金财工程应用支撑平台

工作流组件服务化建设方案

V2.1



北京兴财信息技术有限责任公司

版权所有不得翻录

文 件 更 改 记 录

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 版本号 | 状态 | 修改内容概要/  引见说明 | 修订日期 | 修订人 | 批准人 |
| V1.0 | N | 建立 | 2018-07-18 | 孟凡龙 |  |
| V1.0 | M | 修改 | 2018-07-18 | 李文 |  |
| V1.0 | M | 修改 | 2018-07-18 | 董欢 |  |
| V2.0 | M | 修改 | 2018-10-12 | 孟凡龙 |  |
| V2.0 | M | 修改 | 2018-10-12 | 李文 |  |
| V2.1 | M | 修改 | 2018-10-15 | 郑文浩 |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

状态包括：N——新建，A——增加，M——修改，D——删除。

目录

[1工作流组件升级背景 2](#_Toc527724881)

[1.1工作流组件现状 2](#_Toc527724882)

[1.2工作流组件升级目标 3](#_Toc527724883)

[2工作流组件服务化总体思路 3](#_Toc527724884)

[2.1工作流架构方式升级 4](#_Toc527724885)

[2.2服务化工作流部署结构 5](#_Toc527724886)

[2.3服务化后对当前业务系统的影响评估 7](#_Toc527724887)

[2.4基于服务化工作流开发的标准设计 7](#_Toc527724888)

[3工作流组件服务化技术实现 8](#_Toc527724889)

[3.1工作流组件服务化系统架构 8](#_Toc527724890)

[3.2平台2.0版本选型 9](#_Toc527724891)

[3.3创建工作流数据库 9](#_Toc527724892)

[3.4客户端内存数据库H2 10](#_Toc527724893)

[3.5 UI管理界面 10](#_Toc527724894)

[4工作流组件服务化建设方案 10](#_Toc527724895)

[4.1开发服务化工作流系统 10](#_Toc527724896)

[4.1.1统一的流程定义模板配置和同步 11](#_Toc527724897)

[4.1.2流程监控及统计分析 11](#_Toc527724898)

[4.2开发工作流组件客户端 13](#_Toc527724899)

[4.2.1流程模板自动部署 13](#_Toc527724900)

[4.2.2定义模板更新数据同步 14](#_Toc527724901)

[4.2.3定时推送日志信息 14](#_Toc527724902)

[5业务系统如何接入工作流服务 14](#_Toc527724903)

[5.1新系统接入步骤 14](#_Toc527724904)

[5.2存量业务系统接入 15](#_Toc527724905)

[5.3第三方系统接入 15](#_Toc527724906)

[6嵌入式工作流与服务化工作流对比 16](#_Toc527724907)

# 工作流组件升级背景

## 工作流组件现状

目前应用支撑平台2.0（以下简称“平台2.0”）工作流组件采用的是嵌入式架构。嵌入式工作流引擎不能以独立服务的形式存在，而是以一个软件组件（构件）的形式在各业务系统应用中运行，这些组件（构件）通过提供WAPI（Workflow API）为业务系统应用表现层或业务逻辑层的其他部分提供服务（如启动指定工作流程、查询工作任务、设置流程运行业务数据等）。

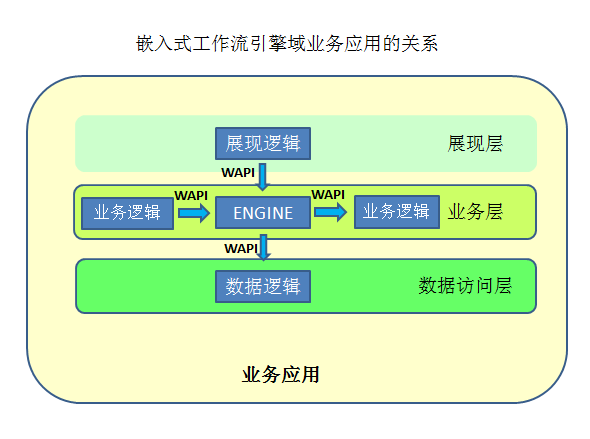


图1.1‑1嵌入是工作流引擎与业务应用的关系图

如上图所示，嵌入式工作流引擎ENGINE分散在不同的业务系统应用当中，业务系统应用通过调用通过本地工作流引擎提供的各类WAPI即可完成业务流程运转。因此，对于平台2.0提供的这种嵌入式工作流组件，存在以下问题：

