
基于构件化的 GIS 工作流中 间件

整体解决方案

成都领君科技有限公司

成都领君科技有限公司

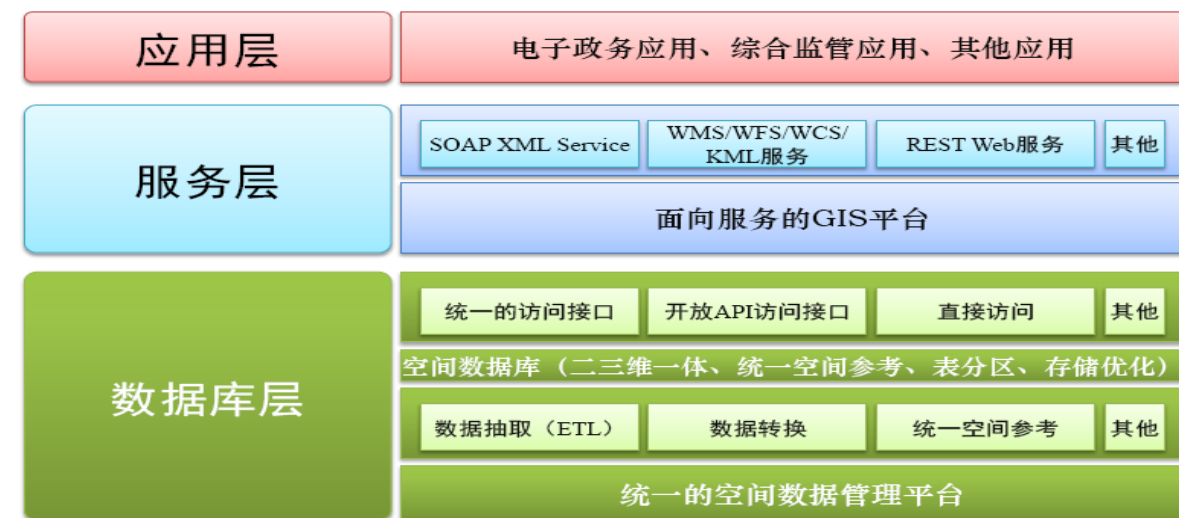
2012 年 4 月

目 录

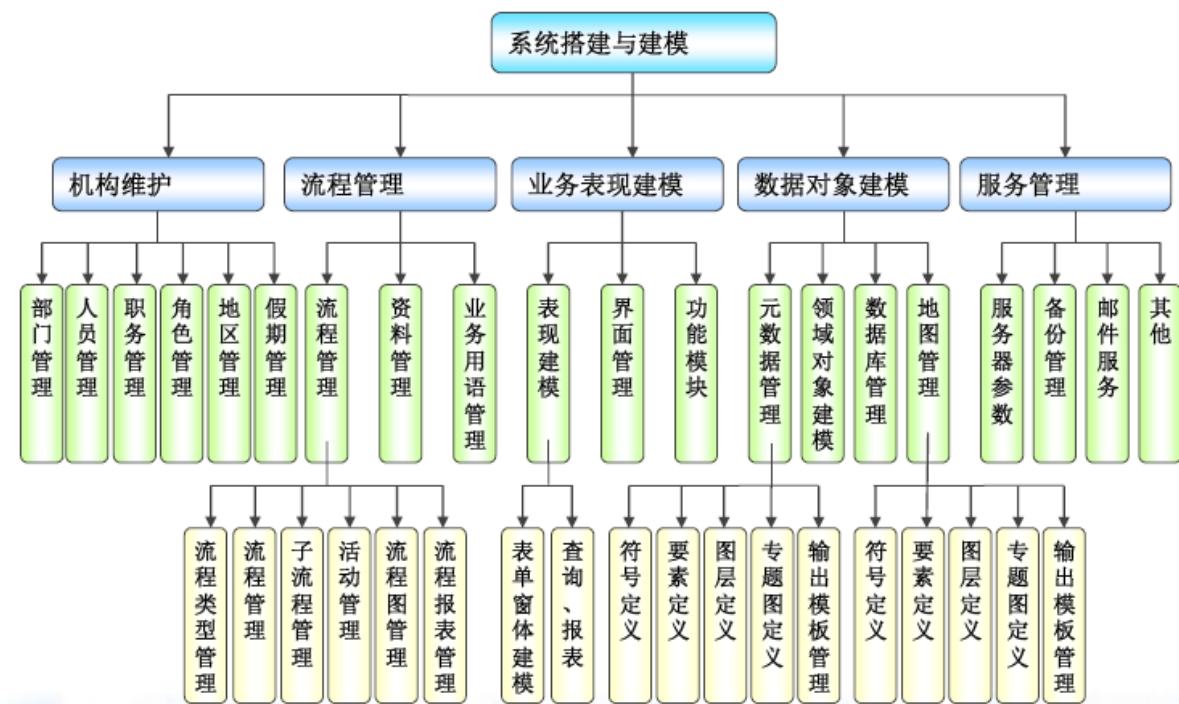
1. GIS工作流中间件平台简介	1
1.1 GIS工作流中间件技术支撑平台	1
1.2 GIS工作流中间件平台构成	2
1.3 GIS工作流中间件平台实现目标	3
2. 平台设计思路与控制模型	5
2.1 数据模型	5
2.2 平台引擎的控制模型	8
3. SuperFlow工作流平台简介	12
3.1 平台技术标准	12
3.2 部分特殊功能点	13
3.3 流程控制模式	13
3.4 建模功能	14
3.5 SuperFlow工作流平台特点	19
4. JLKEngine中间件平台简介	20
4.1 JLKEngine中间件主要功能	20
4.2 JLKEngine中间件特点	22
5. SuperFlow与JLKEngine的无缝集成	23

1. GIS 工作流中间件平台简介

1.1 GIS 工作流中间件技术支撑平台



平台技术架构图



电子政务搭建平台

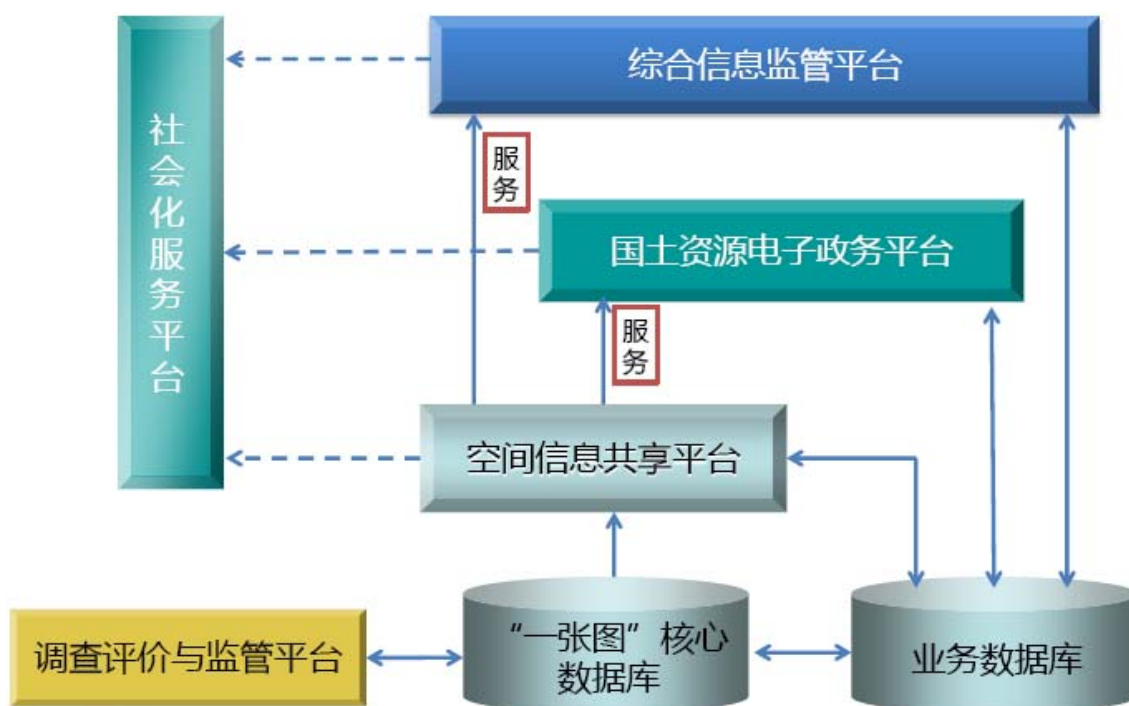
GIS 工作流中间件技术支撑平台即电子政务搭建平台，是基于 GIS 应用中对空间数据处理的特别要求以及 GIS 领域电子政务管理系统的业务需求，基于分布式多层构架和构件技术，采用空间数据处理和电子政务应用系统开发工具进行构建的。该支撑

平台的构件包括消息通信组件、事件管理组件、流程控制组件、数据管理组件。这些构件进行有机装配，实现协同办公。

该技术支撑平台在技术平台上选择先进的 .NET 技术，采用业务分布、数据集中管理的方式，增强数据的安全性。在具体实现上，采用三层体系结构体系，大大减少了系统客户机部分与服务器部分的依赖程度，提高了整个系统的可维护性。采用基于 XML 标准的信息交换技术和 Web 服务等事务处理实现异构数据的共享。

