YTS拆分数据扫描定时任务到独立微服务设计文档

**文档编号：yts-task-schedule**

**创建日期：2023-10-11**

**最后修改日期：2023-10-12**

**版 本 号：V1.0**

**电子版文件名：**YTS拆分数据扫描定时任务到独立微服务设计文档

文档修订摘要

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 日期 | 修订号 | 描述 | 著者 | 审阅者 |
| 2023-10-12 | 序列号1 | 初稿 | 逯永收 |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

目录

目录

[YTS拆分数据扫描定时任务到独立微服务设计文档 1](#_Toc828699910)

[电子版文件名：YTS拆分数据扫描定时任务到独立微服务设计文档 1](#_Toc1376158099)

[1. 概述 4](#_Toc83311282)

[现有方案的问题 4](#_Toc1430834480)

[1.1 目的 4](#_Toc1738859716)

[1.2阅读对象 4](#_Toc668391127)

[1.3 相关资料 4](#_Toc754310856)

[2. 整体设计 5](#_Toc1900390493)

[2.1整体设计 5](#_Toc241831369)

[2.2优化YTS-SDK中spring定时任务线程池数 6](#_Toc446833321)

[3.集中扫描服务详细设计 7](#_Toc424834600)

[3.1 微服务YTS定时任务是否已集中统一服务中接口 7](#_Toc549780354)

[3.2 上报微服务数据保留天数 8](#_Toc464370919)

[3.3 服务端记录YTS已统一定时扫描的微服务列表记录 8](#_Toc810032502)

[3.4 YTS-Schedule定时扫描的逻辑 9](#_Toc425989980)

[3.5 YTS-SDK增加异常事务重试的通用RPC服务 9](#_Toc1978938764)

[2.3.6 YTS-SDK 增加服务启动后查询是否已经统一定时任务判断 9](#_Toc1445661097)

[2.3.7 YTS-scheducle服务查询未进行逻辑数据源共享的服务提醒共享 10](#_Toc1868374792)

[2.4 方案4保留现有定时任务方案优化合包部署的定时任务 10](#_Toc633633662)

# 概述

### 现有方案的问题

1. 解决YTS-SDK启动11个定时任务
2. 定时任务绝大多数是数据库扫描和更新操作以及上报到服务端以及从服务端拉取数据
3. 按服务器实例启动了spring-schedule的线程池(5个线程)简强架构时需要合并线程池

## 1.1 目的

本文将阐述YTS的定时任务由原来仅支持Spring的定时任务框架触发拆分为数据库扫描上报部分迁移到EOS定时扫描的服务中，减少YTS的定时任务分布在每个微服务中占用线程池以及不能集中管理的问题，以及方便以后统一优化扫描逻辑。