1. 难以对工作流组件进行有效的管控。虽然工作流引擎由平台2.0组件提供，但由于工作流核心引擎是分散在各业务系统应用服务中，平台很难做到对工作流组件的权限控制，比如工作流组件应用的安全认证等。
2. 不能有效掌握工作流组件的使用情况。对各业务系统应用工作流组件应用情况无法掌握，如现有工作流组件提供组件功能和服务是否合理，使用频度情况如何等等，不利于工作流组件的迭代更新和完善优化。
3. 不能对业务流程流转情况进行监控和综合分析。在平台2.0嵌入式工作流模式下，工作流流转数据也分布在各不同的业务系统当中，没有提供统一的工作流监管功能和流程分析功能。对业务系统通过工作流组件实现流程运转时的业务流程进度和节点状态等信息，不能进行有效的监控和分析。

## 工作流组件升级目标

重新基于平台2.0构建服务化的工作流组件产品，其主要目标如下：

1. 重构的工作流组件能够基于平台2.0域服务独立运行，对各表层业务系统提供多样的、稳定的、能够满足业务需要的流程服务。
2. 实现平台2.0对工作流组件的使用进行集中管控，对工作流组件的使用提供安全认证，对工作流服务使用情况进行监控。
3. 实现平台2.0对业务流程的流程流转情况（如流程运行状况、节点状态等情况）进行集中监控和数据分析，直观有效的发现各个业务逻辑中无用流程、重复流程等，并为表层业务流程优化提供依据，高效发挥平台的集中管控作用。

# 工作流组件服务化总体思路

工作流组件服务化基于平台2.0重新进行构建，基于平台2.0重新构建后，表层业务系统可逐步将流程服务调用迁移至服务化后的工作流引擎上。其主要思路有：

## 工作流架构方式升级

对工作流的架构方式进行升级，由当前的嵌入式架构升级为能够基于平台域服务独立运行的组件服务化架构模式。

如下图所示，当前平台2.0嵌入式工作流引擎在应用时，需要与业务系统进行深度集成。由平台2.0工作流组件提供jar包构件及对应的流程设计UI页面，业务系统在进行集成后，通过调用在本地运行的工作流引擎WAPI，完成业务流程的定义和流转。因此，此时，无论是流程模板的定义还是流程运行产生的数据均在业务系统应用中流转，而平台2.0与嵌入在业务系统应用中的工作流难以建立有效的联系。