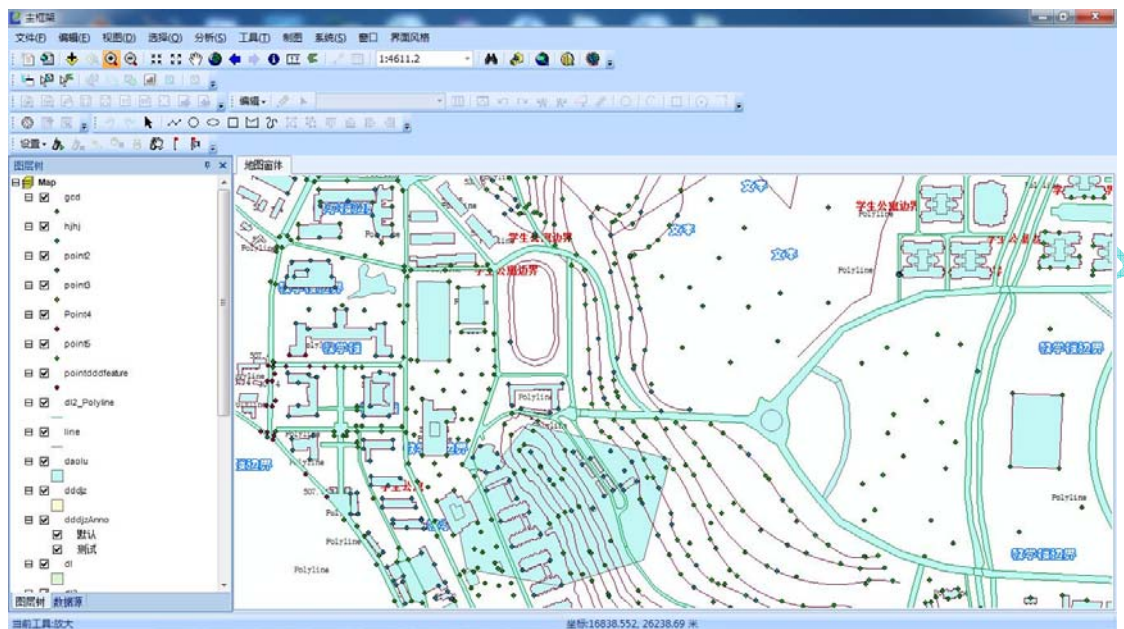
1.2 GIS 工作流中间件平台构成

基于 GIS 工作流中间件平台搭建国土资源一张图综合监管平台设计架构图如下所示：



平台主要两部分组成：即由基于C#+ArcEngine开发的JLKEngine中间件和基于ASP.NET+C#开发的SuperFlow工作流平台组成。JLKEngine中间件是按ArcEngine对象的处理为目的进行的AO底层类库的二次封装，主要实现以下三大功能：一要解决数据管理；二要提供后台服务和拥有二次开发能力；三是要提供基础的应用。SuperFlow工作流平台是按照WFMC工作流标准联盟提出的要求并结合目标行业特点进行设计，

主要包括以下几个方面的内容，数据库建模、业务信息建模、功能环节建模、业务流程建模、组织机构建模等。基于工作流的GIS应用办公系统可自动嵌入JLKEngine中间件、WebGIS栅格平台，将GIS应用相关功能，嵌入到工作流中进行统一调度。



1.3 GIS 工作流中间件平台实现目标

(1) GIS 与工作流一体化

本平台并不是 GIS 组件与工作流的简单组合，两者所起的作用完全不同，但相辅相成。JLKEngine 中间件负责数据处理和具体业务功能的实现，SuperFlow 工作流负责将各种不同类型的数据和业务描述进行逻辑组织，达到在非集中控制的环境中协同使用资源的目的，最终实现 GIS 与工作流的一体化结合，如下图所示。



GIS 工作流系统功能基于成都领君科技 JLEngine 中间件平台+SuperFlow 工作流平台构建系统框架，支持自定义表单、工作流搭建式 Web 界面开发，能很方便的进行功能扩展。根据建设目标，“一张图”管理系统的数据管理子系统（C/S）包含：数据入库、数据管理、数据编辑与处理、图形查询分析、成果输出、综合统计、数据交换、数据更新管理、数据维护等功能模块。Web 发布子系统包含数据发布、业务审批决策支持、输出上报等模块。

(2) 业务处理过程标准化

本平台最终目标用户群体比较广泛，业务应用本身具有较大差异，需要不同的处理过程和基础算法，所以，本平台在设计阶段需要高度抽象，确定标准化的最小处理单元，各单元经过系统逻辑组织，重构各项具体业务处理过程，通过系统逻辑检核确保处理过程的稳定性和可用性。

(3) 业务控制环节智能化

在社会组织的各项活动中，业务控制（如：审核、校对、审批、批阅、审查等活动）活动非常普遍，通常也是一个系统是否成功推广的关键环节。所以，本平台在该环节大量引用先进技术和处理手段，如：网络指纹识别技术、条形码、手写板、扫描识别等。在提高审查环节的安全性同时，确保该环节的工作尽量简单化，尽量适应操作人员现有的技能。

(4) 简单 GIS 应用系统零代码实现

QQ: 26326507
电话: 13982071140

<http://www.linjon.cn>
E-Mail: hy2001al@163.com

应用本系统平台提供的建模工具，从用户需求调查完成后的整个设计过程，采用系统数据建模功能创建数据库、采用表单定义工具定义业务环节中使用的各种业务表单，采用 JLKEngine 中间件定义地理数据格式，建立图形数据库、采用可视化工作流程定义工具定义业务处理流程，在一般小型项目中可以实现零代码编程，如果对于部分复杂的逻辑运算不能完全实现，可以采用编写处理脚本的方式嵌入到业务处理过程中，达到快速开发的目的。

2. 平台设计思路与控制模型

2.1 数据模型

基于构件化的 GIS 工作流中间件应用平台引擎的数据模型包括机构模型和信息模型两部分。机构模型描述的是企业或者部门的组织机构关系，信息模型则定义工作流引擎中所用到的各种控制数据。通过数据模型，可以方便地描述关键业务的业务规则、活动的依赖关系以及任务的指派等特征。它们都通过统一的关系结构来定义。图 1 给出了基于关系结构的轻量级工作流引擎的数据模型的 ER 图。

(1) 机构模型

与机构模型相关的表主要有 **STAFF**、**DEPARTMENT**、**TEAM**、**STAFF_TEAM**、**ROLE** 和 **STAFF_IN_ROLE**，表之间的关系已经在图中通过不同含义的连线标出。

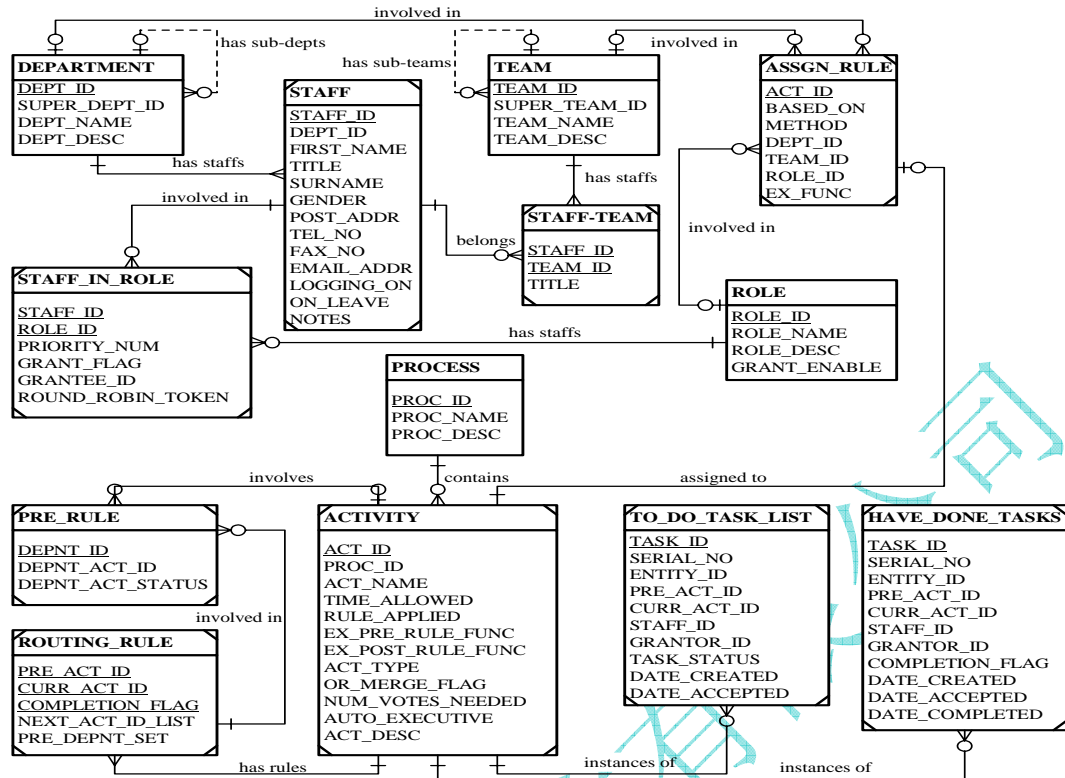
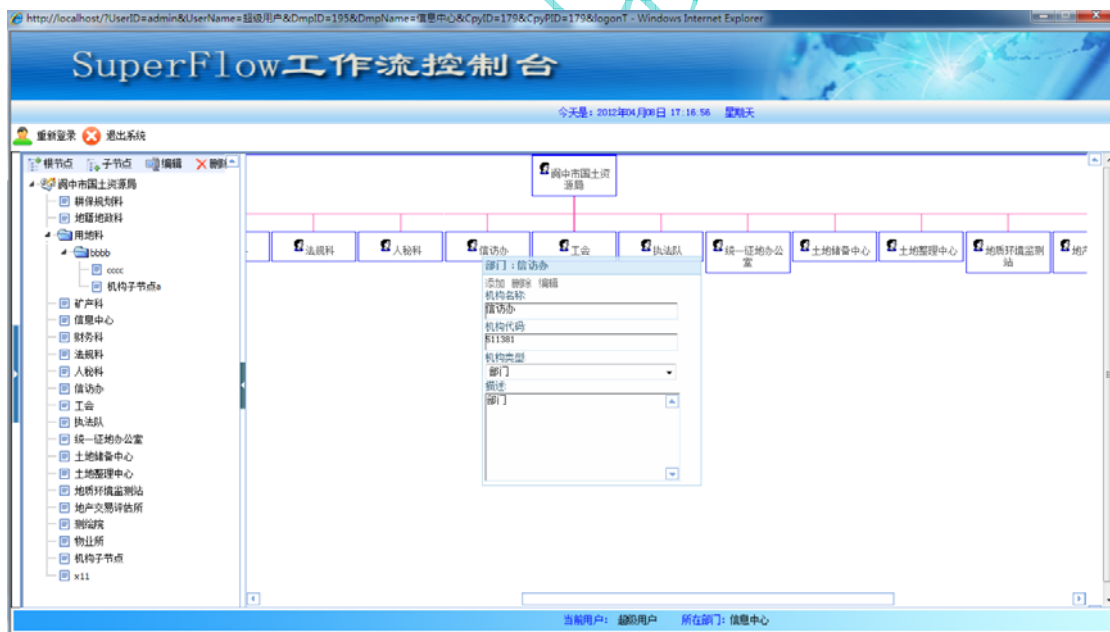