## 1.2阅读对象

此文档将适合如下人员阅读：

领域设计工程师。

产品开发工程师。

产品测试工程师。

其他相关人员。

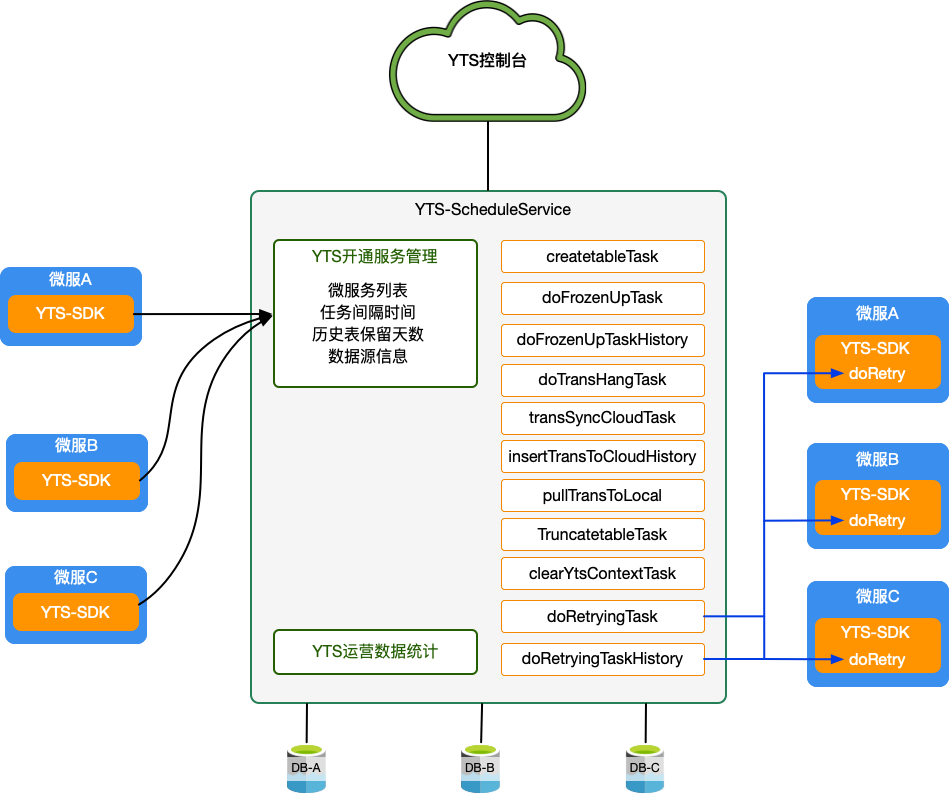
## 1.3 相关资料

《一致性框架(YTS)设计文档》

# 整体设计

为了版本兼容以及防止集中扫描服务异常导致YTS无法正常工作，保留原来的YTS中集成的spring定时任务，在服务启动时请求集中扫描服务的接口，查询该服务是否进行集中扫描，如果没有启用则仍旧启动原有的spring的定时任务使YTS功能正常。

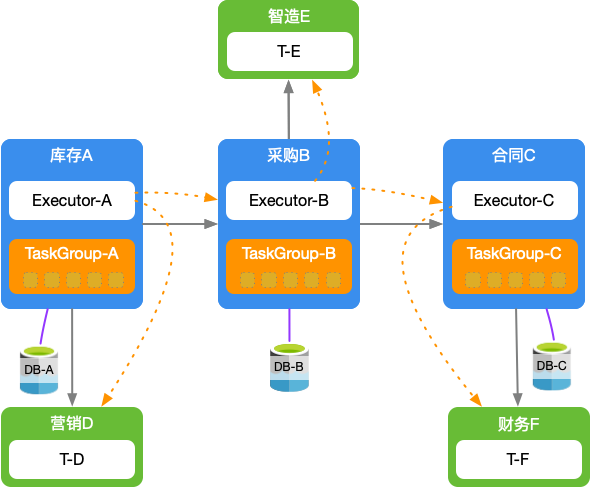
## 2.1整体设计



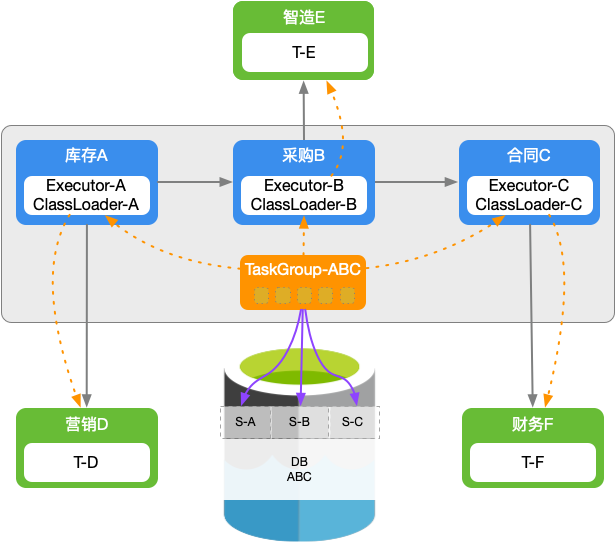
## 2.2优化YTS-SDK中spring定时任务线程池数

但在合包部署时需要优化YTS=SDK中spring定时任务线程数，使在合包部署时一个进程只开启一个spring的定时任务线程池。

优化前:



