

RDI Framework. NET

(.NET快速信息化系统开发整合框架)

workflow组件说明书

开发单位：技术研发部

编制日期：2014 年 11 月

修改日期：2015 年 05 月

产品使用说明书

文件状态： 【 <input type="checkbox"/> 】草稿 【 <input checked="" type="checkbox"/> 】正式发布 【 <input type="checkbox"/> 】正在修改	产品名称	RDIFramework.NET	文档名称	workflow component manual
	当前版本	V2.9	页数	81 页
	文档作者	EricHu、王进	完成时间	
	所属部门	技术研发部	密级	中
	文档审核	王进	审核日期	2014-11-27
	文档批准	EricHu	批准日期	2015-05-28

文档控制

[illegible]

* 修改类型分为 A—Added M—Modified D—Deleted

目录

第 1 章 引言.....	7
1.1 文档目的.....	7
1.2 参考资料.....	7
1.3 术语与缩写词.....	7
第 2 章 产品概述.....	9
2.1 产品说明.....	9
2.2 产品运行.....	10
2.3 产品结构.....	11
2.4 功能特色.....	12
第 3 章 软件安装、配置、运行方法.....	13
3.1 软件安装.....	13
3.2 软件配置.....	14
第 4 章 建模工具.....	17
4.1 流程管理.....	17
4.1.1 流程管理主界面（流程设计器）.....	17
4.1.2 启动节点配置界面.....	19
4.1.3 交互节点配置界面.....	22
4.1.3 查看节点配置界面.....	24
4.1.4 控制节点配置界面.....	25
4.1.5 子流程节点配置界面.....	26
4.1.6 结束节点.....	27
4.1.7 流程模版导入导出.....	28
第 5 章 工作流引擎.....	29
5.1 基本功能.....	29
5.2 任务节点类型.....	29
5.2.1 启动（开始）节点.....	29
5.2.2 结束节点.....	30
5.2.3 交互节点.....	30
5.2.4 子流程节点.....	30

5.2.5 控制节点.....	30
5.2.6 查看节点.....	30
5.3 流程引擎（WorkflowEngine）	31
5.3.1 任务处理器配置规则.....	32
5.3.2 任务认领.....	34
5.3.2 任务放弃认领.....	35
5.3.3 任务指派.....	36
5.3.4 任务授权.....	36
5.3.5 创建流程实例（WORKFLOWINSTANCE）	37
5.3.6 创建任务实例（WORKTASKINSTANCE）	38
第 6 章 业务协同平台.....	39
6.1 WinForm 部分.....	40
6.1.1 日常业务.....	40
6.1.2 未认领任务.....	42
6.1.3 待办任务.....	43
6.1.4 已完成的任务.....	48
6.1.5 流程监控.....	50
6.1.6 我参与的任务.....	51
6.1.7 用户表单管理.....	52
6.2 WebForm 部分.....	57
6.2.1 日常业务.....	57
6.2.2 未认领任务.....	59
6.2.3 待办任务.....	60
6.2.4 已完成任务.....	66
6.2.5 流程监控.....	68
6.2.6 我参与的任务.....	69
6.2.7 用户表单管理.....	70
第 7 章 开发业务流程.....	74
7.1 创建流程图.....	74
7.2 制作业务表单.....	75

7.3 Web 业务表单.....	75
7.3.1 表单开发步骤.....	75
7.3.2 表单界面开发.....	76
7.3.3 表单开发重点说明.....	77
7.3.4 表单继承的基类.....	77
7.4 WinForm 业务表单.....	78
第 8 章 业务流程实例.....	80
Email: 406590790@qq.com.....	81
QQ : 406590790.....	81

第 1 章 引言

RDIFramework.NET 工作流程组件是以 RDIFramework.NET 框架为支撑, 根据我们多年的项目经验和项目实践, 结合国内各大 workflow 产品的特点研发的一套流程管理组件。该组件不仅考虑到从零搭建业务系统, 也考虑到与现有业务系统的整合。从零搭建系统我们可以使用 RDIFramework.NET 框架为基础, 来快速搭建业务系统。与现有系统的整合, 我们的流程引擎提供了丰富的功能接口, 供三方业务系统调用, 并提供组件的全部源码方便用户进行整合。RDIFramework.NET 工作流程组件采用 SOA 架构模式, 流程引擎通过 WCF 方式访问。支持 B/S、C/S 系统, 同时支持 SQLServer、ORACLE 等主流数据库。

随着计算机软件应用的普及, 信息化系统发挥的作用也越来越大, 企业信息化建设的不断深入, 对系统功能和自动化程度要求越来越高。客户要求系统功能与实际的工作情景紧密结合, 对每个业务环节的控制要求越来越精确。如何让我们的信息化系统更加贴近客户需求, 满足客户不断变化的业务流程成了我们软件开发商不得不面对的问题。

信息化建设初期我们的信息化系统担当的是记录信息、查询信息、统计信息的功能, 孤立的管理信息, 各功能之间缺少联系, 也无法做到各环节的精确控制, 很多情况只能靠人为判断来完成。随着信息化程度的加深, 这些功能远远不够。我们实际工作中的每一个环节并不是孤立的, 每个环节前后都有千丝万缕的联系, 要记录这些环节, 做到精确控制, 所以我们引入了 workflow 的概念, 推出了 workflow 组件。

1.1 文档目的

本文档为《RDIFramework.NET workflow 组件》产品说明书。

编写本产品说明书的目的是充分叙述本组件所能实现的功能及其运行环境, 以便使用者了解本组件的使用范围和使用方法, 并为软件的维护和更新提供必要的信息。

1.2 参考资料

1.3 术语与缩写词

- 1) 工作流: 把日常工作中的各个环节连接起来, 让工作按照制定的方向进行, 这就是工作流, 实现这样功能的系统就叫工作流系统。
- 2) 任务: 工作流中的每一个工作环节就叫任务, 每个任务在流程图中以节点的方式展现。
- 3) 模版: 一个固定的被其他对象参照的数据, 在 oop 思想中, 与类的定义相似, 模版可以理解为一个类的声明。

- 4) 实例：一个参照模版创建的数据，在 oop 思想中，与类的实例相似，实例可以理解为一个类的实例。
- 5) 流程模板：用来描述工作过程的流程图，以类似地图的形式展现，这个流程图定义了一个工作的任务、流向和流转控制信息。
- 6) 流程实例：依附于流程模板的一个运行实例，表示一个实实在在的工作。
- 7) 任务模板：流程模板上的任务节点叫任务模板。
- 8) 任务实例：流程实例中的任务节点叫任务实例。
- 9) 处理实例：一个任务实例可能经过指派、回退等多次处理，每一次处理即一个处理实例。一个处理实例是由一个任务实例和一个处理者（对应具体的处理人）实例组成的。
- 10) 处理者：是指可以处理任务的一个类型，它指的是一个人的集合，例如部门、组织机构、岗位、角色或个人。
- 11) 处理人：是指可以处理任务的具体的个人，一般用工号来表示。
- 12) 处理者实例：处理任务实例的人或者组织（即处理者），统称为处理者实例。一个任务实例有一个或者多个处理者实例与之对应。处理者实例就是真正要处理任务的人的集合。
- 13) 流程数据：指流程按照流程图流转时由引擎创建的数据，例如流程实例、任务实例、处理者实例。
- 14) 业务数据：指由客户表单创建的数据，业务数据在流程图上流转从而实现日常工作。

第 2 章 产品概述

2.1 产品说明

随着计算机软件应用的普及，信息化系统发挥的作用也越来越大，企业信息化建设的不断深入，对系统功能和自动化程度要求越来越高。客户要求系统功能与实际的工作情景紧密结合，对每个业务环节的控制要求越来越精确。如何让我们的信息化系统更加贴近客户需求，满足客户不断变化的业务流程成了我们软件开发商不得不面对的问题。

信息化建设初期我们的信息化系统担当的是记录信息、查询信息、统计信息的功能，孤立的管理信息，各功能之间缺少联系，也无法做到各环节的精确控制，很多情况只能靠人为判断来完成。随着信息化程度的加深，这些功能远远不够。我们实际工作中的每一个环节并不是孤立的，每个环节前后都有千丝万缕的联系，要记录这些环节，做到精确控制，所以我们引入了 workflow 的概念，推出了 workflow 系统。

组件主要包括三部分：1-建模工具，2-流程引擎，3-业务协同平台。

首先，流程引擎和业务数据是独立的，两者没有必然联系（流程引擎可以访问业务数据），这样设计的好处是引擎可以跟任何已有的系统整合，比如已经有一个进销存系统，现在要加入 workflow 对单据进行流程控制，只需要调用引擎提供的接口函数就可以。

其次，流程引擎是封闭的，对用户来说是透明的，用户也不需要知道内部是怎么工作的，只需要了解如何通过建模工具配置流程模板就可以，业务平台是可以任意扩展的，通过开发任意多的用户控件，业务平台只要动态加载这些业务控件就可以。