图 1 数据模型 ER 图



(2) 信息模型

信息模型的核心是业务活动表（简称活动）**ACTIVITY**，其他相关的表结构主要有业务过程 **PROCESS**、业务规则（活动流转规则）**ROUTING_RULE**、活动前依赖规则 **PRE_RULE**、任务指派规则 **ASSGN_RULE**、任务列表 **TO_DO_TASK_LIST** 以

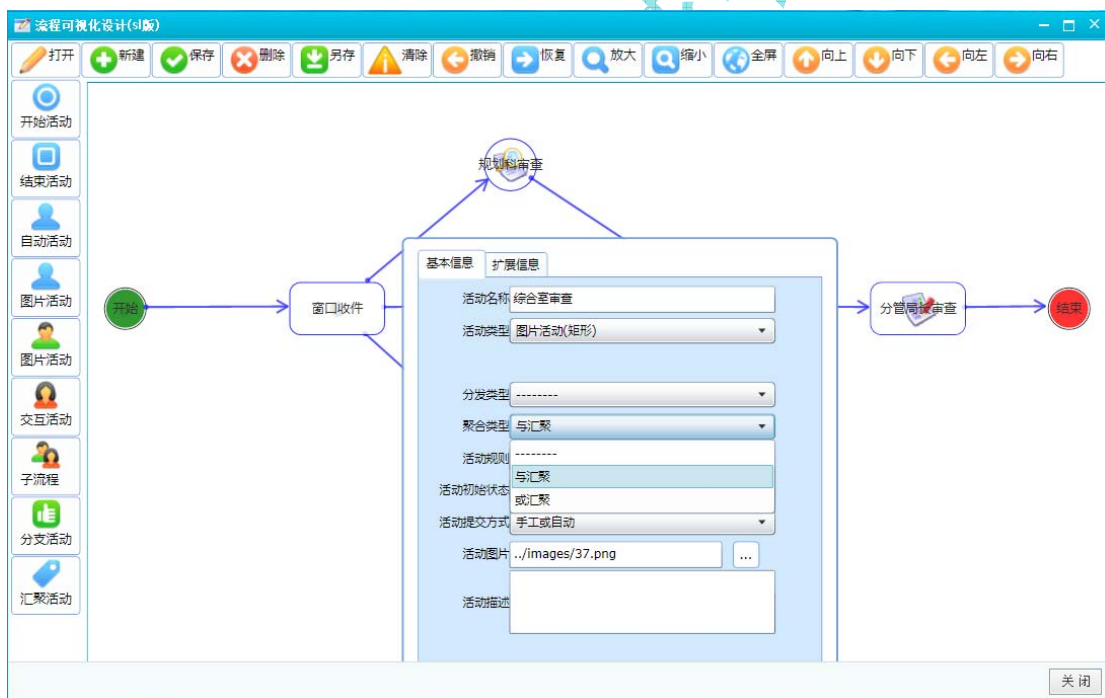
及已完成的任务列表 **HAVE_DONE_TASKS**。

(3) 活动类型

● 每个业务过程由若干业务活动组成，不同的业务活动通过各不相同的 **ACT_ID** 来唯一标识，**ACT_TYPE** 则指明相应活动的类型。同一个业务活动在工作流运行时可能具有多个实例 (instance)。我们将活动的实例称为任务¹，将属于同一业务过程的任务称为属于同一批次的任务。有的业务活动可能针对具体的业务环节，即在前台（后台）对应实际的应用逻辑；有的业务活动则不针对具体的业务环节。

(4) 业务规则的表示

在工作流引擎中，业务规则可以分解成活动的前依赖规则和活动的后转发规则。活动的前依赖规则指明相应活动的启动条件，启动条件是通过相应活动的直接前趋活动以及相应的状态标志来表示的，前依赖规则包含**顺序**、**与汇聚**、**或汇聚**和**投票汇聚**四种规则。活动的后转发规则指的是当前活动所对应的任务结束后该启动哪些后继活动，后转发规则包含**顺序**、**或分支**和**与分支**三种规则。



(5) 任务队列和已完成任务队列

一个活动可以同时具有多个实例，即任务，这些实例可以是属于同一批次的，也可能属于不同的批次，流水号 **SERIAL_NO** 用来标识任务所属的批次，所有属于同一批次的任务具有相同的流水号；不同的任务之间则通过唯一的 **TASK_ID** 进行标识。

(6) 任务指派

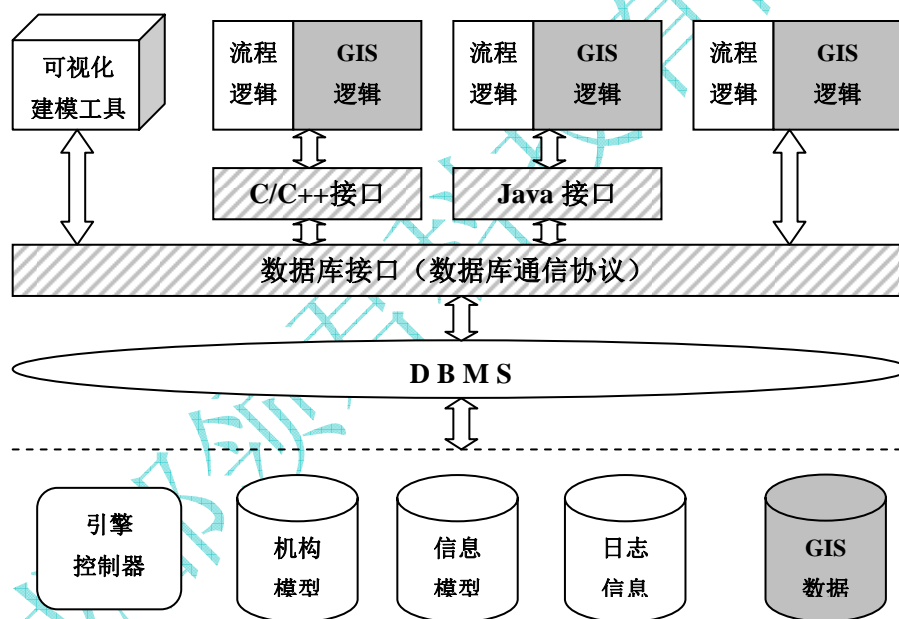
任务指派指的是依照何种准则将任务分配给具体人员来执行。只有常规交互活动才涉及到任务指派的问题；其他活动要么在前台不具备实际的应用逻辑，要么由工作流引擎自动调用，因此与任务指派没有关系。

2.2 平台引擎的控制模型

控制模型将机构模型和信息模型有机地结合在一起，它根据其中定义的业务规则对业务过程中的各项业务活动的流转以及任务指派等工作进行控制和协调。控制模型是工作流引擎的控制中心。

(1) 应用框架

图 2 给出了使用本文所提出的工作流引擎的应用系统的框架结构。



如图 2 所示，“可视化建模工具”即采用一套恰当的图示化的工具来对业务过程进行描述，然后将其转换成如机构模型和信息模型中所述及的关系结构，从而建立起工作流引擎的数据模型。因此，“可视化建模工具”是工作流引擎在构造时的定义中心，而“引擎控制器”则是工作流引擎在运行时的控制中心，它负责工作流引擎在运行时的协调、调度和控制功能。根据具体应用的开发环境的不同，工作流引擎在应用框架中为不同类型的应用提供了不同的接口，例如 C/C++ 接口、Java 接口以及直接基于数据库通信协议的接口，从而为不同类型的应用与工作流引擎的交互提供了方

便。应用框架中的“应用数据”则由具体的应用逻辑自行管理，工作流引擎并不关心这部分的数据格式。

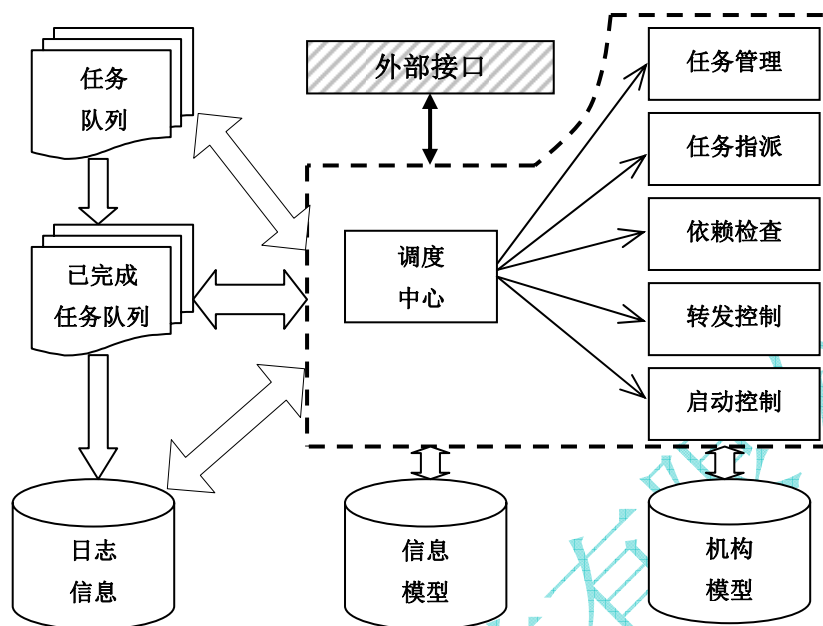


图 3 引擎控制器结构图

(2) 引擎控制器

引擎控制器是工作流引擎在运行时的控制中心，图 3 给出了引擎控制器的控制结构图。

● 调度中心

调度中心接受从外部接口发送过来有关流程控制的请求（如业务初始化、获取任务以及结束任务等），然后根据不同的请求类型调用相应的处理模块完成与本次请求相关的操作并将结果返回。由于是在 DBMS 内部实现工作流引擎的控制模型，因此有关请求的并发处理等问题完全可以交给数据库管理系统来完成，也不需要诸如请求队列等形式的数据结构。因此，事实上可以将调度中心看成一个多线程的并发服务器，它可以对多个外部请求提供并发服务。对外部请求的处理过程中肯定会涉及到对内部数据结构（即工作流引擎的数据模型）中有关数据的读写和更改操作，这些数据的完整性和互斥操作则可以通过 DBMS 提供的各种加锁机制来实现，从而实现了多个外部请求之间的独立性。