优化后：



# 集中扫描服务详细设计

## 3.1 微服务YTS定时任务是否已集中统一服务中接口

参数列表

|  |  |  |  |
| --- | --- | --- | --- |
| 参数编码 | 参数名说明 | 参数数据类型 | 是否必填 |
| serviceName | 微服务编码 | string | 是 |
| env | 部署的环境信息 | string | 是 |
| logicDsCode | 微服务逻辑数据源编码 | srring | 是 |
| ytsVersion | yts的版本信息 | string | 是 |

返回结果

|  |  |  |  |
| --- | --- | --- | --- |
| 参数编码 | 参数名说明 | 参数数据类型 | 是否必填 |
| isYtsSchedule | 是否已被统一调度 | boolean | 是 |

如果根据查询的微服务编码已经在统一的服务中进行统一定时服务中，并且已把服务的逻辑数据源共享则返回true否则返回false

## 3.2 上报微服务数据保留天数

参数列表

|  |  |  |  |
| --- | --- | --- | --- |
| 参数编码 | 参数名说明 | 参数数据类型 | 是否必填 |
| serviceName | 微服务编码 | string | 是 |
| env | 部署的环境信息 | string | 是 |
| ytsVersion | yts的版本信息 | string | 是 |
| truncateBeforeDays | 数据保留天数 | int | 是 |

返回结果

|  |  |  |  |
| --- | --- | --- | --- |
| 参数编码 | 参数名说明 | 参数数据类型 | 是否必填 |
| reportSuccess | 是否上报成功 | boolean | 是 |

服务端查询该服务的记录，确认如果数据的保留天数和上报的不同则修改服务的保留天数的记录，修改成功或者记录的保留天数和上报的一致则返回成功，否则返回false。

## 3.3 服务端记录YTS已统一定时扫描的微服务列表记录

记录YTS已统一定时扫描的服务列表

yts\_schedule\_service\_info

|  |  |  |
| --- | --- | --- |
| 属性 | 类型 | 说明 |
| id | bigint | oid 主键 |
| service\_name | verchar(255) | 微服务编码 |
| logic\_ds\_code | varchar(128) | 微服务逻辑数据源编码 |
| ds\_share\_status | int | 是否已共享数据源  如果没有共享数据源不开启集中扫描，启动原有的定时任务 |
| truncate\_before\_days | int | YTS记录保留的天数 |
| sdk\_version | varchar(50) | YTS的版本 |
| env | varchar(50) | 部署的环境信息 |
| report\_time | datetime | 第一次上报时间 |
| update\_time | datetime | 最后一次上报时间 |