最后，建模工具采用 `winform` 的方式设计，是从操作的灵活性考虑的，`bs` 结构的系统操作上确实不灵活，速度也慢，再加上建模工具使用的人员有限，也没必要开发 `bs` 结构的。业务平台则不然，尽量使用 `bs` 结构设计，现在大部分系统是通过 `bs` 来实现的。

2.2 产品运行

RDIFramework.NET— workflow 组件采用目前主流的 C# 语言开发设计完成，运行环境需 .NET Framework 4.0 以上，使用 WINDOWS 操作系统。如果你的电脑没有安装 .NET Framework 4.0，可以在打开下面的链接进行下载安装。

<http://msdn.microsoft.com/zh-cn/netframework/aa569263.aspx>

本产品支持多种产品的数据库，如 Sql Server、Oracle、MySQL 等，可根据你的需要，选择适合的数据库数据产品后台数据库。系统默认优先采用微软的 Sql Server 2008。

2.3 产品结构

RDIFramework.NET—工作流组件系统结构如下图所示：

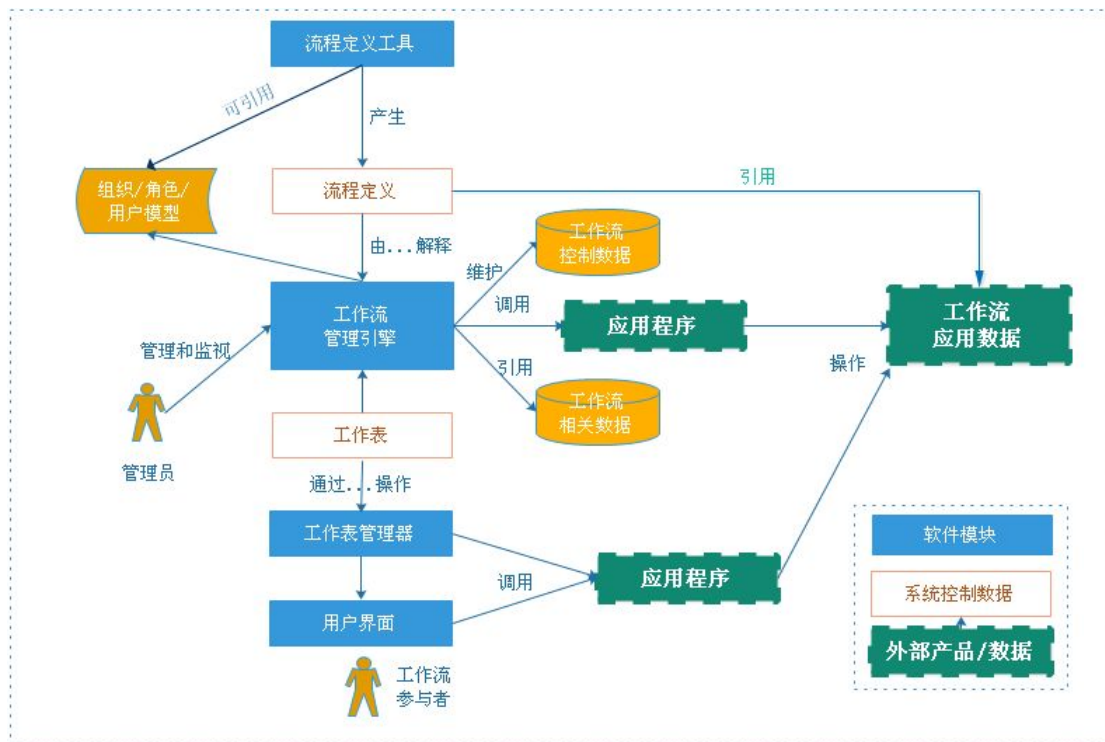


图 2.3.1 RDIFramework.NET—工作流组件系统结构

2.4 功能特色

RDIFramework.NET— workflow 组件具有以下特色：

- 1、与 RDIFramework.NET 框架完美结合，支持 B/S、C/S 系统，SQLServer、Oracle 等多数据库类型支持，同时可以使用框架本身提供的所有服务接口。
- 2、可定制的表单：在 BS 结构中，用户可自由开发 UserControl，并通过流程管理工具建立表单与业务平台的联系，表单即可在系统中流转。C/S 结构中，组件通过动态加载 DLL 或者 EXE 文件实现加载不同的功能模块，便于软件管理和升级。
- 3、分布式流程引擎：真正面向 SOA 架构（服务层可部署为 WCF、Windows Service、Web Service、Remote 等），提供标准的 WCF 接口调用，可以实现远程调用，分布式部署。丰富的开发接口，用户可通过在程序中调用接口实现流程启动、流转等。
- 4、流程引擎支持复杂业务流转：独立研发的流程引擎，考虑到国内用户的特点，支持复杂多变的流程流转，流程图配置简单，业务逻辑一目了然。组织机构、处理者和岗位分配贴近用户企业。
- 5、流程设计器：可视化的流程图制作工具，所见即所得的制作流程图，并可以把流程图导出到 XML 文件中，也可以从 XML 文件中导入流程。
- 6、与已有系统集成：可方便与已有系统集成，可通过调用流程引擎接口把流程功能集成到现有业务系统中，而用户原有的系统只需做少许改动即可。流程引擎全都是以接口的方式体现，具备 2 个特性：**业务无关性，程序结构无关性**。即适应任何业务系统，又适应 B/S 和 C/S 结构的程序，只要调用相应的接口方法即可。
- 7、即时消息：即时消息可以帮助在线和离线用户及时得到流程通知，便捷的实现协同办公。
- 8、流程统计图表：可以统计流程任务的处理情况，考察任务处理者的工作效率，实现绩效考核，并可以监控流程的其他状态方便决策。
- 9、编码简单易懂，适合常用设计模式，注释详细。提供全部的设计文档、源码实现、配套说明文档，可以在不同软件项目中反复使用，提高工作效率。提供完善的接口函数调用说明、开放接口、开放源码、开放数据库结构设计。完全支持最终用户的个性化 2 次开发实际需求，可以在不同软件项目中反复使用，提高工作效率。
- 10、其他。

第 3 章 软件安装、配置、运行方法

3.1 软件安装

一、工作流的安装。

1)、打开 SqlServer 企业管理器（确保已经安装了 Microsoft SQL Server 2008 及以上版本，若没有安装，请参考相应的安装方法）。

方法一：开始菜单→运行。输入：“SSMS”后确定。

方法二：开始菜单→程序→Microsoft SQL Server 2008→SQL Server Management Studio 确定即可。

2)、附加框架数据库。

步骤 1、右键单击数据库，选择“附加”，如下图。

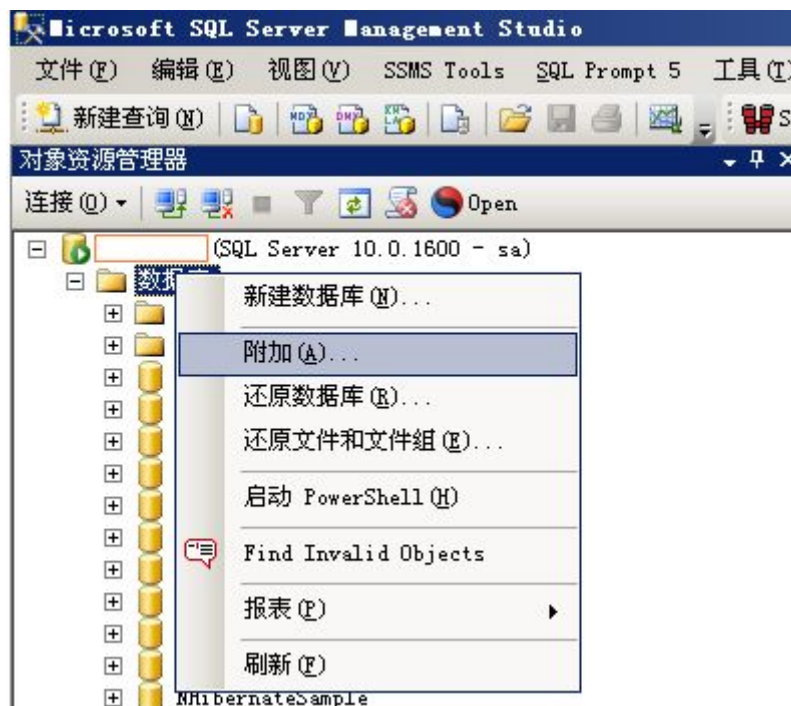


图 3.1.1 附加框架数据库

步骤 2、打开“附加数据库”对话框，单击“添加”按钮，找到框架 Db 目录下的“RDIFrameworkWfV2.8.mdf”后确定，即可附加框架数据库。另外需要说明的是，工作流组件是依托于 RDIFramework.NET 框架的，因此要先安装部署 RDIFramework.NET 框架，对于框架的安装部署方法可以参考框架的产品使用说明书。