● 任务管理

任务管理主要根据调度中心的指示完成诸如任务创建、任务状态的转换以及相关

数据的维护等工作。每次“结束任务”的外部请求将触发调度中心调用“任务管理”为后继活动（如果存在的话）创建新的实例，其状态为“**Pending**”；同时，其他不同的外部请求也将触发“任务管理”实施任务状态的切换。任务状态转换图如图 4 所示。

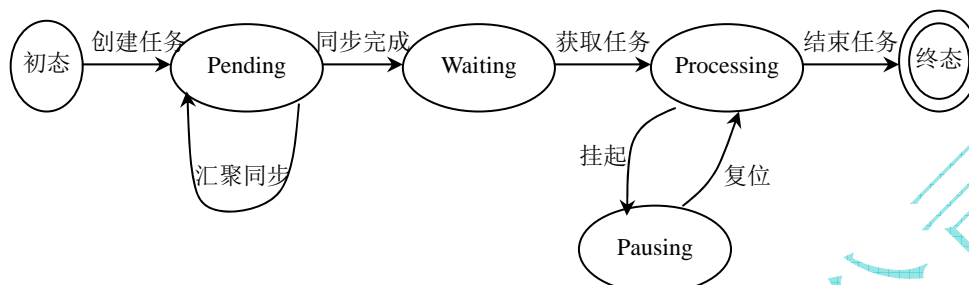


图 4 任务状态转换图

● 任务指派

任务指派处理只是针对常规交互活动，通常情况下，在任务状态由“**Pending**”切换到“**Waiting**”过程中完成任务的指派工作，即处于就绪状态的任务在通常情况下都确定了其执行者（**FCFA** 除外）。任务指派过程首先根据任务指派基准确定可以执行此任务的群体人员，通常情况下这是一个包含多个人员的集合；然后根据任务指派方法确定由这个群体中的哪些个体来执行任务，执行任务的个体标识记录在相应任务记录的 *STAFF_ID* 字段中。在 3.4 节中已经对任务指派方法进行了解释，这里有两点需要特别强调：

- 1) 如果任务指派方法是“**ALL**”的话，将对当前的任务记录进行拷贝，即保证每一执行任务的个体在 *TO_DO_TASK_LIST* 中都有一条对应的记录；
- 2) 如果任务指派方法是“**FCFA**”的话，事实上在任务指派阶段不作任何工作，即相应任务记录的 *STAFF_ID* 字段为空。此时任务指派工作自动隐含在获取任务的请求中，即谁先发出获取任务的请求，就自动将此类型的任务分配给谁。

● 依赖检查

依赖检查指的是活动的前依赖规则的检查，调度中心在将任务切换到就绪状态之前将进行相关的前依赖规则检查，只有满足检查条件的任务才可以进行状态的切换。前文已经描述了前依赖规则在数据模型中的表示方法，这里主要讨论在控制模型中是如何对各种前依赖规则进行处理的。

对于顺序前依赖规则，很显然，从前趋活动流转是当前活动跟其他前趋活动没有

关系, *PRE_DEPNT_SET* 为空集, 当前活动的启动没有其他约束条件, 相应任务可以立即由 “**Pending**” 状态转换到 “**Waiting**” 状态。

对于与汇聚前依赖规则, *PRE_DEPNT_SET* 中指明所有参与与汇聚的其他前趋活动, 只有所有相关的前趋活动均到达各自指定的结束状态 *DEPNT_ACT_STATUS*, 当前活动方可启动。

对于或汇聚前依赖规则, *PRE_DEPNT_SET* 为空集, 此规则的检查将涉及到 *ACTIVITY* 表中的 *OR_MERGE_FLAG*, *OR_MERGE_FLAG* 的取值可以是所有相关的前趋活动的结束标记之一或者是一个特殊的标记 “**ANY**”。如果 *OR_MERGE_FLAG* 的值不是 “**ANY**”, 则将检查相应前趋活动的结束标记 *COMPLETION_FLAG* 是否与 *OR_MERGE_FLAG* 相同, 若相同, 则启动当前活动, 若不相同, 则不作任何处理; 否则, 如果 *OR_MERGE_FLAG* 的值为 “**ANY**”, 则首先结束的前趋活动将启动当前活动, 后结束的活动将被丢弃。

对于投票汇聚活动, *PRE_DEPNT_SET* 同样为空集, 当前活动要等到属于同一批次任务数目达到 *NUM_VOTES_NEEDED* 的要求方可启动。属于同一批次的任务数目可以通过对 *TO_DO_TASK_LIST* 按照 *ACT_ID* 和 *SERIAL_NO* 进行统计得到。

● 转发控制

当应用发出 “结束任务” 的外部请求时, 该请求将触发调度中心启动 “转发控制”。转发控制的主要依据在工作流数据模型中定义的后转发规则, 后转发规则定义了当前活动与其后继活动之间的关系。转发控制的处理过程是根据 “结束任务” 请求中所携带的 “任务结束标记” 以及相应前趋活动和当前活动的活动标识匹配 *ROUTING_RULE* 表中的记录, 从而得到相应的后继活动列表 *NEXT_ACT_ID_LIST*; 然后由调度中心根据后继活动列表启动 “任务管理” 为相应的后继活动新建任务。

对于顺序转发以及或分支转发规则, *NEXT_ACT_ID_LIST* 只包含一个活动; 对于与分支转发规则, 则 *NEXT_ACT_ID_LIST* 中将包含多个活动。

● 启动控制

启动控制负责常规自动活动的所对应的自动执行体的启动并对其活动进行监控。

3. SuperFlow 工作流平台简介

基于 ASP.NET+C# 开发 SuperFlow 工作流平台，运行于网络环境，支持分布式数据库管理，提供跨地区的信息管理模式，实现资源与数据的网络共享。结合 GIS 的应用特征和数据表现需要，工作流平台对处理单位组织结构、业务流转模式、业务内容划分等进行高度抽象，建立工作流模型和工作流建模工具，包括业务数据库建模、业务信息建模、功能环节建模、业务流程建模、组织机构建模等；对工作流产生的各种数据，包括工作流控制数据（Workflow Control Data）、工作流相关数据（Workflow Relevant Data）、工作流应用数据（Workflow Application Data）进行有效的组织管理。

3.1 平台技术标准

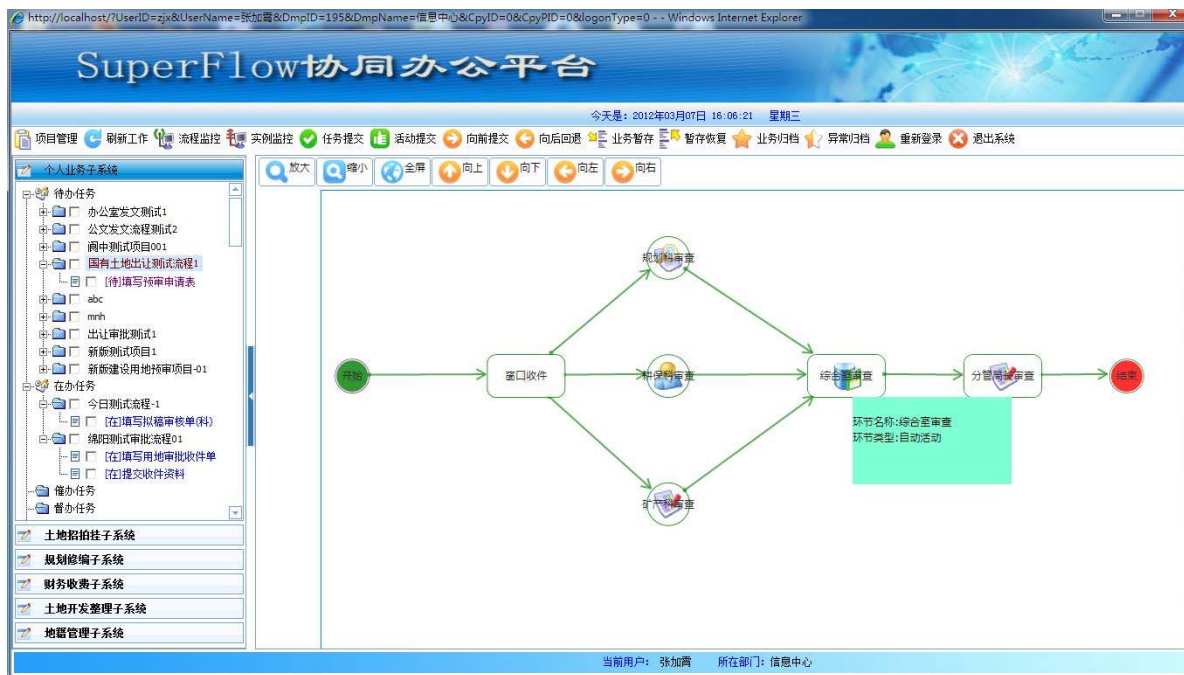
- 图形化流程图界面
- 模拟工作流应用测试
- 基于角色的路由
- 能够调用其他工作流进程
- 基于关系的路由
- 工作流协调和管理
- 条件型路由
- 工作流统计
- 自动调用第三方应用
- 自动化工作流导入和维护
- 与数据库的无缝链接
- 基于 JLKEngine 中间件的应用插件无缝嵌入
- 任务列表
- 例外处理