表结构设计要遵循数据库设计规范

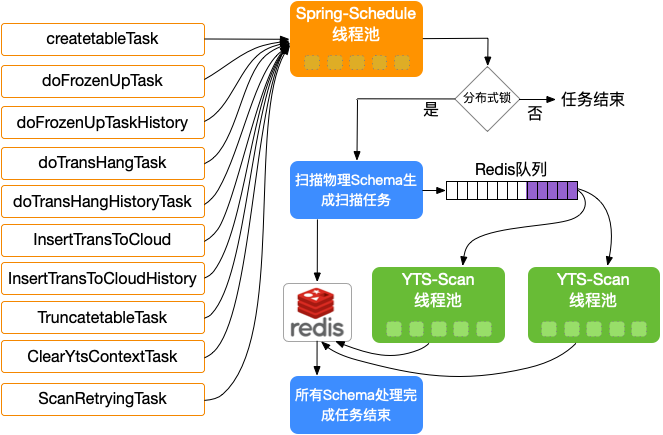
服务合并的场景（对比领域服务合并文档和yts接入文档需要处理YTS相关事项）

合并场景包括rpc短路和数据库事务合并场景的区分。

## 3.4 YTS-Schedule定时扫描的逻辑

根据YTS定时任务设置的间隔，定时运行YTS的定时任务逻辑。

定时任务的逻辑根据微服务名遍历逻辑数据源中的所有schema扫描数据，并完与YMS-console云服务的交互，如果有需要重试的定时任务则调用该微服务的重试RPC接口完成该异常事务的重试操作。每个定时任务的主要功能参考 （3.8章节）



\* 建表的定时任务如果YTS-Scan线程池的任务队列满则等待，其他任务直接忽略，等待下一轮扫描。

## 3.5 YTS-SDK增加异常事务重试的通用RPC服务

*/\*\*  
 \* 重试异常事务  
 \* @param gtxId  
 \* @return  
 \*/*ProcessResult retry(String gtxId);

该服务有YTS-schedule的统一定时任务根据微服务名进行调用

## 3.6 YTS-SDK 增加服务启动后查询是否已经统一定时任务判断

服务启动后YTS-SDK增加从YTS-schdule服务中查询是否已经统一定时任务如果已经统一定时任务则无需启动内部的定时任务，交由统一的定时任务服务扫描数据库并YMS-console进行交互，本服务有需要重试的异常事务时有统一定时任务调用该服务的重试接口进行重试。

启动后上报本服务的相关信息到YTS-schedule服务中。

## 3.7 YTS-scheducle服务查询未进行逻辑数据源共享的服务提醒

YTS-schedule微服务查询已经上报信息而且没有进行逻辑数据源共享的服务，提醒该微服务进行数据源共享。

## 3.8 YTS定时任务

定时任务扫描所有开启集中扫描的微服务，为每个微服务额所有物理数据源进行定时任务的操作。

### 3.8.1 定时建表的定时任务(CreateTableTask)

默认执行时间为凌晨1:05/2:05/3:05/4:05/5:05/6:05执行6次避免由于某个时间段数据库被锁等原因无法建表成功的问题。运行建表成功后，将标记进行缓存，下次任务执行时直接跳过，无需进行多次数据库建表操作，建表类型有2个一个为yts\_trans\_yyyyMMdd一个为yts\_rel\_yyyyMMdd，每个类型表均创建未来10天的表，避免服务某些原因异常导致定时任务没有正常执行而无法建表成功的问题。

### 3.8.2删除过期历史表的定时任务(TruncateTableTask)

根据微服务设置的保留历史天数的设置，删除保留天数以前的yts\_trans\_yyyyMMdd,yts\_rel\_yyyyMMdd两种类型的表。

### 3.8.3 删除过期YTS上下文数据的定时任务(CleanYtsContextTask)

根据微服务设置的保留历史数据天数清理yts\_context表中过期的YTS上下文的数据，清理数据时根据gtxId中的日期清理过期数据，清理需要分批并且在执行过程可以根据设置间隔休眠一定时间，避免对数据库形成过大的压力。

### 3.8.4 防挂起(当前日期)处理的定时任务（DoTransHangTask）

默认间隔为10分钟一次，查询当天yts\_trans\_yyyyMMdd表中状态为confirming的Root节点的数据，根据调用下游的rel记录完成该事务的后续逻辑，使该事务处理能完成处理。

### 3.8.5 防挂起(历史)处理的定时任务(DoTransHangHistoryTask)

默认间隔时间为1小时一次，查询该服务保留天数并遍历查询yts\_trans\_yyyyMMdd表中状态为confirming的Root节点的数据，根据调用下游的rel记录完成该事务的后续逻辑，使该事务处理能完成处理。

