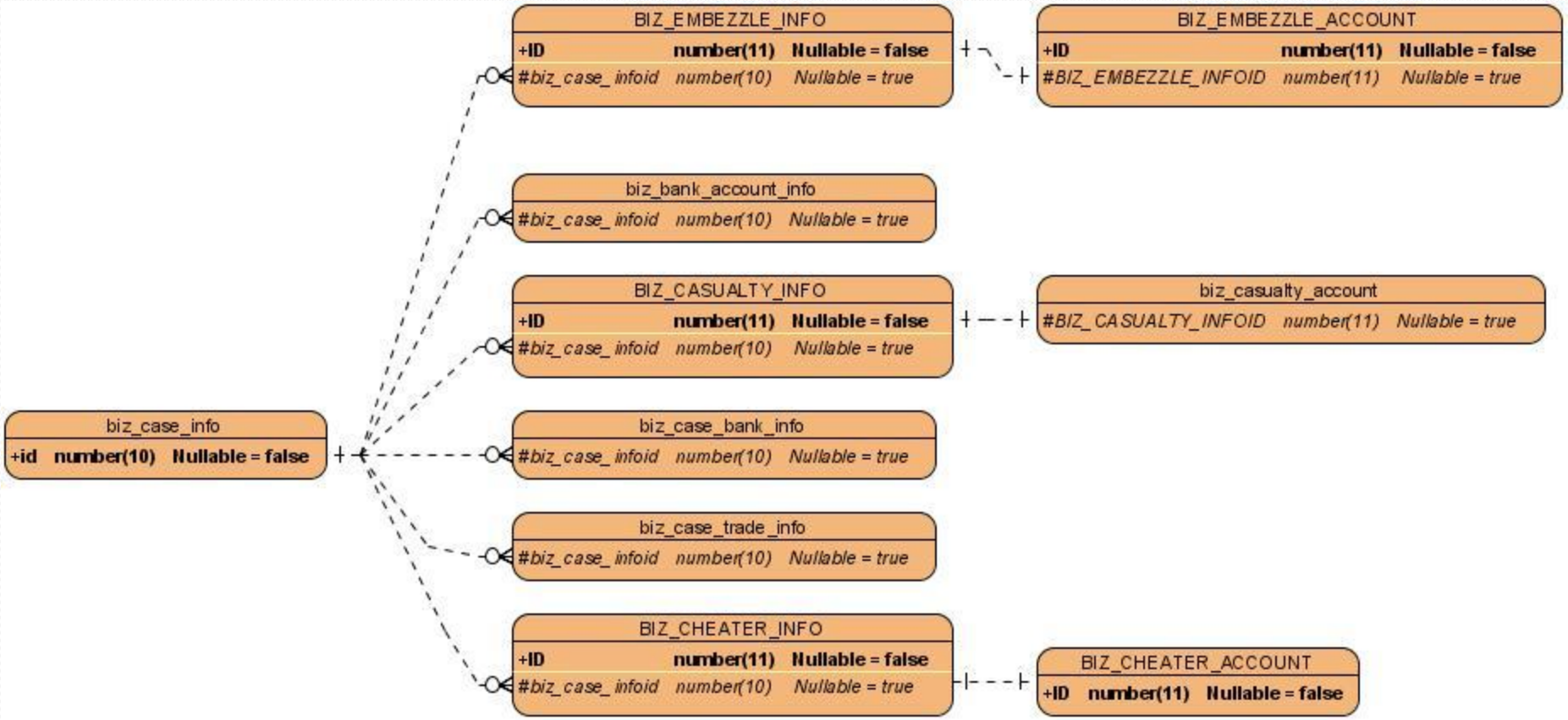
数据—关系

- 任务分配


beyond_user_group
+ID number(10) Nullable = fal...
beyond_group_member
#beyond_user_groupID number(10) Nullable = false
beyond_user_work_time

数据—关系

- 案件库


BIZ_EMBEZZLE_INFO
+ID number(11) Nullable = false
#biz_case_infoid number(10) Nullable = true
BIZ_EMBEZZLE_ACCOUNT
+ID number(11) Nullable = false
#BIZ_EMBEZZLE_INFOID number(11) Nullable = true
biz_bank_account_info
#biz_case_infoid number(10) Nullable = true
BIZ_CASUALTY_INFO
+ID number(11) Nullable = false
#biz_case_infoid number(10) Nullable = true
biz_casualty_account
#BIZ_CASUALTY_INFOID number(11) Nullable = true
biz_case_info
+id number(10) Nullable = false
biz_case_bank_info
#biz_case_infoid number(10) Nullable = true
biz_case_trade_info
#biz_case_infoid number(10) Nullable = true
BIZ_CHEATER_INFO
+ID number(11) Nullable = false
#biz_case_infoid number(10) Nullable = true
BIZ_CHEATER_ACCOUNT
+ID number(11) Nullable = false