3.2 部分特殊功能点

- 影像处理：扫描、编码、OCR
- 审批手段：指纹识别、条形码
- 三方插件接口
- 数据打包
- 远程加密数据交换
- 数据解包
- 模板库定制与引用
- 流程嵌套
- 分布式数据库
- 流程监控
- 流程日志
- 流程启动与终止
- 新旧流程的同时流转
- 自定义报表
- 流程挂起/启动
- 注销流程
- 组织管理

3.3 流程控制模式

顺序 (Sequence)、并行分叉 (Parallel Split)、同步 (Synchronization)、排它选择、简单合并、多路选择 (Multiple Choice)、多路合并 (Multiple Merge)、路径鉴别器 (Discriminator)、M 中选 N 合并 (N-out-of-M Join)、同步合并 (Synchronising Join)、任意循环 (Arbitrary Cycles)、绝对终止 (Implicit Termination)、多实例 (设计时已知实例数目)、多实例 (运行时才知实例数目)、多实例 (无预知)、多实例 (要求同步的多实例)、延期选择 (Deferred Choice)、交叉路由 (Interleaved Routing)、里程碑 (Milestone)、取消任务 (Cancel Activity)、取消流程 (Cancel Case)



3.4 建模功能

(1) 数据库建模



支持分布、异质数据库：支持分布、异质的数据库方案。

强大和丰富的扩展属性定义：为数据建模提供了强大和丰富的扩展属性定义，使数据库应用中的大量功能可以在免代码的情况下实现。

复杂、交互、可视的商务规则程序：支持高度复杂、交互、可视的商务规则程序，以支持实现复杂的商务逻辑和业务要求。

对企业现有数据库的直接重用和访问：支持对企业现有数据库的直接重用和访问。可以直接从数据库中抽取并产生数据定义。

数据字典资源的导入、导出：可以对数据字典资源进行导入和导出。

业务语义属性：为数据库引入了一个业务语义层，通过这一业务语义层，业务人员就可以依据业务属性来管理和操控数据。

业务参数管理：提供统一的业务要素和参数管理，可以让用户灵活修改业务活动中的大量业务参数。

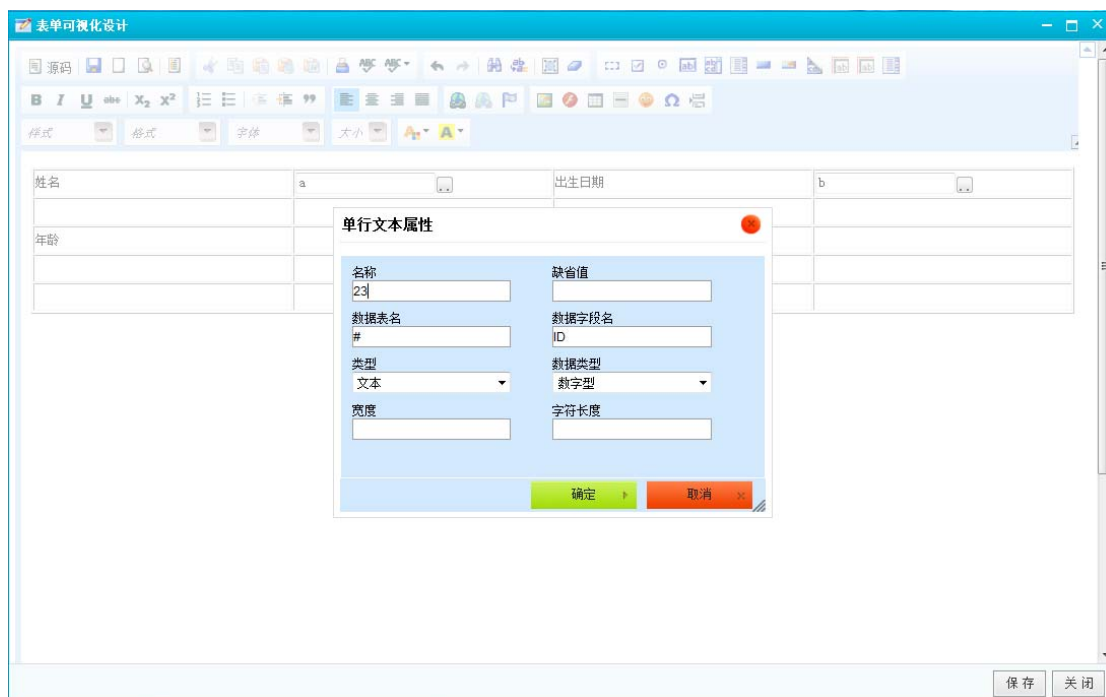
强大的数据限定和查找定义：支持各种不同数据来源、不同表现风格、不同操作方式的数据限定和查找定义。

字段级数据加密：支持针对每一个字段的数据加密。

(3) 业务信息建模

丰富和强大的业务表单支持：业务表单除了支持常用表格类型、标准数据窗体、RTF 文档、电子表格、HTML 表单、各类 OLE 文档(Word、Excel、AutoCAD 等)、等，开发人员可以利用他们轻而易举的创建各种形式多样、风格各异的业务文档。

所见即所得的表单设计：企业中的大量业务，都是通过表单来表现的，由此可见表单在一个企业应用中的重要性，系统业务信息建模工具，对于复杂界面表单的设计，均提供了功能强大的所见即所得的表单设计方法。



多表单复合信息支持：在一个复杂的业务信息中，往往需要处理多种格式和表现业务表单，如表格、图标、树形结构等，系统支持将多种格式的表单复合在同一个信息窗体内，以实现高度复杂、灵活和强大的信息表现。

表单与数据的连接以及输入自动感知：系统可以在上述大多数类型的表单中定义数据域，实现数据输入输出。所有类型表单均使用统一的数据接口，便于数据感知表单的设计。

常用数据感知格式表单的自动生成：只要用户选择了数据源和表单类型，系统就能够根据数据字典的定义，智能地生成相应格式的表单。现有系统的表单自动生成的智能程度已经相当高，对于要求一般的输入界面，直接使用系统自动生成的表单即可。如果要求更高，则可以在自动生成的表单基础上再加以修改。

复杂表单窗体的设计：系统对所支持的不同类型表单提供了统一的表单窗体设计支持，让开发人员在各类表单基础上进行可视化的窗体设计，添加各类组件，编写控制代码，完成复杂的业务功能等等。

表单视图动态测试：所有业务表单，系统都支持在设计期任何阶段对设计的商务规则、交互界面、运行效率、数据编辑、打印输出等重要特征进行动态测试。

(3) 功能环节建模

管理功能环节所处理的信息：支持任意类型、任意数量的表单。

管理功能环节的操作行为：定义功能环节中对业务信息的操作行为，如是否允许增加、删除、修改，可以指定特定域的操作行为。

业务信息的特殊表现形式：对所处理的业务信息指定一些特殊的表现形式，如域的隐藏，标题，表单处理顺序等等。

定义数据范围：把所处理的业务信息的数据范围限定在一定的条件范围内。
支持复杂的功能界面设计和开发：在自动生成和资源化描述的基础上，系统能够对业务界面进行复杂编程和设计、以满足用户对特殊功能业务的要求。

支持 SuperFlow 工作流平台内所有功能、信息、表单和数据对象的重用和访问：只要是建立在 SuperFlow 工作流平台上的应用，开发人员均可以重用其所有的功能、信息、表单和数据对象。这些业务的对象完全以可视化的设计方式进行。

(4) 业务流程建模

支持专业的流程建模：支持各种可能的复杂流程定义，包括流程的合并，分流，事件驱动，逻辑控制等等。

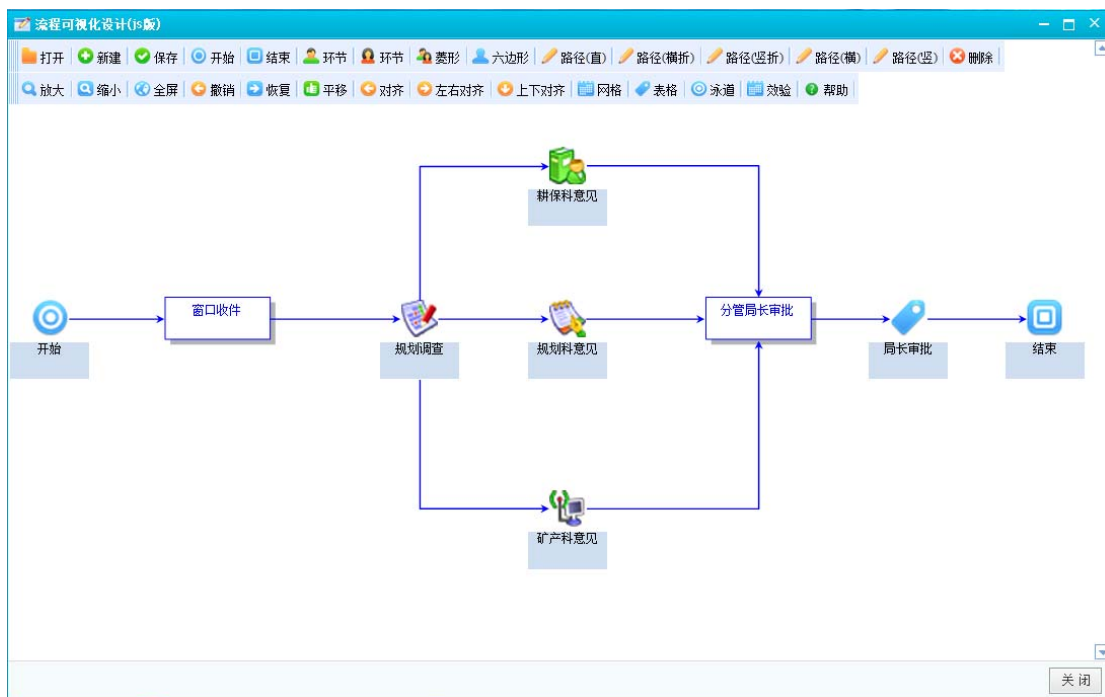
流程协作定义：支持复杂流程协作，包括复杂的人员，部门协作，默认和强制合作方式等。