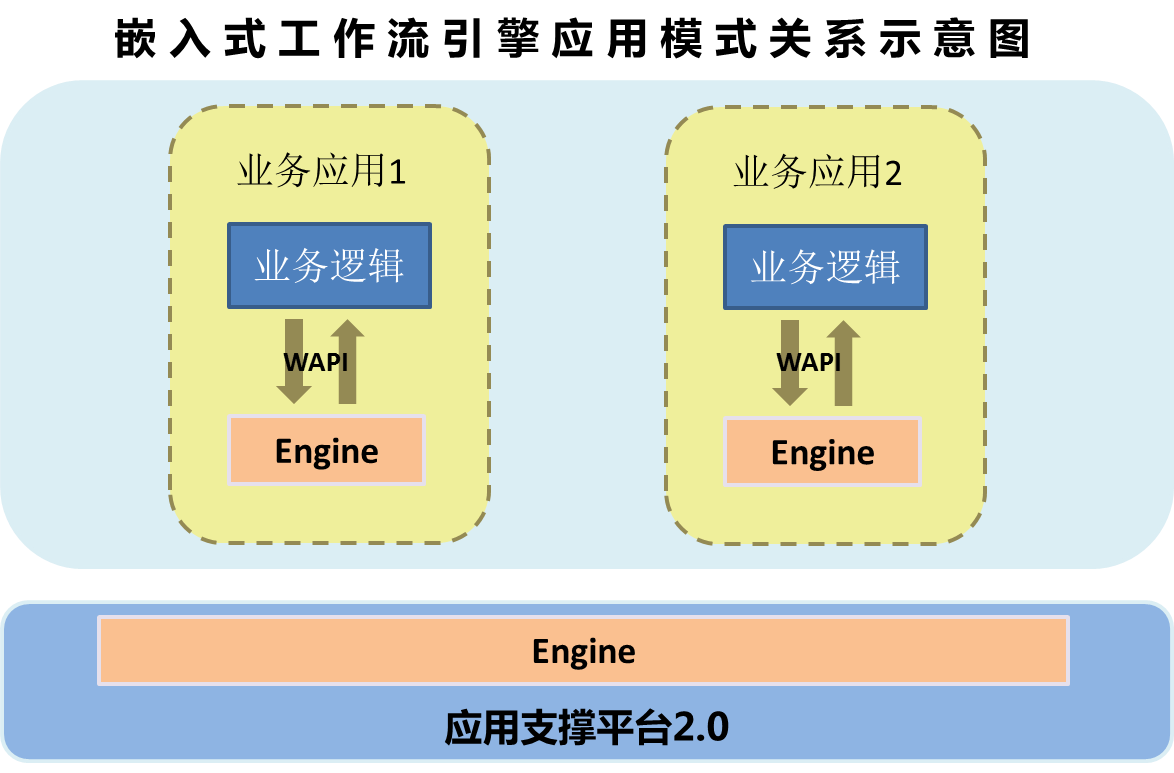


图2.1‑1嵌入式工作流引擎应用模式关系示意图

基于平台2.0重新构建服务化模式的工作流组件（后续简称“服务化工作流”），利用平台2.0的“纵向到底，横向到边”的连通能力和优势，将工作流引擎从业务系统中独立出来，基于平台2.0从域独立部署运行，对外提供统一、标准的WAPI，包括Web Service、restful等，业务系统进行远程调用，实现自身业务流程的流转。服务化运行的工作流引擎与业务系统及平台的关系如下图所示：

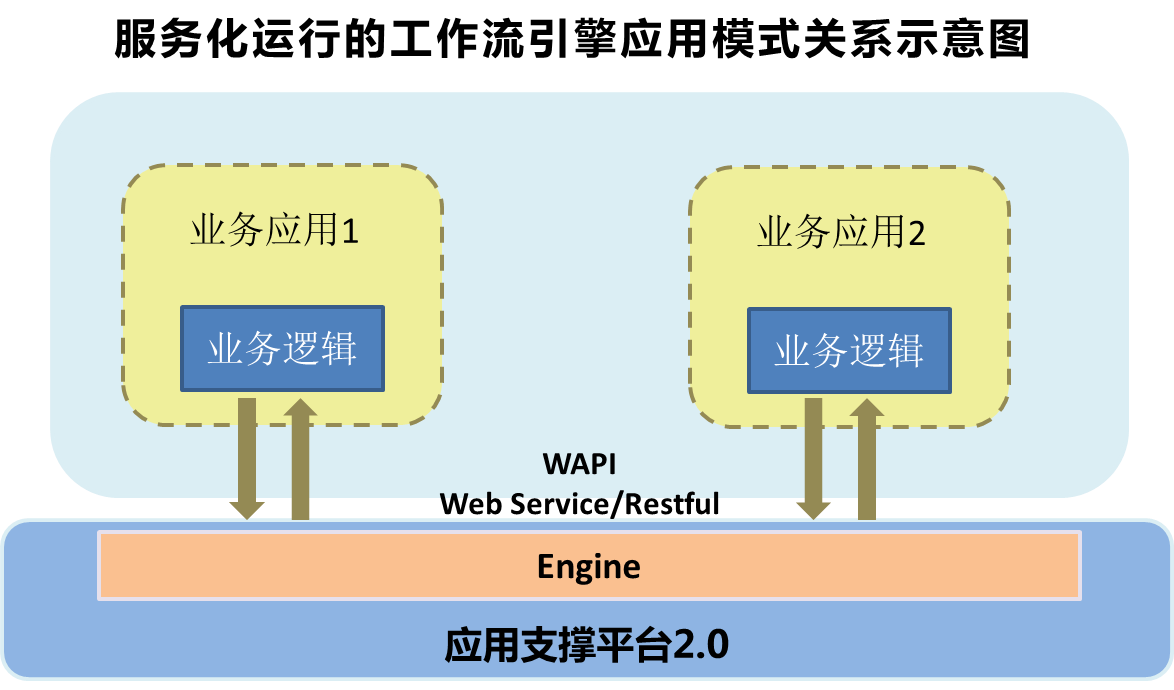


图2.1‑2服务化运行的工作流引擎应用模式关系示意图

## 服务化工作流部署结构

重构后的工作流组件采用服务化方式，建立统一、独立运行的工作流引擎，充分利用平台2.0认证管理功能，对工作流引擎的使用进行管控；统一工作流模板创建和管理，包括工作流模板的定义、横向模板信息同步、纵向模板信息下发；对各业务应用使用工作流引擎的情况接流程流转情况进行监控和分析，从宏观上了解各业务的业务流转情况，为财政业务的业务流程优化提供数据依据和参考。其部署结构图如下图所示：