以上为 SQLServer 版本数据库的安装，对于 Oracle 数据库的安装，直接通过 imp 命令导入我们提供给你的 ORACLE 备份文件即可。

二、框架的安装。

RDIFramework.NET (.NET 快速信息化系统开发整合框架) 无需安装即可直接运行，类似于绿色软件，略去了软件安装所带来的繁琐步骤。

3.2 软件配置

打开可执行文件所在的文件夹后，双击“RDIFramework.NET.exe”即可运行框架，如下图所示。

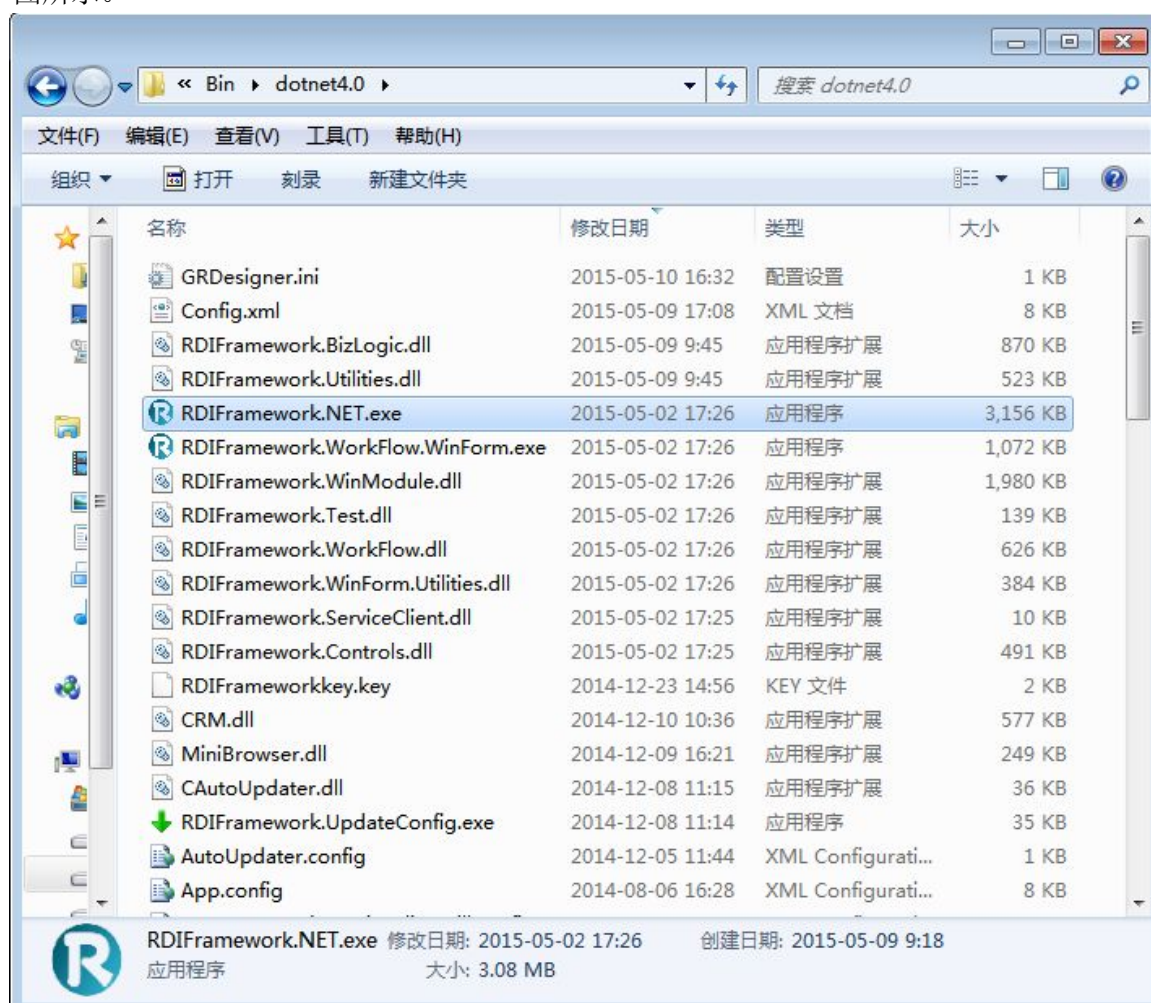


图 3.2.1 框架运行

说明：

框架第一次运行，需要配置框架数据库的连接字符串，配置方法为：

1. 在框架可执行文件 RDIFramework.NET.exe 所在文件夹（如图 3.2.1 所示）中找到框架配置文件 Config.xml，用记事本或其他文本编辑器打开 Config.xml。
2. 在 Config.xml 文件找到“数据库连接相关配置”部分，修改各数据库类型与数据库的连接字符串与你的数据库服务器相符即可，参考配置如下图所示：

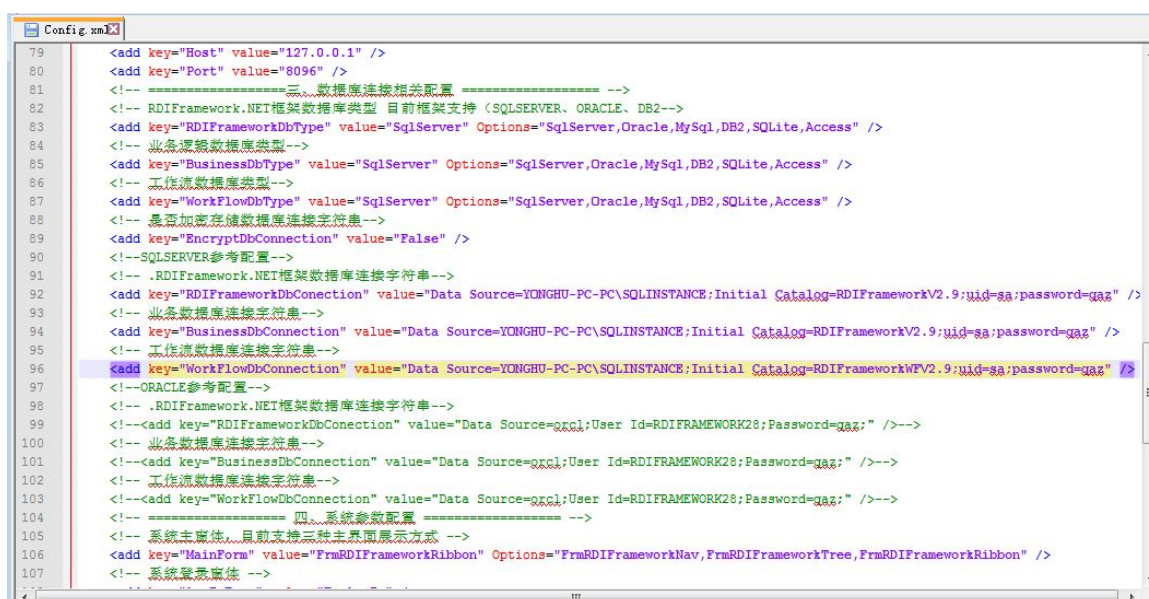


图 3.2.2 框架数据库 SQLServer 版连接配置

3. 把 Value 值修改为你的数据库连接相配置的字符串即可，此处为 SqlServer 数据库的连接字符串，对于框架数据库的类型是通过“RDIFrameworkDbType”来进行设置的，如上图所示，此处为“SqlServer”。当然，如果为 ORACLE 类型的数据库，则同步修改“RDIFrameworkDbType”为 Oracle，

“RDIFrameworkDbConection”为 Oracle 的连接字符串即可。如图：3.2.3 框架数据库 ORACLE 版连接配置。

```
<!-- RDIFramework.NET框架数据库类型 目前框架支持 (SQLSERVER、ORACLE、DB2)-->
<add key="RDIFrameworkDbType" value="SqlServer" Options="SqlServer,Oracle,MySql,DB2,SQLite,Access" />
<!-- 业务逻辑数据库类型-->
<add key="BusinessDbType" value="SqlServer" Options="SqlServer,Oracle,MySql,DB2,SQLite,Access" />
<!-- 工作流数据库类型-->
<add key="WorkFlowDbType" value="SqlServer" Options="SqlServer,Oracle,MySql,DB2,SQLite,Access" />
```

图 3.2.3 框架数据库 ORACLE 版连接配置

4. 下面给出常见配置的数据库连接字符串的配置。

SqlServer 2008 连接字符串设置如下：

- SQL Server 身份验证设置如下：

```
<addkey="RDIFrameworkDbConnection" value="Data Source=myServerAddress;Initial
Catalog=RDIFramework;uid=myUsername;password=myPassword" />
```

- Windows 身份验证设置如下：

```
<addkey=" RDIFrameworkDbConnection" value="Data Source= myServerAddress;Initial
Catalog= RDIFramework;Integrated Security=SSPI;" />
```