任务协作定义：支持复杂的任务协作模式，包括任务分配模式、执行模式、抢占模式等等。

事件驱动和流转条件控制：通过事件机制，实现灵活、分布式的流程管理和条件控制。

时间控制：支持流程和环节的时间控制，包括时间限定，提醒方式等等。

数据传递定义：数据传递定义建模工具主要完成业务数据传递, 回填 并进行传递的条件定义。



(5) 组织机构建模

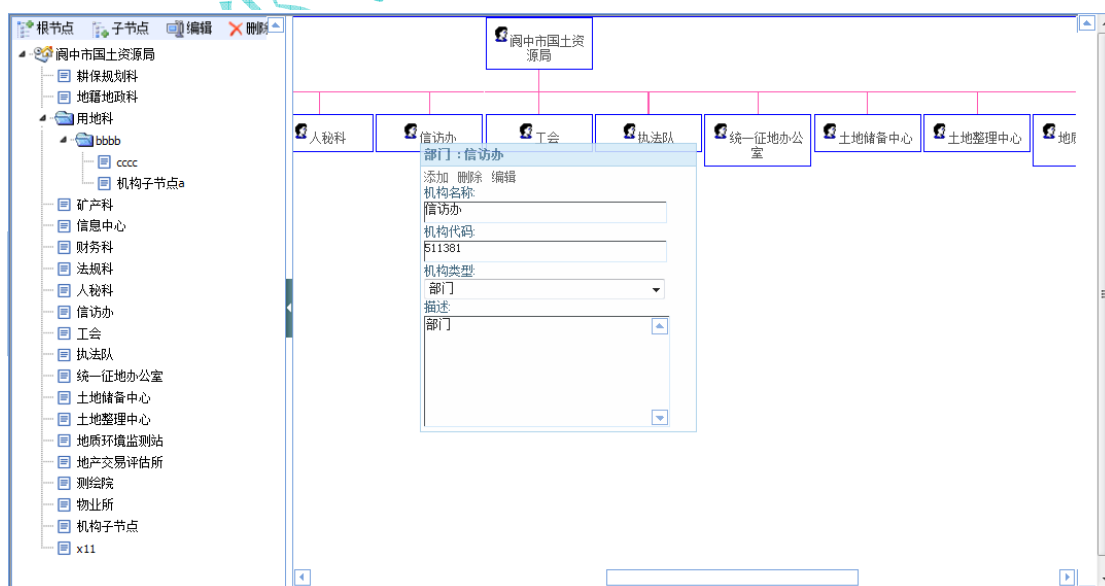
组织机构管理:包括机构、组织、部门、岗位、人员的管理。

支持工作组和矩阵式机构:除了传统的职能式组织机构外,系统也支持工作组和矩阵式机构。

业务分工:这是系统组织机构建模中最主要的一项功能,用以完成企业中部门、岗位、人员的业务分工。

业务管理权限:支持业务监控、上下级业务管理等。

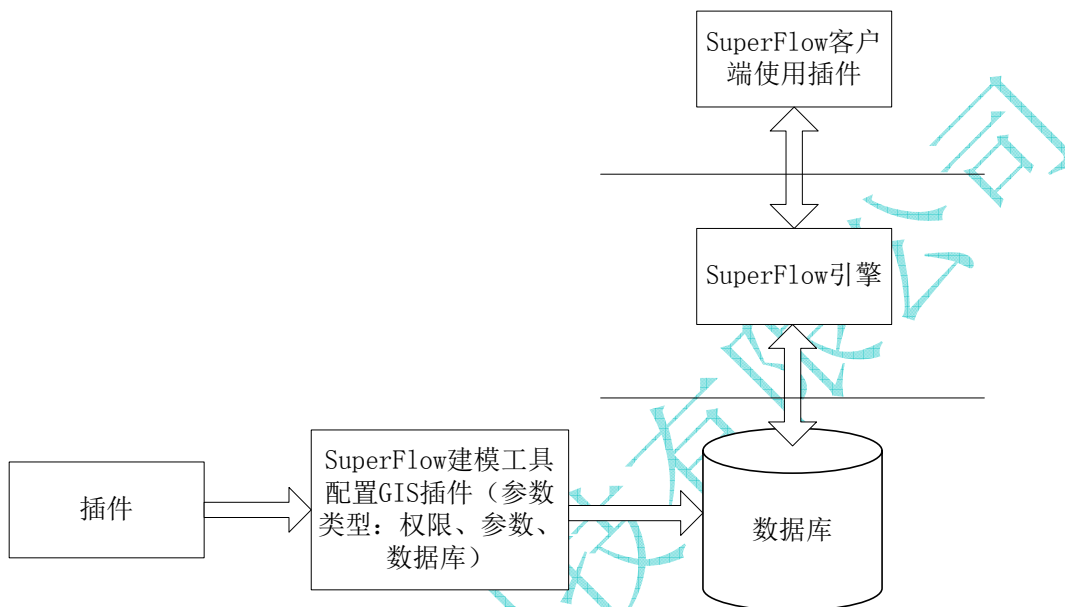
数据权限管理:支持复杂和灵活的业务数据权限管理。



3.5 SuperFlow 工作流平台特点

1) SuperFlow 工作流平台具有良好的可扩展能力

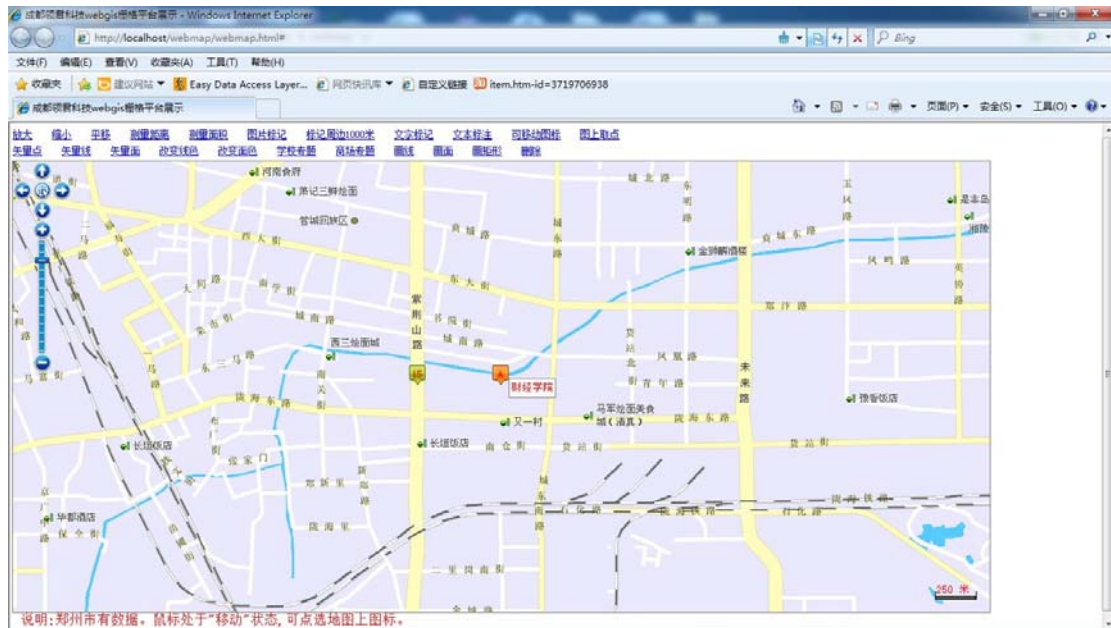
该中间件具有模型驱动机制，不仅能快速的构建系统，而且能建立新的模型、维护已有模型。另外，还具有插件机制进行应用系统的功能扩展。



SuperFlow 工作流平台设计了良好的系统扩展接口，提供了与 JLEngine 中间件，以及 WebGIS 栅格平台的交互。通过平台消息构件进行工作流与 GIS 图形操作之间的消息传递，通过工作流权限管理构件进行 GIS 图层操作的浏览、编辑控制。

2) SuperFlow 工作流平台具有良好的应用集成能力

SuperFlow 工作流平台提供了一个良好的应用集成框架，能够通过 Web Service 和基于 ArcIMS/ArcServer 的信息发布系统进行集成，并能有机将 GIS 应用插件嵌入到该平台进行 GIS 图形处理。