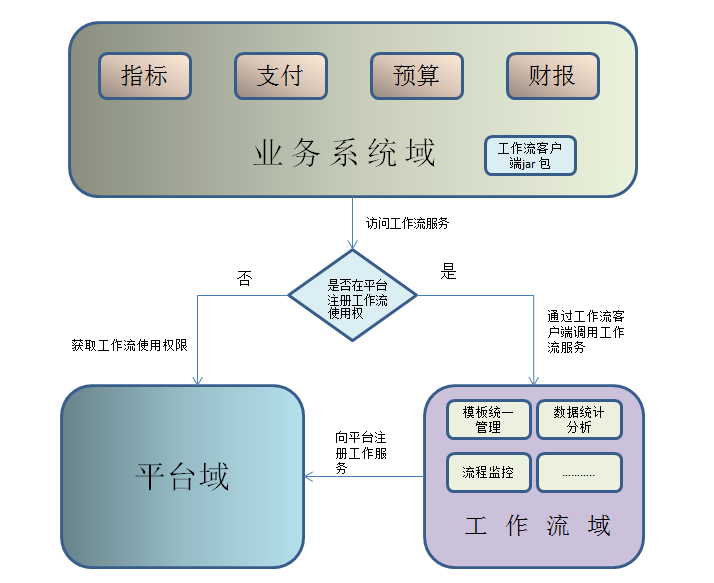


图2.2‑1工作流组件服务化部署结构示意图

如上图所示，工作流组件服务化后，工作流组件作为平台一个从域进行独立部署，同时，对业务系统提供客户端jar包和WAPI服务。其应用流程和访问描述如下：

1. 业务系统应用访问工作流组件，需要先进行平台2.0的认证，认证通过后，方可访问。
2. 业务系统在平台2.0工作流UI界面中统一维护相关流程模板，并完成流程模板的同步。
3. 业务系统端工作流客户端jar包远程调用工作流域提供的WAPI，在进行业务流程流转的同时，返回节点状态及流程流转日志等工作流数据信息。
4. 工作流组件完成对各业务系统的业务流程监控与日志分析。

## 服务化后对当前业务系统的影响评估

目前，基于平台2.0生长开发的各业务系统，包括预算、指标、支付、财报以及OA系统，均使用的是当前嵌入式工作流组件。在对工作流组件服务化方案编制、选型、确定时，综合考虑上述业务系统情况，以不影响当前业务系统为原则，以能够平滑升级到服务化工作流为过程，最终顺利实现现有业务系统以及新增业务系统均能够使用组件服务化工作流引擎的目标。

在工作流组件服务化后，当前使用嵌入式工作流组件向服务化工作流迁移工作主要包括以下两个方面：

* 归集当前所有业务系统的嵌入式工作流流程模板信息，并在服务化工作流引擎中进行统一初始化和管理。
* 业务系统升级工作流服务客户端JAR包功能。

通过上述步骤即可完成现有业务系统嵌入式工作流引擎到服务化工作流引擎的平滑升级。

## 基于服务化工作流开发的标准设计

对今后规划或者正在开发过程中的业务系统，应按照服务化后的工作流开发标准和规范使用工作流组件。

1. 严格遵照《应用支撑平台开发技术规范》，在平台注册业务系统应用相关信息，在使用工作流组件前，将经过平台认证模块进行合法认证。
2. 平台2.0统一管理所有业务系统工作流流程模板信息，流程模板信息维护管理完成后，将自动推送至各业务系统端。
3. 业务系统通过工作流组件提供的统一、标准的WAPI，完成最终的工作流相关流程服务。