### 3.8.6 重试的定时任务(DoRetryTransTask)

默认间隔时间为10分钟，服务从YTS云端控制台查询该服务需要重试的异常事务额列表，将异常事务更新状态为待重试，查询数据库中状态为retrying的事务然后遍历调用该服务的重试接口（3.5章节）进行重试。

### 3.8.7 解锁业务数据(当天)的定时任务(DoFrozenUpTask)

默认执行时间间隔为10分钟，查询当天有待解锁的业务数据记录，调用解锁接口进行解锁操作。

### 3.8.7 解锁业务数据(历史)的定时任务(DoFrozenUpHistoryTask)

默认执行时间间隔为10分钟，根据微服务设置的保留历史天数，遍历查询所有历史表中待解锁的历史记录调用解锁接口进行解锁操作。

### 3.8.8 上报当天异常事务的定时任务(ReportToCloud)

默认执行时间为10分钟，上报当天异常事务数据或者当天重试后状态有变更的数据，以及该服务业务成功但上游业务失败的事务数据到云端控制台，方便云端进行管理，以及重试等操作。

### 3.8.9 上报历史异常事务的定时任务(ReportToCloudHistory)

默认间隔时间为1小时，根据微服务设置的历史数据保留天数查询所有历史表中需要上报的异常事务数据上报到云端。

# 服务部署

由于调用YTS的重试任务需要使用RPC的服务调用，所以部署未和微服务对应的环境每个环境部署一份，先期该集中扫描的服务功能集成到EOS的统一服务中。

# 5.附录

## 5.1 已接入YTS的微服务列表：

微服务：86

Shema：76

|  |  |  |  |
| --- | --- | --- | --- |
| 领域 | 微服务名 | schema | 专属化schema |
| 平台 | iuap-apdoc-customer | ugoods | iuap\_apdoc\_coredoc |
| iuap-apdoc-material | iuap\_apdoc\_coredoc |
| ubpm-web-rest | ubpm |  |
| 基础数据 | yonbip-am-ambd | amc\_ambd | amc\_ambd |
| yonbip-am-aum | amc\_aum | amc\_aum |
| yonbip-am-rmm | amc\_mro | amc\_mro |
| 采购云 | yonbip-cpu-bcprice | cpu\_baseservice | cpu\_baseservice |
| yonbip-cpu-bcres | cpu\_baseservice |
| yonbip-cpu-bcsourcing | cpu\_pubbiz\_sourcing | cpu\_pubbiz\_sourcing |
| yonbip-cpu-bid | cpu\_bid | cpu\_bid |
| yonbip-cpu-cooperation | cpu\_cooperation | cpu\_cooperation |
| yonbip-cpu-lawbid | cpu\_lawbid | cpu\_lawbid |
| yonbip-cpu-mall | cpu\_mall | cpu\_mall |
| yonbip-cpu-sourcing | cpu\_sourcing | cpu\_sourcing |
| yonbip-cpu-suppliermgr | cpu\_suppliermgr | cpu\_suppliermgr |
| yonbip-cpu-synergy | cpu\_synergy | cpu\_synergy |
| yonbip-cpu-yop | yuncai\_yop | yuncai\_yop |
| 财务 | yonbip-fi-arap | fiproduct | fiproduct |
| yonbip-fi-ctmbam | ctmbam | ctmbam |
| yonbip-fi-ctmcam | ctmcam | ctmcam |
| yonbip-fi-ctmcmp | ctmcmp | ctmcmp |
| yonbip-fi-ctmcmp-import | ctmcmp |  |
| yonbip-fi-ctmcspl | ctmcspl | ctmcspl |
| yonbip-fi-ctmdrft | ctmdrft | ctmdrft |
| yonbip-fi-ctmfdtr | ctmfdtr | ctmfdtr |
| yonbip-fi-ctmgrm | ctmgrm | ctmgrm |
| yonbip-fi-ctmlcm | ctmlcm | ctmlcm |
| yonbip-fi-ctmlgm | ctmlgm | ctmlgm |
| yonbip-fi-ctmstct | ctmstct | ctmstct |
| yonbip-fi-ctmstwb | ctmstwb | ctmstwb |
| yonbip-fi-ctmtlm | ctmtlm | ctmtlm |
| yonbip-fi-eafrecord | yms\_pre | idoc |
| yonbip-fi-eafsys | yms\_pre | idoc |