3) SuperFlow 工作流平台具有很好的开放性

提供建模平台：用户自主维护系统。

开放插件接口：用户或开发商均可扩展新功能，或集成其他系统。

提供功能组件：便于其他系统调用、集成本系统。

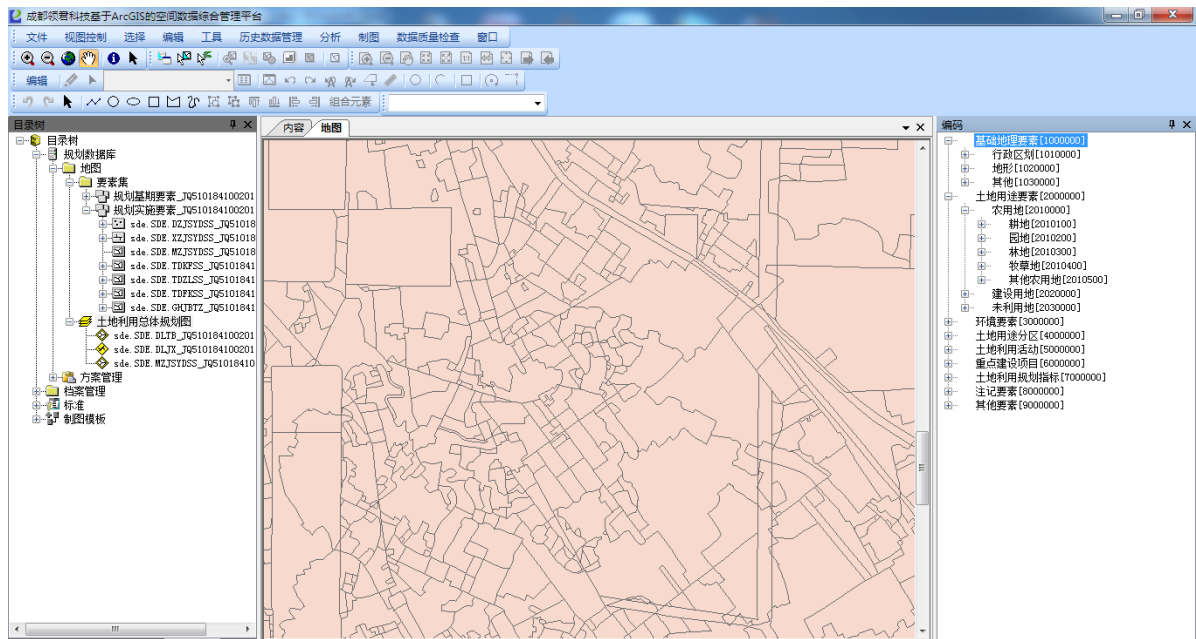
4. JLKEngine 中间件平台简介

4.1 JLKEngine 中间件主要功能

主要实现以下三大功能：一要解决数据管理；二要提供后台服务和拥有二次开发能力；三是要提供基础的应用。

1) 数据管理

数据管理模块具体包括属性数据管理、数据转换、数据备份以及数据统一管理等功能。另外，数据管理模块还提供对数据操作进行权限控制等功能。

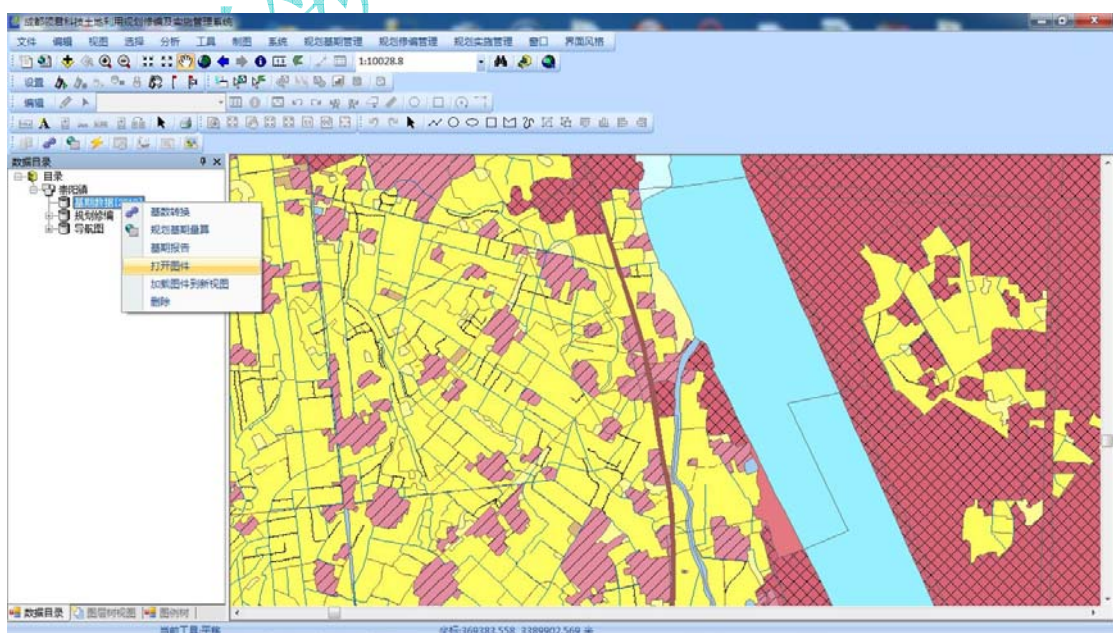


2) 后台服务

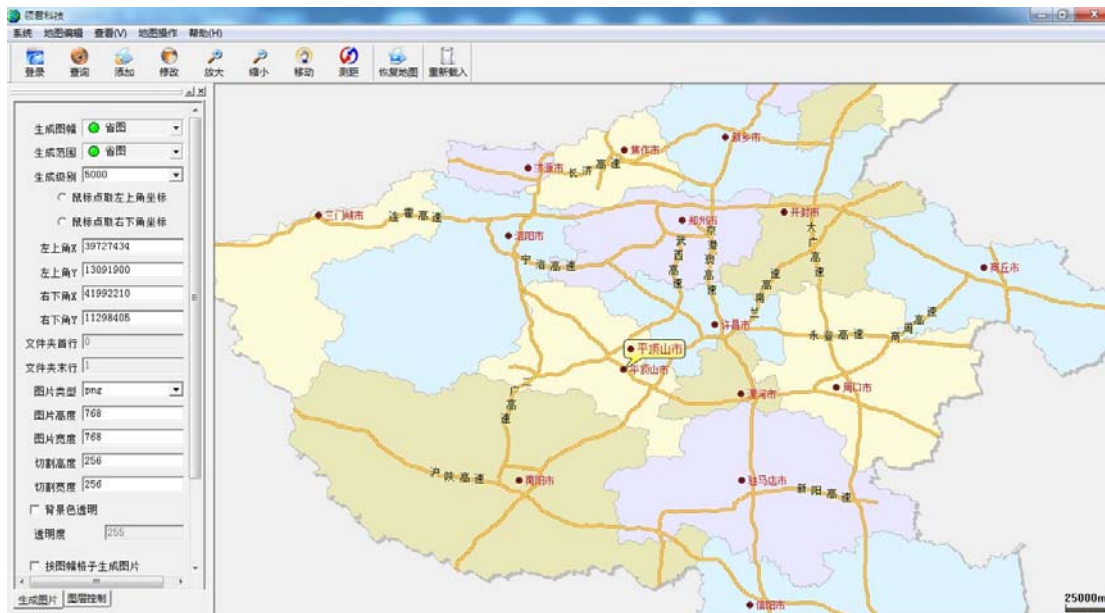
后台服务是 JLKEngine 中间件平台的核心，具有较高的开放性和二次开发能力。后台服务具有地图服务、属性服务、空间分析服务、元数据服务和专题图服务等，还提供强大的开发组件库，为 ArcServer Web 应用提供强大的二次开发接口。

3) 应用框架

基于 JLKEngine 中间件平台提供基于 GIS 应用系统所必需的基础应用模块。该应用框架主要包括地图浏览、地图编辑、信息查询、空间分析、专题图制作、数据共享、符号库管理等应用模块。



基于 HTML5+js 开发的瓦片式栅格地图发布管理平台，适用于几乎所有浏览器（IE6+，遨游，火狐，搜狗，谷歌，360，世界之窗，Opera 等）。功能模块包括如下两个部分：矢量地图生成切片工具和前台 javascript 调用切片开发包。



4.2 JLKEngine 中间件特点

1) JLKEngine 中间件具有良好的可扩展能力

JLKEngine 中间件设计了良好的系统扩展接口，提供了与工作流平台交互、ArcIMS 信息发布等扩展方式，软件开发商可以对该系统进行扩展，使之更加符合行业特点。

2) JLKEngine 中间件具有良好的应用集成能力

JLKEngine 中间件提供了一个良好的应用集成框架，能够通过 Web Service 和基于 ArcIMS 的信息发布系统进行集成，并能与工作流平台进行有机集成。

3) JLKEngine 中间件具有很好的开放性和嵌入能力

JLKEngine 中间件采用组件化设计，具有定义清晰稳定的 .NET 组件等类型接口，除了用于基于 ArcEngine 的二次开发应用外，还能够用于基于 ArcMap 的二次开发应用。

4) JLKEngine 中间件支持多服务器及多数据源管理

JLKGIS 中间件提供了类 ArcCatalog 的数据管理工具。可根据需要自定义显示资源图层、地物符号库、关联数据库、与业务管理数据库。支持对图层修改编辑，可添加、修改各种图形要素，修改对应属性数据，并可提供因地形地物变化后的图形更新

QQ: 26326507
电话: 13982071140

<http://www.linjon.cn>
E-Mail: hy2001al@163.com

功能。

5) JLKEngine 中间件具有实时信息监控功能

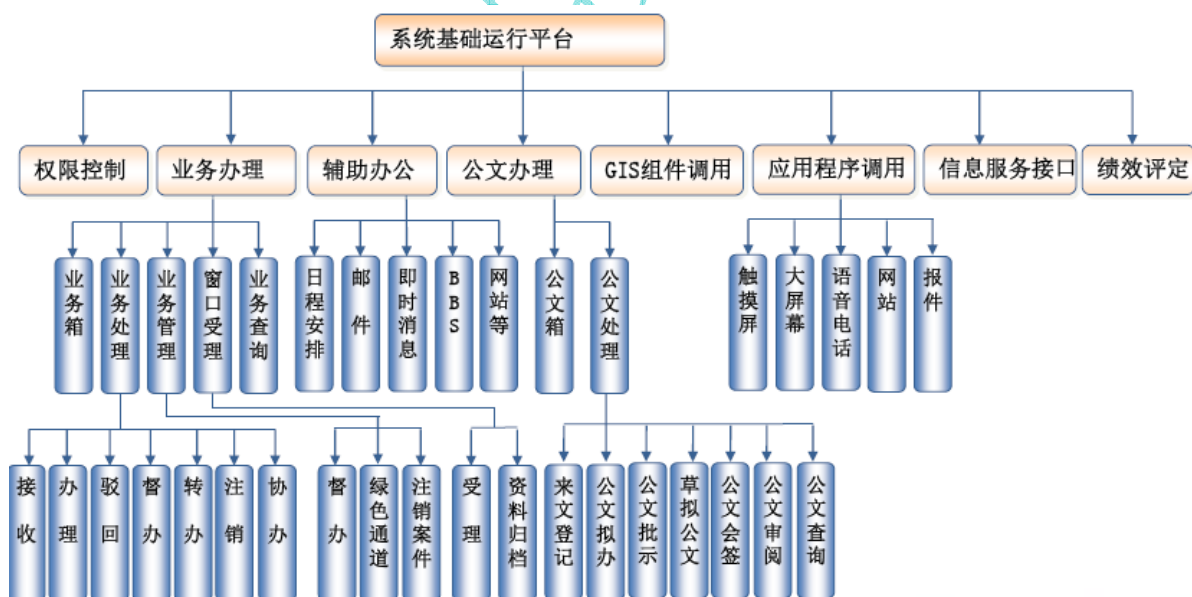
通过与其他各硬件设备和软件系统接口，完成信息实时显示的要求。获取视频监控系统的位臵信息，在电子地图上显示切换，实现视频信息的实时显示。在 GIS 地图上显示环境的变化情况和设备的工作情况。