# 工作流组件服务化技术实现

## 工作流组件服务化系统架构

工作流组件服务化后的总体系统架构如下图所示：

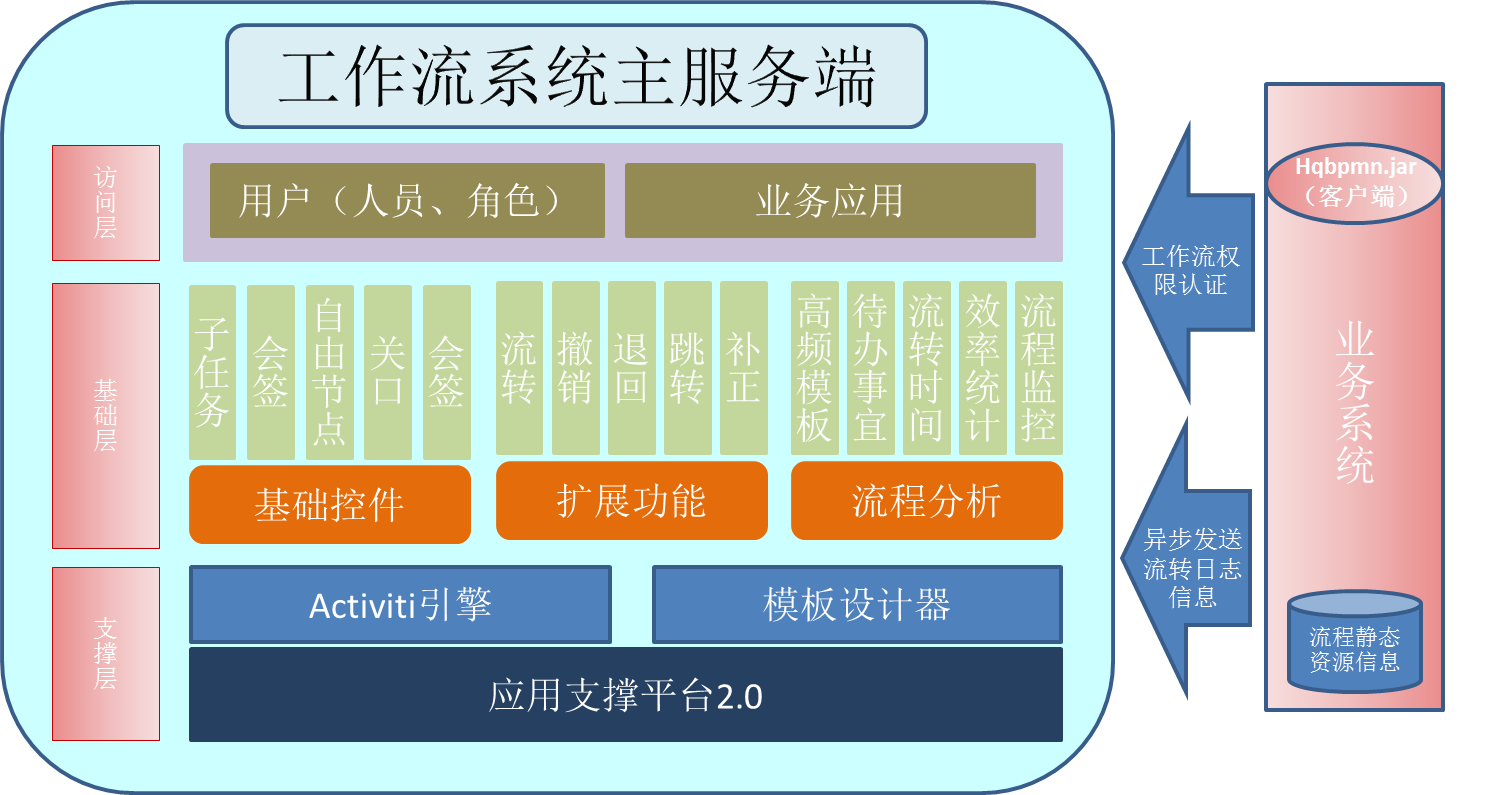


图3.1‑1工作流组件服务化系统架构图

工作流组件服务化是基于平台2.0开发工作流系统，包括基于平台2.0从域独立运行的工作流系统主服务端以及提交业务系统集成的客户端JAR包两部分。

1. **在工作流系统主服务端，分为三层结构：支撑层、基础层和访问层。**

* 支撑层主要包括应用支撑平台2.0（工作流系统权限、系统认证管理等）、Activiti引擎、模板设计器以及操作系统、数据库、负载等软硬件环境。
* 基础层主要包括基础控件、扩展功能、流程分析等工作流组件提供的各类功能。
* 访问层主要是对工作流系统的用户，包括终端用户及业务系统，主要提供功能UI访问页面以及WebService、Restful等方式的WAPI服务。

1. **在工作流系统客户端，作为一个JAR包，与业务系统应用包进行集成，完成与工作流系统主服务端的通信。包括：**

* 通过对主服务端WAPI接口服务的调用，实现业务系统工作流流程流转。
* 将业务系统端业务流程流转日志信息，通过异步方式推送到工作流系统主服务端，用于流程监控和监控数据的统计与分析。