ORACLE 连接字符串设置如下：

- Using TNS 设置如下：

```
<addkey=" RDIFrameworkDbConnection" value="Data Source= RDIFramework;User
Id=myUsername;Password=myPassword;" />
```

- Using integrated security 设置如下：

```
<addkey=" RDIFrameworkDbConnection" value="Data Source= RDIFramework;Integrated
Security=SSPI;" />
```

- Using ODP.NET without tnsnames.ora 设置如下:

```
<addkey=" RDIFrameworkDbConnection" value="Data
Source=(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP) (HOST=MyHost) (PORT=MyPor
t))) (CONNECT_DATA=(SERVER=DEDICATED) (SERVICE_NAME=MyOracleSID))) ;User
Id=myUsername;Password=myPassword;" />
```

5. 对于其他类型的数据库连接字符串可通过下面的网址进行参考设置。

<http://www.connectionstrings.com/>

第 4 章 建模工具

workflow 组件最重要的就是建模工具，所有业务流程的核心都是从建模开始的。建模工具使用 C# 语言开发，采用 WinForm 的方式，完全整合到 RDIFramework.NET 框架中，拥有框架的一切特性。同时业务逻辑层的代码可以毫无修改的应用到 web 方式上，这也是充分考虑使用的灵活性。通过建模工具可以：管理组织机构，管理流程模板，管理权限，管理工作流程业务表单等。

4.1 流程管理

流程模板也可以叫流程模型，采用分类管理的方式，可以无限级分类，流程模板必须属于某一分类，流程模版分类管理如下图所示，流程图采用可视化的制作，可以通过拖拽图形的方式画流程图，图形化的操作一目了然。让软件系统真实的体现工作过程，成为企业经营和管理的重要工具，是使用 workflow 系统的主要目的。

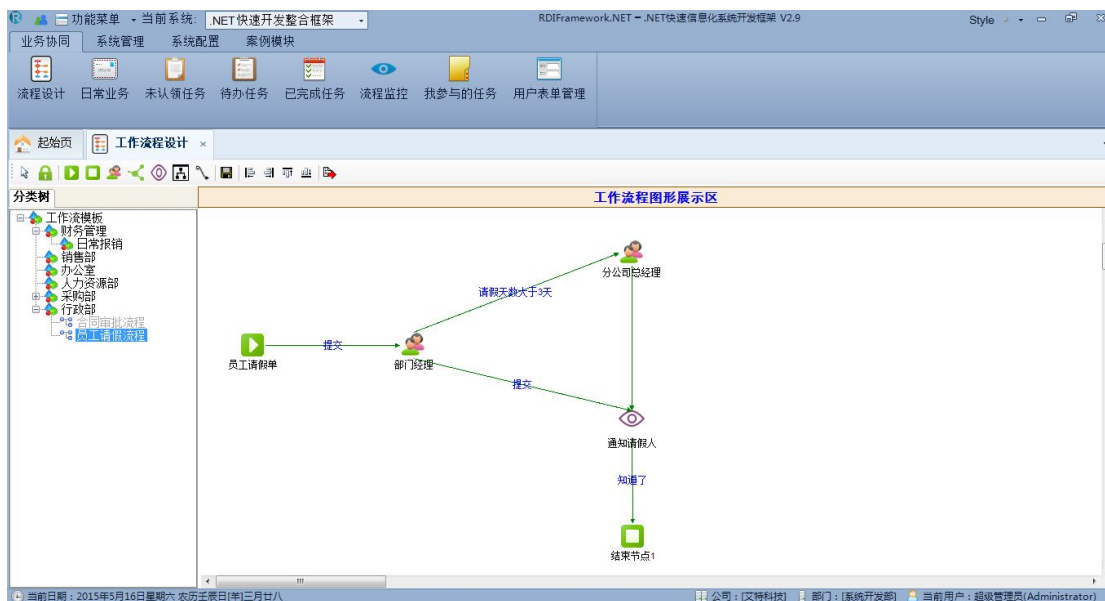


图：流程模版分类管理

4.1.1 流程管理主界面（流程设计器）

在开始之前，先介绍一下目前市场上流程设计器的几种形式，无非就是两种形式一种是 B/S 的，一种是 C/S，个人感觉这两种形式没必要苛求那一种，各有优缺点，B/S 的不用安装可直接使用，但操作灵活性比不上 C/S 的，B/S 的流程设计器目前有几种技术可以实现，一种是 js，一种是 Silverlight 或者 FLEX，还有一种是用 ActiveX 插件。流程设计器一般是管理员、实施人员使用，用户群体比较单一固定，但使用要求比较高，必须方便灵活，所以我们的流程设计器采用 C/S 的方式，结合框架提供的各种服务，绝对不压于 B/S 的流程设计器。

流程设计器是编辑流程模版的工具，可视化的流程设计器能直观的编辑流程模版，所见即所得，流程模版存放流程流转所需要的信息，使用流程设计器来管理流程模版是比较常见的方式，如下图所示是我们开发的流程设计器：



图：工作流程设计器

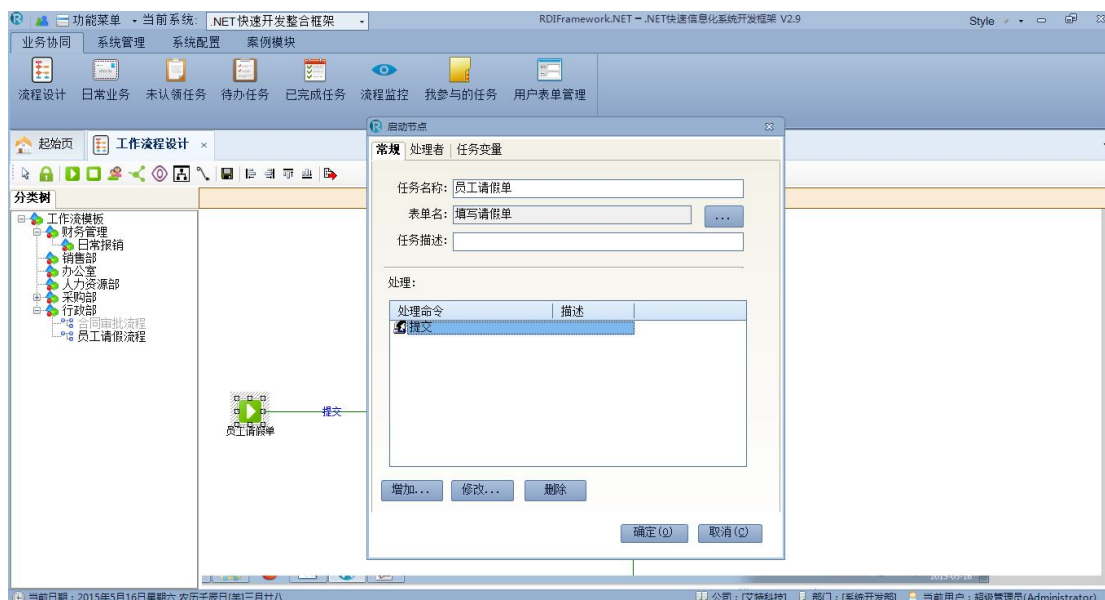
在工作流程设计器主界面我们可以进行流程设计，工作流组件提供了 6 种任务节点类型和一个连线配置，如下图所示，在后面我们会分别介绍。



4.1.2 启动节点配置界面

启动节点是一个业务流程的开始，所有业务都是从启动节点开始执行的。流程定义必须包含一个以上启动节点，可以在该界面中配置交互表单上的命令按钮，有权限启动流程的处理者，流程中用到的变量，对应的表单等。启动节点的配置界面如下图所示：

1、启动节点-常规配置页

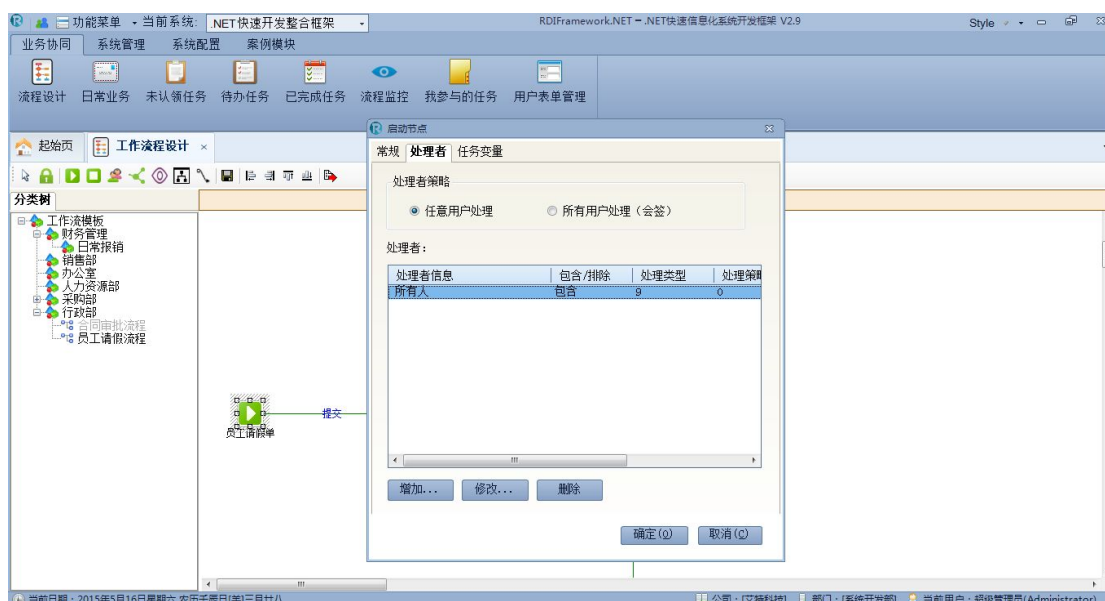


图：启动节点-常规配置页

配置项说明：

- ❖ 任务名称：启动节点的名称。
- ❖ 表单名：启动节点要加载的主表单列表。
- ❖ 任务描述：启动节点的描述信息。
- ❖ 处理命令：移交到下一任务节点时的处理命令，一般默认“提交”诸如此类的字眼，用于做为业务操作界面上按钮的展示名称。