6) JLKEngine 中间件具有快速故障诊断定位功能

系统支持多层设备关联，显示实际设备的连接状态，发生故障的情况下，系统立即锁定故障单元，并显示其影响区、及正常工作区设施路线。并可根据关联关系进行设施、设备统计，快速定位故障位置。

5. SuperFlow 与 JLKEngine 的无缝集成

基于 GIS 应用的电子政务管理信息系统需要在工作流办公系统中加入 GIS 功能，实现图文一体化统一管理，即由工作流平台统一控制管理 GIS 权限，实现业务办公系统与 GIS 系统的无缝集成。



由于审批过程是一个多角色参加多活动的流程，使用的具体 GIS 功能也将由于角色、活动的不同而有所区别，主要表现在：

地图内容不同：

根据不同的项目，用户应访问不同的空间数据库、空间数据集及图层。

地图图层显示不同：

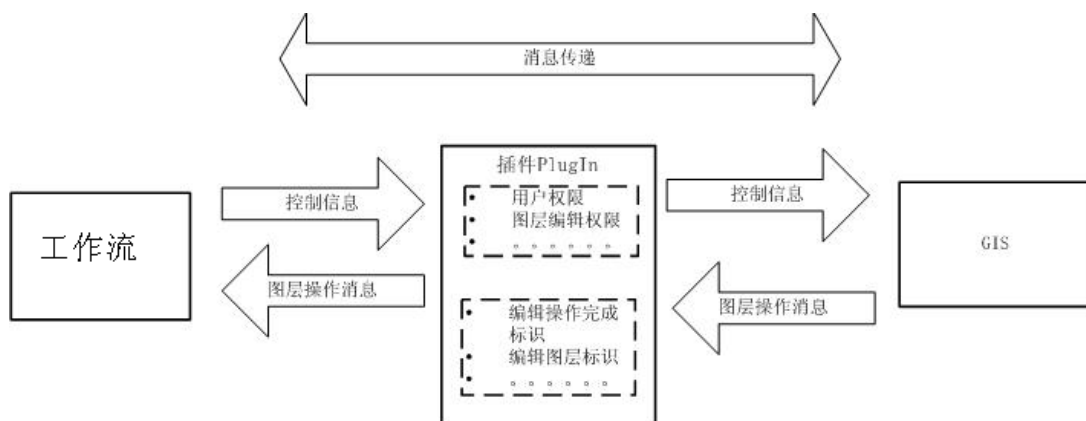
根据具体项目的不同，以及为了提高地图显示效率、突出显示效果等，用户应访问不同范围的地图，并设置显示最大最小显示比例、不同的图层的显示符号、标注、颜色等。

地图编辑权限不同：

为了保证数据安全性，根据活动不同以及权限的不同，地图使用的权限也有不同。例如：在执行业务审批过程中，经办员应具有编辑权限，制图完成并提交后，可供其他人员浏览、查询，但不可对地图要素进行修改。



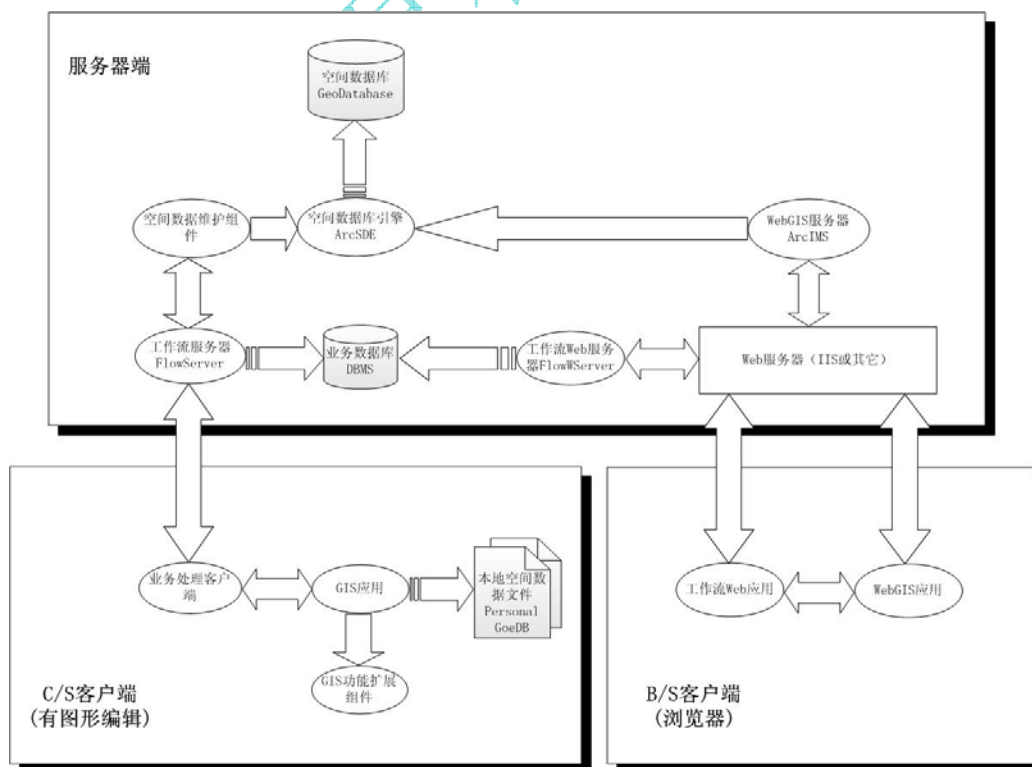
JLKEngine 中间件平台是以 ArcGIS 的 A0 底层进行构建的，提供标准的 COM 组件接口及 .NET 组件接口。因此，可以将上述 GIS 的可变因素抽取出来作为工作流平台的模型参数，由系统管理员统一配置管理，并充分利用工作流平台独特的插件技术实现业务办公系统与 GIS 系统的无缝结合（如下图），客户端只需解释和执行模型，在权限允许下使用不同的 GIS 功能，形成一个图文结合的统一平台。



工作流与 GIS 结合示意图

插件机制是采用开发插件并定制 GIS 应用的启动和退出方式来实现的。如上图所示，工作流将用户权限和图层编辑权限等控制信息写入插件并启动 GIS 应用后，等待 GIS 应用返回的操作消息，经过定制开发的 GIS 应用启动后首先读入这些控制信息，根据控制信息进行图层操作，完成操作后，将编辑完成操作标识和所编辑图层标识写回插件，工作流检测到这些图层操作消息后，将操作流程转入工作流程的下一个活动及执行相应的任务。在上述过程中，工作流平台控制了 GIS 相关的操作，实现了两者的无缝结合，为用户提供以业务审批工作流为主线的图文一体化办公系统。

按照对 GIS 应用需求的不同，系统将采取不同 C/S 与 B/S 混合模式来工作(如图)。



C/S 与 B/S 混合模式运行示意图

对于需要编辑地图的用户，系统采取 C/S 模式

用户通过业务处理客户端程序与服务器链接；

客户端正常的处理工作业务数据，GIS 部分则通过工作流的插件机制启动 GIS 应用，直接获得项目相关的审批、处理业务数据，相关数据作为 GIS 实体属性；

GIS 编辑操作针对的是本地 GIS 数据，编辑的扩展功能需要相应的功能组件支持，最后编辑好的 GIS 数据以及相关的属性数据作为业务流程处理的结果由业务处理客户端程序提交业务系统服务器。

对于一般的业务用户，系统采取 B/S 模式

用户通过浏览器与 Web 服务器链接；

业务处理及 GIS 应用都是基于 Internet 的，数据的交互是直接通过网页参数来实现的。其各自的服务器应用都需要各自相应的服务器应用程序支持的。

上述工作流与 GIS 无缝集成方案具有以下优点：

系统保持高效率

工作流通过插件接口与 GIS 系统进行消息传递，并不把 GIS 系统完全内嵌到工作流办公系统中，因此工作流办公系统并不影响 GIS 系统的效率，且避免了内嵌方式容易产生的异常错误。

系统具有统一维护界面

将业务流程审批中需要的 GIS 功能抽取出来固化为模型参数，交由工作流管理，工作流可对 GIS 功能使用权限、图层配置等模型进行统一管理，为用户提供了统一的管理界面。



工作流系统与 GIS 系统之间可实现数据交换

工作流和 GIS 系统都是开放的平台, 通过工作流的插件接口和 GIS 系统的开放接口, 两者之间可进行数据交换。例如, 在地图要素编辑操作中, 工作流可将流程实例编号等信息传递给 GIS 系统, 而 GIS 系统可将所操作地图要素的坐标信息传递给工作流。

可见, 利用工作流平台插件技术实现业务办公系统与 GIS 系统的无缝集成, 可充分实现图文一体化系统, 是非常适合基于 GIS 应用的电子政务管理信息系统的需求。