## 平台2.0版本选型

平台2.0从发布至今，进行过多版本迭代和优化完善。本次工作流组件服务化建设将以平台2.4.X.X为基础，支持平台2.4开头及以上版本。

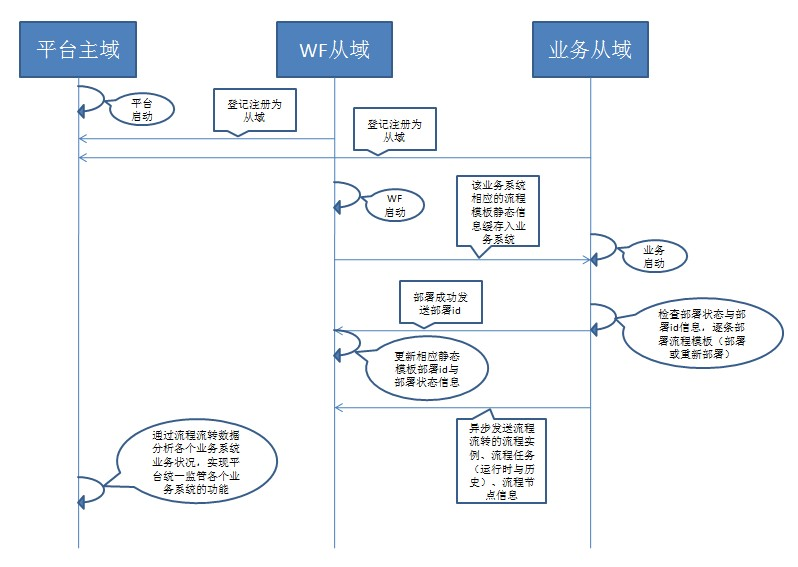


图3.2‑1工作流组件服务化时序流程图

基于平台2.0主域服务，建立工作流从域。通过dubbo方式完成主从域服务和数据的访问和交互。

工作流UI页面与平台2.0管理首页集成，作为平台2.0工作流组件模块功能。由平台2.0统一用户入口。

通过平台2.0控制业务系统工作流系统的使用权限，监控并展示业务系统工作流流程和节点状态。

## 创建工作流数据库

工作流组件服务化后，其基于平台工作流域独立运行时，建立工作流数据库，包括一些工作流基础表，包括：

* 流程定义数据表——业务流程定义数据表，存储每个流程定义文件记录；
* 属性数据表 ——存储整个流程引擎级别的数据；
* 二进制数据表——保存流程定义图片和xml、Serializable(序列化)的变量,即保存所有二进制数据。

在此基础上，扩展工作流组件服务化后对应的一些业务表，比如：

* 日志表——存储各个业务系统流转的工作流日志信息；
* 流程定义模板与业务系统中间表——控制业务系统对流程定义模板的使用权限以及模板的部署状态。

同时，根据用户的需求以及开发的需要还需要对一些表进行分区处理。

## 客户端内存数据库H2

引入开源内存数据库H2，利用其免部署，占用空间小，不受平台限制等特点，作为工作流客户端的模板数据库使用：

1. 业务系统通过调用工作流系统的服务获取静态资源信息（如流程定义模板），并将其保存到H2内存数据库中。
2. 业务系统从H2中获取流程定义模板，并将模板部署到业务系统的数据库中。
3. 工作流客户端依据H2静态模板部署的工作流正常流转工作流程。

## UI管理界面

工作流组件服务化后，将提供统一的流程定义管理、监控配置和分析页面，其功能页面将与平台2.0集成，作为平台2.0工作流组件模块功能。

# 工作流组件服务化建设方案

## 开发服务化工作流系统

工作流组件服务化的主要内容是基于平台2.0重构一个集中部署和管理的工作流系统主服务端，可以基于平台2.0从域独立部署和运行，集中管理一级财政所有业务系统的业务流程。其主要功能如下：

### 统一的流程定义模板配置和同步

工作流组件服务化后，由工作流系统主服务端提供统一的流程定义模板配置功能。工作流系统主服务端对流程定义模板进行权限控制，不同业务系统流程模板定义完成后，分发至对应权限的业务中。

### 流程监控及统计分析

工作流系统主服务端建设流程监控及统计分析功能。对业务流程执行过程和状态进行实时监控，多维度获取流程执行过程和状态，管理者可以完整、全面、系统的分析流程执行情况。可以按照组织结构、人员、流程步骤、流程处理状态（同意、退回、撤销）等方面考量分析。