2、启动节点-处理者配置页



图：启动节点-处理者配置页

配置项说明：

- ❖ 处理者策略：启动节点的名称。
 - 任意用户处理：所有用户共享任务。
 - 所有用户处理（会签）：所有用户处理任务。
- ❖ 增加处理者

增加处理者主要是增加当前任务可以处理的用户、组织机构、角色或岗位等。增加处理者界面如下图所示：

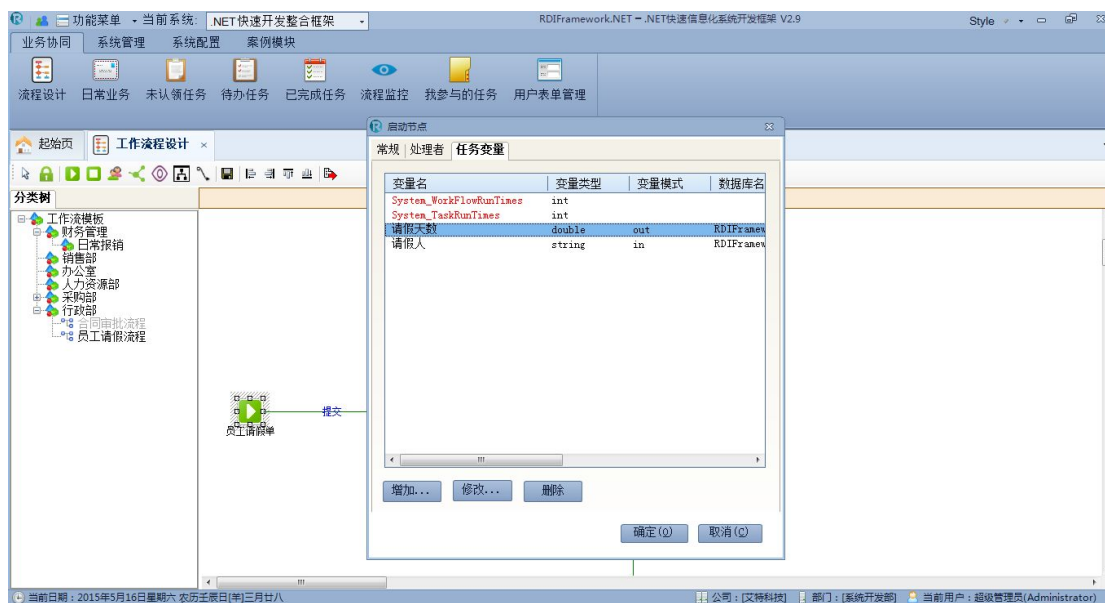


图：增加处理者

对于处理者详细配置规则可以参见章节“任务处理者配置规则”。

3、启动节点-任务变量配置页

任务变量配置页可以为当前任务节点增加变量，以备在业务流程执行过程中做相应的判断需求。

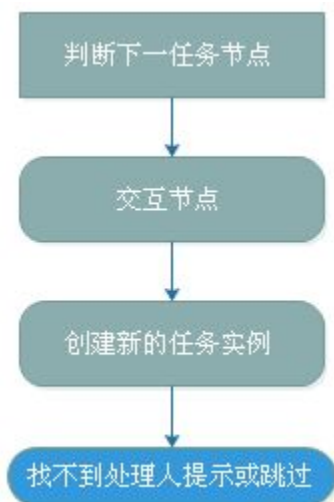


图：启动节点-任务变量配置页

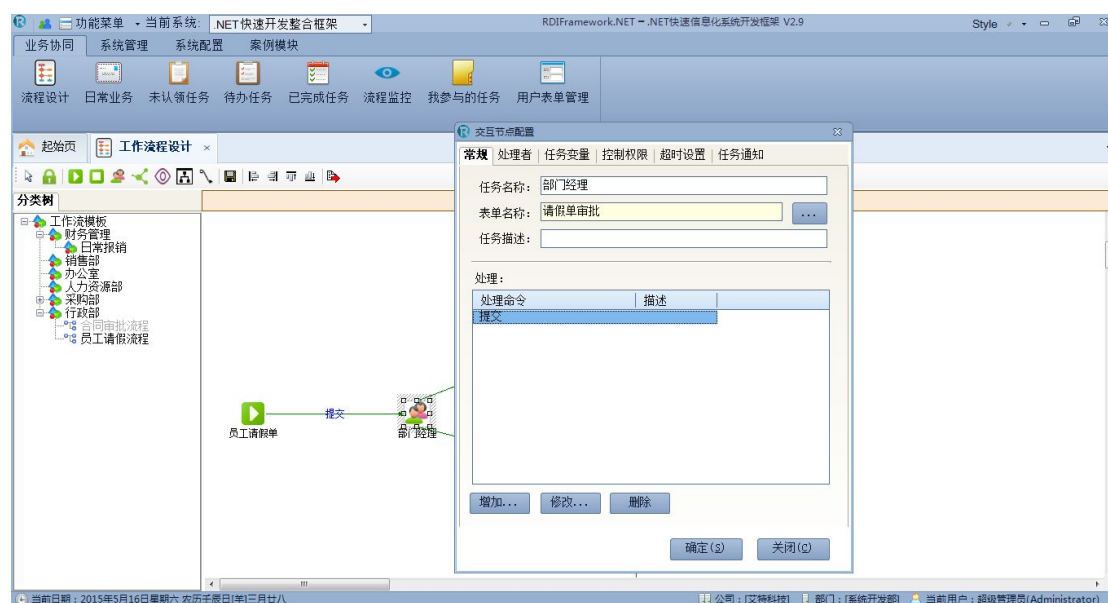
4.1.3 交互节点配置界面

与用户交互的节点，该节点可以配置业务表单供用户操作。还可以创建流程变量和任务变量，供流程引擎使用，变量是引擎与业务平台交互的途径。

交互节点执行流程如下：



交互节点配置主界面如下图所示：

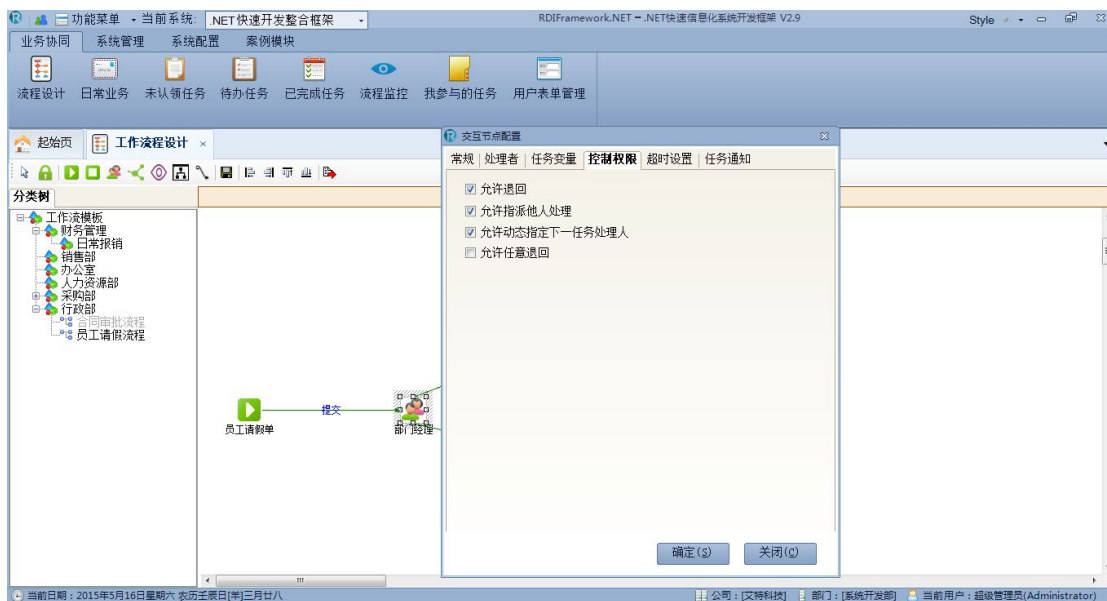


图：交互节点配置主界面

交互节点的“常规”、“处理者”、“任务变量”配置页与启动节点类似，可以参考“启动节点”部分的说明。

1、交互节点-控制权限配置页

控制权限配置页主要用于配置当前任务节点是否允许“退回”、“指派他人处理”、“动态指定下一任务处理人”、“任意退回”。控制权限配置页如下图所示：

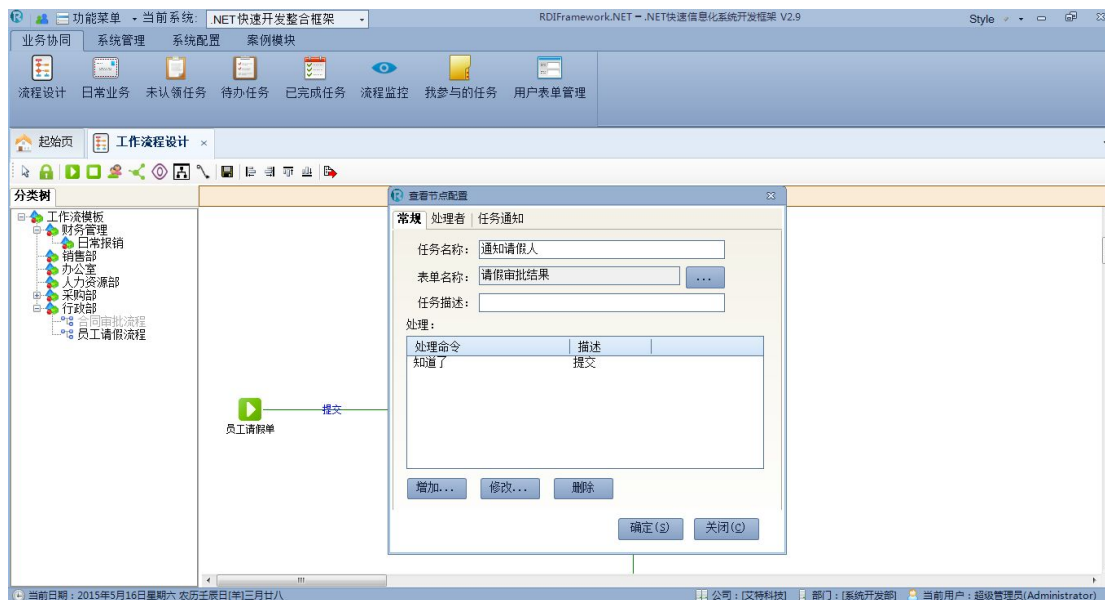


图：交互节点配置-控制界面

- ❖ 允许退回：当前任务节点提交后，在待处理用户未认领的情况下是否允许退回修改。
- ❖ 允许指派他人处理：当前任务处理者在特殊情况下可以指派他人来处理自己的任务。
- ❖ 允许动态指定下一任务处理人：在流程运行的时候，需要有操作人去选择下一步任务的执行人，而不是在流程定义的时候先固定好的执行人。