| yonbip-fi-etl | figl | figl |
| yonbip-fi-expsapct | apct | apct |
| yonbip-fi-expsrbsm | znbz | znbz |
| yonbip-fi-fa | fiefa | fiproduct |
| yonbip-fi-gl | figl | figl |
| yonbip-fi-ia | fieia | fi\_report |
| yonbip-fi-it | fiproduct | fiproduct |
| yonbip-fi-miderp | fiproduct | fiproduct |
| yonbip-fi-otp | figl | figl |
| yonbip-fi-sepmbcdc | epmbcdc | epmp\_bcdc |
| yonbip-fi-taxit | diwork\_input\_tax | diwork\_input\_tax |
| yonbip-fi-taxot | diwork\_piaoeda\_client | diwork\_piaoeda\_client |
| 人力 | yonbip-hr-contract | corehr | corehr |
| yonbip-hr-paybiz | diwork\_wa\_mdd | diwork\_wa\_mdd |
| yonbip-hr-paybizo | diwork\_wa\_split | diwork\_wa |
| yonbip-hr-staff | iuap\_cloud\_basedoc | iuap\_apdoc\_basedoc |
| yonbip-hr-staffmgo | iuap\_cloud\_basedoc | iuap\_apdoc\_basedoc |
| yonbip-hr-staffmgr | corehr | corehr |
| yonbip-hr-staffproc | corehr |
| 共享服务 | yonbip-mid-sscpf | shared\_service | shared\_service |
| 营销 | yonbip-mkt-crm | crm | crm |
| yonbip-mkt-dst | omsdata | omsdata |
| yonbip-mkt-dstbiz | omsdata |
| yonbip-mkt-mkc2b | uorders | uorders |
| yonbip-mkt-retailbiz | uretaildata | uretaildata |
| yonbip-mkt-retailweb | uretaildata |
| yonbip-mkt-usmp | usmp | usmp |
| 制造 | yonbip-mm-mfed | engineering\_data | engineering\_data |
| yonbip-mm-mfmr | requirements\_planning | requirements\_planning |
| yonbip-mm-mfpo | production\_order | production\_order |
| yonbip-mm-qic | qms\_dfm | qms\_dfm |
| yonbip-mm-qms | qms\_qit | qms\_qit |
| yonbip-mm-sfc | imp\_sfc | imp\_sfc |
| 项目云 | yonbip-pm-budgetme | pmcloud | pmcloud |
| yonbip-pm-commonbd | pmcloud |
| yonbip-pm-planme | pmcloud |
| yonbip-pm-project | pmcloud |
| yonbip-pm-projectme | pmcloud |
| yonbip-pm-revexp | pmcloud |
| yonbip-pm-workhours |  | pmcloud |
| 供应链 | yonbip-scm-inttrade | uinttrade | uinttrade |
| yonbip-scm-inttradebiz | uinttrade |
| yonbip-scm-pu | upurchase | upurchase |
| yonbip-scm-pubiz | upurchase |
| yonbip-scm-sact | sact | sact |
| yonbip-scm-scmbd | scmbd | scmbd |
| yonbip-scm-scmmp | scmmp | scmmp |
| yonbip-scm-scmpub | ustock | ustock |
| yonbip-scm-scmpubbiz | uscmpub | ustock |
| yonbip-scm-scmsa | uorders | uorders |
| yonbip-scm-stock | ustock | ustock |
| yonbip-scm-stockbiz | ustock |
| yonbip-scm-stockimport | ustock |