数据--表说明

- 见附件

SCHEMA	表名	在线	历史
ApayBiz	Beyond_Group_Member	永久	/
	Beyond_User_Group	永久	/
	BEYOND_USER_WORK_TIME	1年	/
	CTU_ACTION_RECORD	1年	/
	SYSTEM_PARAMETER	永久	/
Apay	BEYOND_CTU_MONITOR_LOG	1年	/
	BEYOND_CTU_MONITOR_RULE	永久	/
	BEYOND_CTU_RULE_GROUP	永久	/
	BEYOND_CTU_RULE_HIS	不再使用	/
ApayCTU	BEYOND_CTU_TASK	永久	暂定永久
	CTU_CERTIFICATE_RECORD	1年	/

数据—存储周期

SCHEMA	表名	在线	历史
ApayCTU	CTU_FACTOR_RISK	30天	永久
	CTU_FACTOR_RISK_TEST	30天	1年以上
	CTU_SCENE_RISK	永久	/
	CTU_SCENE_RISK_TEST	30天	1年以上
	CTU_RISK_FACTOR	永久	/
	CTU_RISK_FACTOR_HIS	6个月	1年以上
	CTU_RISK_SCENE	永久	/
	CTU_RISK_SCENE_HIS	6个月	1年以上
	CTU_RULE_EXCLUDE	1年	1年以上
	RISK_FACTOR_CONDITION	永久	/
	RISK_FACTOR_CONDITION_HIS	6个月	1年以上

数据—存储周期

SCHEMA	表名	在线	历史
ApayCTU	RISK_SCENE_FACTOR	永久	/
	RISK_SCENE_FACTOR_HIS	6个月	1年以上
	USER_IPS	永久	/
	USER_MACS	永久	/
	MAC_INDEX	永久	/
	BEYOND_EVENT_DAILY	30天	1年以上
	CTU_BS_LIST_CURRENT	按天同步	
	CTU_HIS_TRADE_CURRENT	按天同步	
	CTU_HIS_TRADE_SELLER	按天同步	
ApayMKT	CTU_NEED_DATA	按天同步	/
ApayctuHis	Beyond_event_daily	保留一年	/

规则

- 规则组

  采用drools规则引擎。用一个xml文件表示。由多个规则组成。

样式：

```
<?xml version="1.0" encoding="GB2312"?>
<rule-set name="CTU规则"
        xmlns="http://drools.org/rules"
        xmlns:java="http://drools.org/semantics/java"
        xmlns:ctu="http://alipay.com/xsd/ctu"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:schemaLocatation="http://drools.org/rules rules.xsd http://drools.org/semantics/java java.xsd">
      <java:import>com.ctu.system.define.domain.*</java:import>
   <java:import>com.ctu.system.define.cache.*</java:import>
   <java:import>com.beyond.biz.event.domain.*</java:import>
   <java:import>com.beyond.biz.event.EventFinder</java:import>
   <java:import>com.beyond.biz.event.enums.*</java:import>
   <java:import>com.iwallet.biz.core.enums.OperatorEnum</java:import>
   <java:import>com.beyond.biz.event.util.*</java:import>
   <java:import>com.ctu.system.define.context.CTUContextHolder</java:import>
```

规则

```
<java:import>com.ctu.system.define.tool.*</java:import>
<java:import>com.iwallet.biz.common.util.DateUtil</java:import>
<java:import>com.alibaba.common.lang.StringUtil</java:import>
<java:import>com.alibaba.common.lang.ArrayUtil</java:import>
<java:import>java.util.*</java:import>
<application-data identifier="ipTool">IPTool</application-data>
<application-data identifier="certTool">CertTool</application-data>
<application-data identifier="bizTool">BizTool</application-data>
<application-data identifier="dateTool">DateTool</application-data>
<application-data identifier="riskTool">RiskTool</application-data>
<application-data identifier="tradeTool">TradeTool</application-data>
<application-data identifier="profileTool">ProfileTool</application-data>
<application-data identifier="dataLoadTool">DataLoadTool</application-data>
<application-data identifier="eventFinder">EventFinder</application-data>
<application-data identifier="auditTool">AuditTool</application-data>
<application-data identifier="commonTool">CommonToolBean</application-data>
<application-data identifier="riskQueryTool">RiskQueryToolBean</application-data>
```

规则

```
<rule name="D1" salience="30">
    <parameter identifier="event"><class>Event</class></parameter>
      <java:condition>
        bizTool.deposit(event)
      </java:condition>
      <java:condition>
        StringUtil.equals(EventUtil.getString(event, EventPropertyEnum.BANK_TYPE), "VISA")
      </java:condition>
      <java:condition>
        bizTool.payToDeposit(event)
      </java:condition>
      <java:consequence>
        CTUContextHolder.getContext().addProperty("TRANS_LOG_ID", event.getEventMainTarget());
      CTUContextHolder.getContext().addProperty("REFUND_MONEY", bizTool.getMoney(event));
        riskTool.fireScene(event, "0");
      </java:consequence>
  </rule>
</rule-set>
```

规则

- 规则
  - 有一组断言组成
  - 断言之间可以是与的关系
  - 一个断言可以由多个或的断言组成

规则监控

- 定位
  - CTU规则健康检查
  - CTU规则的建议者
  - 特殊情况的报警（附加）

规则监控

- 运行逻辑