##### 高频流程模板统计

针对所有的流程模板进行分析，精确定位使用率高频与低频的流程模板，为业务系统的业务逻辑优化提供数据依据，如整改低频模板逻辑，优化高频模板效率等。

##### 待办事宜统计

针对待办事宜多少来进行邮件或者短信通知相应的处理用户进行业务处理，实现整个项目的运转不会因为某个节点的停顿而导致项目超时。

##### 流程流转时间分析

针对使用流程模板流转的项目的流转时间进行分析，从而确定哪些项目需要增加人员进行重点处理，哪些项目可以适当增加业务来提高业务量。

##### 流程效率统计

可以按日报，月报，季报，年报，自定义方式，对办件量，办件状态进行统计，通过统计分析，可以帮助管理人员分析工作流状态，定位工作瓶颈，改进工作流程，提高工作效率。

* 待办事宜最多人员统计
* 已办事宜最多人员统计
* 节点操作效率人员统计

##### 数据的图形化分析

对工作流系统实现流程实例运行状况进行分析并对结果进行图形化展示。

* 柱状图
* 饼状图

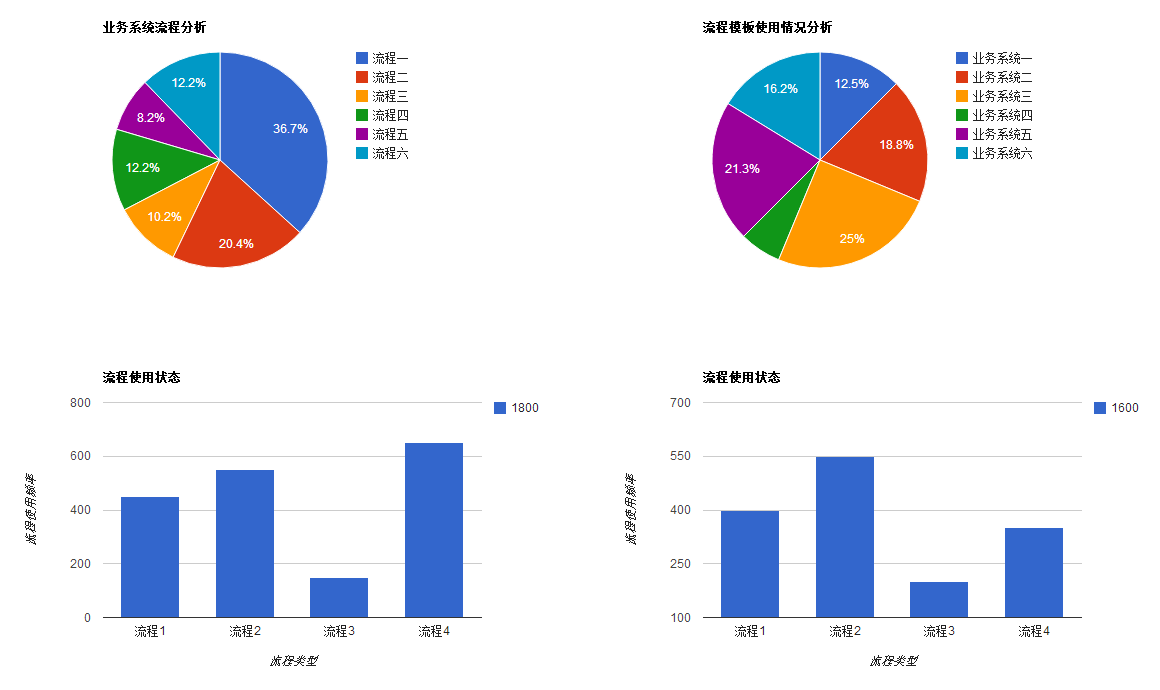


图4.1‑1流程监控原型示例1

* 待办已办任务线性分析

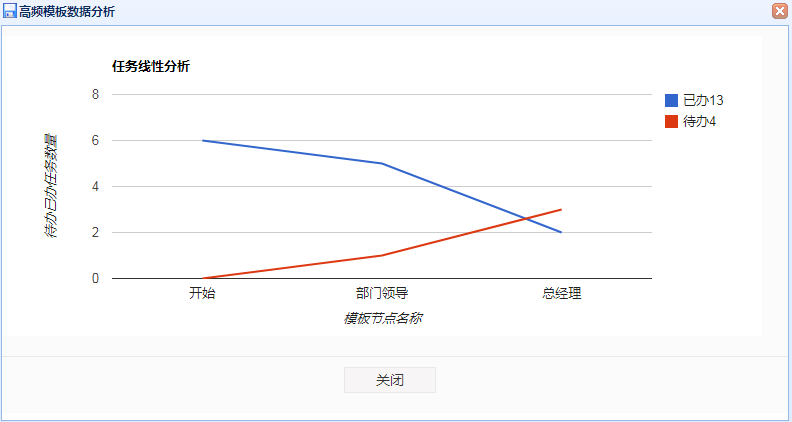


图4.1‑2流程监控原型示例2

## 开发工作流组件客户端

为分担业务系统系统业务流程运行对流程引擎运行的压力，同时兼容现有嵌入式工作流引擎。开发工作流组件客户端，与基于平台2.0从域部署的工作流组件对应，以JAR包形式提供给业务系统。

工作流组件客户端主要功能包括：

* 支撑本地业务系统业务流程运转；
* 接收工作流组件主服务端的流程定义模板，并在业务系统库中进行自动部署；
* 记录工作流流程日志并推送至工作流组件主服务端。

### 流程模板自动部署

开发流程模板自动部署功能，在平台2.0工作流组件中完成模板定义后，模板发送至工作流客户端，支持模板到业务系统的自动化部署。工作流客户端工作流流程模板自动部署业务描述如下：

1. 工作流组件流程定义模板添加业务系统标识。
2. 客户端获取工作流组件主服务端创建的流程定义模板。
3. 客户端将获取到的未部署的流程定义模板部署到本地，部署完成后将部署信息反馈到工作流组件主服务端。
4. 工作流组件主服务端对模板部署状态进行更新（如部署状态等）。