- ❖ 允许任意退回：常有这样一种情况，流程进行了很多步了，到后面时审批人员发现问题，可以指定退回到那一步重新开发流转。

4.1.3 查看节点配置界面

该节点只有处理者和任务通知，一般是用来通知任务审批情况的，供用户查阅信息的交互节点，该节点可以配置业务表单，供用户查看，可以看做是交互节点的简化版。查看节点配置界面如下图所示：

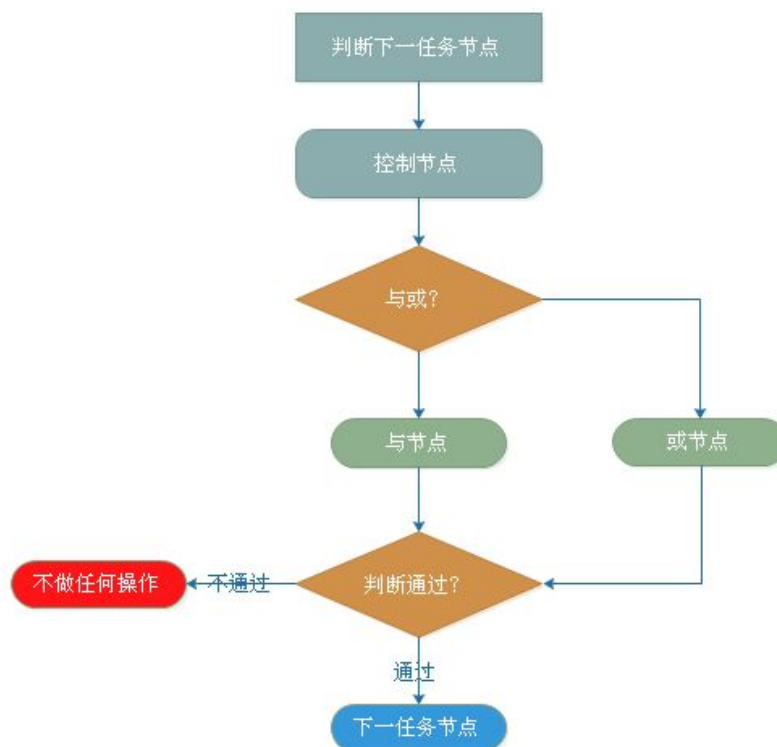


图：查看节点配置界面

4.1.4 控制节点配置界面

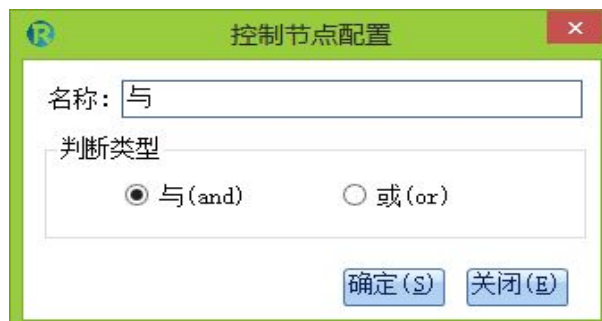
控制节点是控制流程流转的节点，使用多种组合的控制节点可以实现流程的合流、分流。控制流程是否继续流转，控制节点有两种类型:and 和 or，任何节点后面都可以放置一个控制节点，and 表示只有前面节点的实例全部完成才能继续流转，or 表示只要前面节点的实例有一个通过就可以继续流转。使用控制节点我们就可以很好的处理会签等复杂的情况。

控制节点处理流程如下：



与节点判断是否通过，首先寻找与节点前面的所有节点，判断每个节点的任务实例是否通过，只要有一个未通过则该与节点不通过。与节点通过后也创建一个实例。与控制节点相连的实例不允许后退，也不允许抓回。

控制节点配置界面如下图所示：



图：控制节点配置界面

4.1.5 子流程节点配置界面

子流程节点是可以把任何一个已有的流程作为子流程，实现流程的重用，减少工作量。

子流程节点配置界面如下图所示：



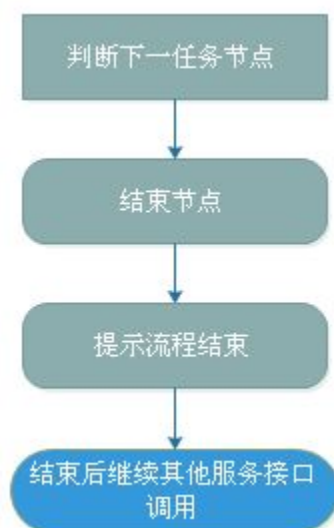
该图显示了一个名为“修改子流程配置”的对话框。对话框顶部有一个绿色的标题栏，左侧有一个带有“R”图标的按钮，右侧有一个红色的关闭按钮。对话框主体分为两个部分：上方是一个名为“常规”的选项卡，下方是配置区域。配置区域包含三个输入项：1. “子流程名称”：文本框内显示“员工请假流程”，右侧有一个蓝色的省略号按钮。2. “启动节点”：下拉菜单，当前显示“员工请假单”，右侧有一个向下的箭头。3. “任务描述”：一个黄色的文本框，目前为空。在配置区域下方，有两个蓝色的按钮，分别是“确定(S)”和“关闭(E)”。