### 定义模板更新数据同步

工作流客户端支持即时接收工作流组件主服务端新增或修改的业务流程定义模板，并进行业务系统库的部署（新增时）或重新部署（修改时）。

### 定时推送日志信息

工作流组件客户端在业务系统库中创建工作流临时日志信息表，记录业务系统业务流程流转日志信息。

客户端通过定时任务将日志临时表中的流程流转日志推送至工作流组件主服务端，同时清空临时日志表数据。

# 业务系统如何接入工作流服务

考虑工作流组件服务化后，在业务系统应用过程中，将面临以下三类情况：

* 新开发业务系统：指新开发基于平台2.0的业务系统，使用新工作流服务。
* 存量业务系统：指存量的基于平台2.0工作流开发的业务系统。
* 第三方业务系统：第三方存量业务系统，有自己的工作流服务。

针对上述情况，可通过下述步骤完成新工作流服务端接入。

## 新系统接入步骤

基于平台2.0开发新的业务系统，按照服务化后的工作流组件开发规范，实现位于系统对服务化工作流的应用。

1. 集成工作流客户端JAR包
2. 业务系统通过平台2.0注册和认证
3. 按服务化工作流系统开发规范实现流程模板设计和流程服务调用。

## 存量业务系统接入

对于当前已存在的基于平台2.0开发的业务系统，使用当前的嵌入式工作流服务，需要升级为服务化的工作流组件服务。业务系统不需要进行大的变动，其使用工作流的方式与当前保持一致，从应用的角度，其主要变化是工作流流程定义模板设计由工作流系统进行统一的管理，不再是分散在各业务系统中维护。同时，新的工作流系统提供更多的监控类、数据分析类服务。

其升级工作主要包括：

1. 升级工作流客户端（替换业务系统中客户端JAR包）。
2. 升级工作流脚本。
3. 在平台2.0工作流系统中，导入各业务系统现有流程模板数据，并标记对应业务系统标识。
4. 升级完成后，按服务化工作流系统标准，设计流程模板与调用流程服务。

## 第三方系统接入

服务化后的工作流组件支持对第三方系统业务流程的关键节点流程监控。通过影子流程技术，在不改变现有业务系统工作流引擎的情况下，第三方业务系统调用平台工作流组件提供WAPI，将关键流程节点流程流转数据推送至平台2.0工作流组件，工作流组件对第三方系统推送到流程流转数据进行事后数据分析和展示。其主要接入步骤包括：

1. 影子流程模板设计

梳理需求后，按照业务系统工作流业务流程，在平台2.0工作流组件中设计对应的流程模板。

1. 业务系统获取工作流接入标准

第三方业务系统在调用平台2.0工作流组件提供的标准WAPI，获取已设计好的流程模板（标准）。第三方业务系统按照该标准对业务流程流转数据进行整合。

1. 业务系统推送分析数据

第三方业务系统调用平台2.0工作流系统提供的WAPI接口将整合好的流程流转数据推送到工作流系统，工作流系统对流程流转数据进行数据分析并进行展示。

# 嵌入式工作流与服务化工作流对比

嵌入式工作流与服务化工作流主要差异对比如下表所示：

表格1嵌入式工作流与服务化工作流对比

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **对比项** | **嵌入式工作流** | **服务化工作流** | **备注** |
| 1 | 部署方式 | 分散在各业务系统应用中 | 基于平台2.0独立部署 |  |
| 2 | 运行是否依赖业务系统 | 是 | 否 |  |
| 3 | 流程模板定义管理 | 分散管理 | 集中管理 |  |
| 4 | 是否平台统一认证 | 否 | 是 |  |
| 5 | 是否提供流程监控 | 否 | 是 |  |
| 6 | 业务流程引擎位置 | 本地 | 本地（客户端） |  |

综上所述，服务化工作流相对于嵌入式工作流具有以下优势：

1. **服务安全认证，有效掌控系统应用**

服务化工作流提供业务系统安全认证，对所有使用工作流的业务系统进行登记注册和应用认证，实现对工作流使用情况的初步掌控。

1. **统一流程配置，规范流程应用**

服务化工作流对业务系统流程模板定义进行统一管理，统一分发，规范使用。

1. **流程流转监控，为流程预警、流程优化等奠定基础**

服务化工作流提供业务流程流转监控，对所有业务系统业务流程流转情况进行事后监控，并对监控数据（包括流程流转情况和节点状态等）进行统计和分析，为后续财政业务流程优化工作提供依据和支撑。

1. **影子流程，支持第三方业务系统流程监控**

服务化工作流通过建立影子流程，支持在第三方业务系统存在自身工作流引擎时，对第三方业务系统业务流程流转的监控。

1. **流程引擎本地化，有效保证工作流高性能**

服务化工作流分工作流主服务端及客户端，驱动业务系统业务流程流转的引擎仍保留在客户端中，业务系统调用本地客户端JAR包中流程引擎即可完成流程流转，有效保证工作流高性能。