图：子流程节点配置界面

4.1.6 结束节点

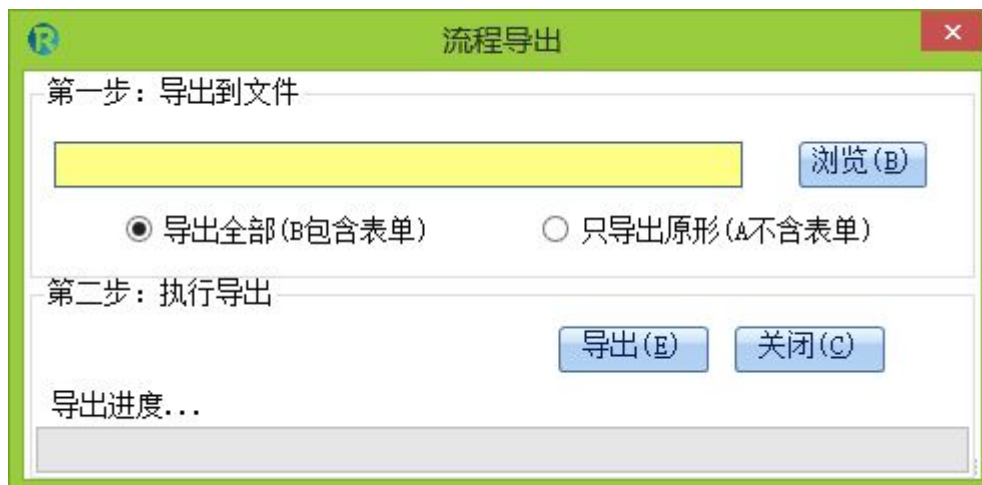
表示当前流程结束，没有配置界面。结束分两种情况，一种是流程全部结束，一种是退出当前流程，进入主流程。

结束节点执行流程如下：



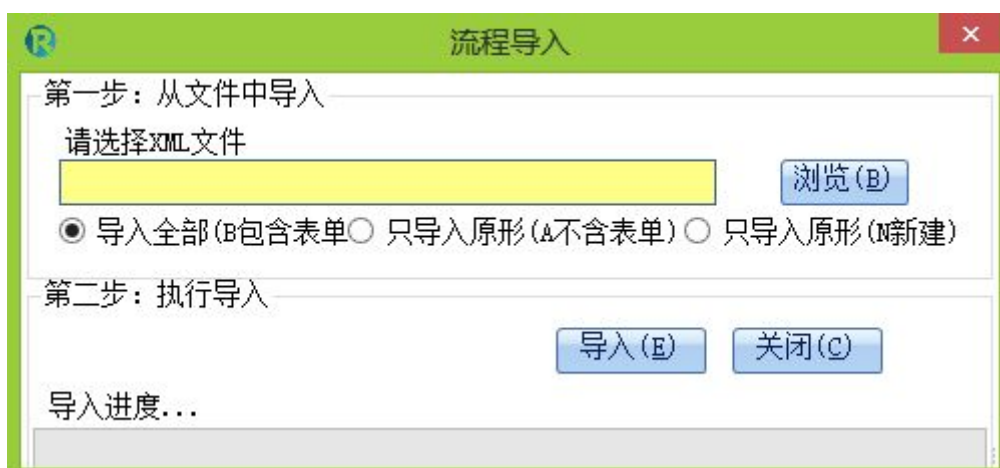
4.1.7 流程模版导入导出

在流程分类树中，点击鼠标右键，我们可以对所选流程模版执行导入或导出操作。流程模版可以导出为 xml 文件。如下图导出界面：



分两种情况导出，一种是包含表单配置信息，一种是不包含表单配置信息。一般情况都要包含表单配置信息。

导入流程模版界面如下图：



从 Xml 文件中导入流程模版，导入的时候分三种情况，包含表单配置信息，不包含表单配置信息，这两种情况必须保证要导入的流程模版不存在，否则无法导入。第三种情况是只导入原型，如果流程模版已经存在，那么会新建一个同名的流程模版。

第 5 章 工作流引擎

设计非常灵活的引擎，满足任意复杂的业务需求，使最终用户通过修改流程图，配置业务表单来完成流程再造，不需要修改程序代码，流程引擎能按照新的模板流转。流程引擎设置 6 种类型的任务节点，基本可以完成任何情况的流传，例如，会签，审批，根据业务条件流转等，流程引擎独立设计，提供对外接口，可以与任意系统整合。


5.1 基本功能

- 顺序流转：像发邮件一样，工作流转到下一处理人,处理人像处理邮件一样查看任务和任务。
- 条件流转：根据业务数据，满足不同条件流转不同的处理人。例如采购金额大于等于 10 万的采购单需流转至总经理，小于 10 万的采购单需流转至分管经理。
- 任务回退：在领导审批时不满意处理结果，需要退回前一任务处理人，让他重新处理后再提交。
- 任务会签：对某些工作需要多位领导都签字后才能执行的任务，设定该任务会签的领导，所涉及到的领导都审批后，该任务才能继续向下流转。
- 任务合流：多个任务流向一个任务的汇总流程
- 任务超时提醒：任务超过设定时间时向任务执行者或者指定人员进行提醒。
- 任务授权：可以授权某人暂行自己的业务执行权利。
- 任务指派：任务流转到自己时，我们可以指派某人执行此任务。
- 支持共享任务处理：多人都有权处理的任务，都能看到该任务，别其中一人处理后，其他人不会再看到该任务。
- 支持召回：在下一任务未处理前可召回。
- 支持子流程(流程嵌套)。
- 支持多启动节点：在流程环节基本相同只有开始任务不同时，可以利用多个开始节点并免多创建一个流程。
- 简化任务节点类型实现复杂流转：RDIFramework.NET 工作流组件的任务节点类型是经过深入的归纳和抽象，类型少功能强，流程图更贴近实际的业务流程，没有多余的节点，一目了然，所见即所得。
- 任务到达通知：任务达到时向任务执行者或者指定人员进行提醒。

5.2 任务节点类型

工作流模板是由连线和任务节点组成的，任务节点配置相关的信息供引擎使用，连线表示流转方向，连线上可以配置流转条件。


5.2.1 启动（开始）节点

启动（开始）节点图标为：

一个流程可以有多个启动节点，这是与其它流程系统所不同的地方，这样设计的是适应启动节点所挂接的表单不同，但后面的任务环节相同的业务流程。例如一个合


同审批流程，包括 2 个合同：维保合同和代理合同，这两个合同启动节点不同但后面的审批环节相同，这种情况通过设置多个启动节点就可以避免创建两个流程。

5.2.2 结束节点

结束节点图标为：


表示当前流程结束。结束分两种情况，一种是流程全部结束，一种是退出当前流程，进入主流程。

5.2.3 交互节点

交互节点图标为：


交互节点是需要用户交互操作的节点，在交互节点上配置的表单，通过表单进行人机交互。

5.2.4 子流程节点

子流程节点图标为：


子流程节点是可以把任何一个已有的流程作为子流程，实现流程的重用。

5.2.5 控制节点

控制节点图标为：

控制节点是控制流程流转的节点，使用多种组合的控制节点可以实现流程的合流、分流。

5.2.6 查看节点

查看节点图标为：

查看节点是只具有任务查看权限的节点，不需要做数据处理，最常用的是审批结果通知发起人。

5.3 流程引擎 (WorkflowEngine)

工作流引擎，是 workflow 组件的核心部件，主要功能是流程调度和冲突检测，流程定制是通过工作流引擎来完成的。

RDIFramework.NET workflow 组件是一款自主研发的工作流引擎，它使用最流行的 .NET 语言编写，具有可视化的配置工具和完善的业务展现平台，能够适应任何复杂情况流转。RDIFramework.NET workflow 组件不但具有流程引擎一般的功能，还可以很方便的嵌入到您现有的业务系统中，不影响现有的业务系统的代码结构，是您在现有业务系统中增加流程功能的最佳选择。

RDIFramework.NET workflow 引擎的基本功能：

- ❖ 顺序流转
- ❖ 条件流转
- ❖ 分流
- ❖ 合流
- ❖ 指派
- ❖ 授权
- ❖ 回退
- ❖ 异常终止
- ❖ 流程挂起
- ❖ 召回(在下一任务未处理前可召回)
- ❖ 子流程(流程嵌套)

RDIFramework.NET workflow 引擎独有特色：

1、支持共享任务处理

多人都有权处理的任务，都能看到该任务，别其中一人处理后，其他人不会再看到该任务。

2、支持多启动节点

在流程环节基本相同只有开始任务不同时，可以利用多个开始节点并免多创建一个流程。

3、简化任务节点类型实现复杂流转

RDIFramework.NET workflow 的任务节点类型是经过深入的归纳和抽象，类型少功能强，流程图更贴近实际的业务流程，没有多余的节点，一目了然，所见即所得。

5.3.1 任务处理者配置规则

处理者类型		
标示	处理者类型	说明
1	流程启动者	启动流程的人
2	某一任务实际执行者	实际执行某一任务的人，某一任务的执行人不一定是模板中配置的人，例如任务实例被指派，或其他人被授权。
3	指定人员	指定某人处理
4	部门	指定某一部门处理
5	角色	指定某一角色处理
6	岗位	指定某一岗位来处理，必须是具体的岗位，例如人事部经理，财务主管等。
7	从变量中获取	动态指定处理者，对无法知道该任务处理者时使用，只能从变量中获取某个人。
8	某一任务选择的处理者	在某一任务中指定一批处理人，把任务发给从这些人中选定的的人。例如分发信息的业务，如同发电子邮件一样。
9	所有人	所有人都可以处理该任务，一般在启动节点使用所有人。
10	指派	被指派（此类型隐藏，有程序处理）
11	授权	被授权(此类型隐藏，有程序处理)

说明：开始节点只支持 3，4，5，6，9

与处理者的关系和筛选		
标示	关系名	说明
1	本部门正领导	只适用于处理者为 1，2，3，7，8。
2	本部门副领导	
3	上级部门正领导	
4	上级部门副领导	
5	所在机构	
6	上级机构	
7	所属角色	
8	排除	只能排除单个人

配置处理人注意事项：

- 1、每一个任务的处理者尽量独立，减少与其他任务的联系。
- 2、任务的处理者尽量是直接的处理者而不是经过运算获得的。
- 3、考虑如果该任务被指派时是否影响后续处理。
- 4、考虑处理人被授权后是否影响后续处理。
- 5、支持在流程提交时动态指定处理人。

处理者实例创建方式：每一个任务节点提交时，流程引擎根据模板配置自动生成下一任务处理者的事例，并在 OPERATORINSTANCE 表中为每一个处理者创建一个事例(即一条数据记录)。

5.3.2 任务认领

名称		任务认领: TaskClaim
功能描述		
认领任务设计	条件	status=' 1' //表示任务实例未处理 Operstatus=' 0' //表示处理实例未执行
	动作	✓ 修改任务实例表: WORKTASKINSTANCE 表: status=2,//标识任务实例为认领状态 OperatorInsId=@OperatorInstanceId//记录被该处理者认领
		✓ 修改处理者实例表: OPERATORINSTANCE 表: userId=@userId,//认领任务的用户 operstatus='3'//标识处理者实例为认领状态
参考		

5.3.2 Task Abandon Claim

名称		任务放弃认领：TaskUnClaim
功能描述		放弃认领的任务，任务重新回到未认领任务中，其他人就可以认领该任务（共享的任务），是认领任务的逆过程。
任务放弃设计	条件	Status='2',//表示任务实例为认领状态，但没处理 Operstatus='3' or operstatus='2 //表示处理者实例为认领状态，但没处理
	动作	✓ 修改任务实例表 WORKTASKINSTANCE 表： status=1,//标识任务实例为未处理状态 operatorInsId=null//清除处理者实例
		✓ 修改处理者实例表 OPERATORINSTANCE 表： userId=null,//清除认领人信息 operstatus='0'//标识处理者实例为未处理状态
参考		

5.3.3 任务指派

名称		任务指派: TaskAssign
功能描述		<p>任务在需要指派给别人处理时使用指派功能, 指派规则:</p> <ol style="list-style-type: none"> 1、只有认领的任务才能指派, 通过指派任务就会到达被指派人的未认领任务里, 原处理者实例标记为已指派某人处理。 2、被指派人如果不接受被指派的任务, 可以放弃认领任务。 3、只能指派给具体的一个人, 指派后原处理人还可以继续处理该任务。 <p>被指派过的任务, 可以再次指派。</p>
任务指派设计	条件	已认领的任务 operstatus='3'
	动作	在原任务实例的基础上, 再复制新的处理人实例, 创建一个新的处理者实例。
参考		

5.3.4 任务授权

名称		任务授权: TaskAccredit
功能描述		<p>当因出差等因素造成处理者不能处理流程时, 为不影响正常流转需要把流程和任务的操作权指定给某一一人。</p>
任务授权设计	授权规则	<ol style="list-style-type: none"> 1、任务授权可分为两种情况: <ol style="list-style-type: none"> A: 启动任务, 被授权人可以发起被授权的流程; B: 处理任务, 被授权人可以处理被授权流程的任务。 2、授权后, 已经在授权人“未认领任务”和“已认领任务”中任务还有授权人处理, 新创建的任务才有被授权人处理, 授权取消后, 被授权人已经认领的任务还继续处理, 新的任务不在到达被授权人。
参考		

5.3.5 创建流程实例（WORKFLOWINSTANCE）

用户在流程模板上启动一个新的流程时即创建一个新的流程实例，用户可以输入流程实例的名称，超时时间等信息。只有创建了流程实例才能创建流程实例的每个任务节点。在创建流程实例的同时自动创建一个开始的任务实例。

- 1、创建 WORKFLOWINSTANCE 表内容
- 2、创建 WORKTASKINSTANCE 表内容

5.3.6 创建任务实例（WORKTASKINSTANCE）

一个处理实例包括两部分：任务实例和处理者实例。

(1)、创建任务实例

根据符合条件的分支找到符合条件的下一任务模板节点，根据任务模板配置的处理者创建任务实例，每个处理者都要创建一类实例，并根据处理者策略决定每个处理者的实例化方式和实例数。

处理者策略：

A：所有处理者同时处理

该处理策略，需要为每个处理者类型中的所有人都创建一个任务实例和处理人实例。例如，处理者类型是部门，使用“所有处理者同时处理”的处理者策略的时候，需要为部门下每个人生成任务实例和处理人实例。

B：任意处理者处理

该处理策略，生成一个任务实例和多个处理者实例，即多个处理者共同处理一个任务，只要有一个处理者（谁处理都可以，不需要每个处理者）处理了该任务，流程就可以继续向下流转。例如，处理者是某部门，只要该部门下任何一个人处理该任务就可以继续流转。

(2)、创建处理者实例

处理者的选择可以通过两个途径来完成，一是从流程模板中获取，每个任务节点中都配置了处理者。二是，用户指定处理人，提交时动态指定处理人，处理人的范围可以通过流程模板配置。

每个任务实例对应一个或者多个处理者，由处理者策略来决定（见创建任务实例）。

注意：在获取处理者时需要判断该处理者对应的处理人是否已经授权，如果已授权，创建实例的时候应该创建被授权人的实例。

第 6 章 业务协同平台

业务协同平台是流程系统与用户交互的工作平台，所有的业务表单都通过平台展现，用户在该平台上处理日常业务（启动任务、处理任务、查看参与的任务，查看流程状态）。用户不必关心流程如何流转，只要处理好自己的业务表单即可，任务即会根据条件自动流转到下一个处理者。

业务平台同时支持 WinForm 与 WebForm 两种方式。该业务平台的工作过程遵循以下规则：

一、系统登录，每个用户使用自己的帐号登录。

二、“我的任务”是用户已经处理和未处理的业务，用户必须首先从“未认领任务”中领取任务，领取后的任务会进入“已认领任务”中，已认领的任务才能被处理。

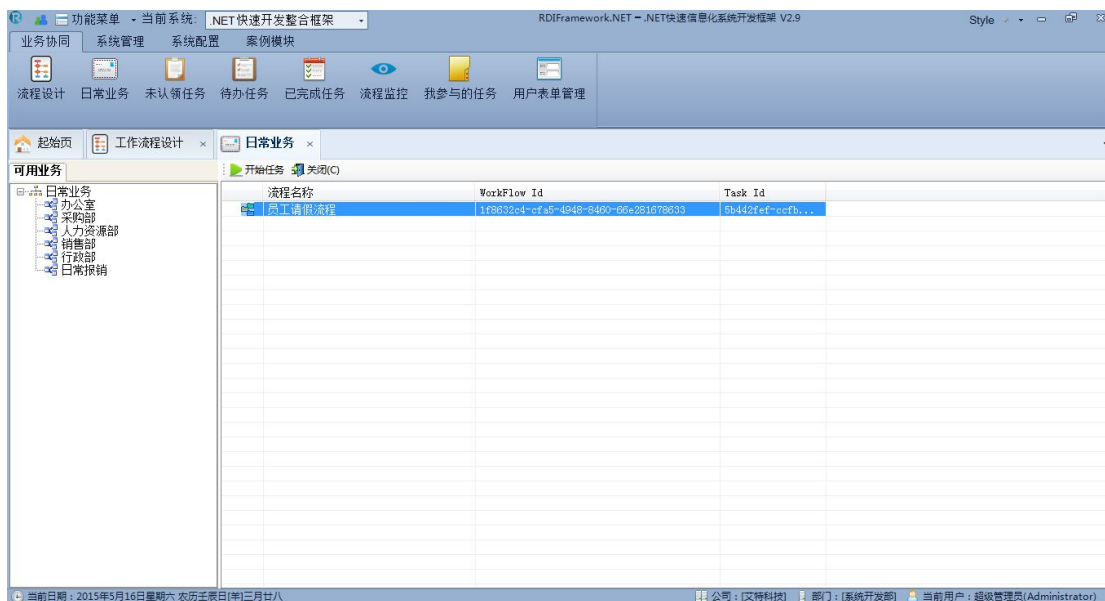
三、“启动任务”也叫启动流程，根据流程模板中的启动节点，来选择启动的流程，流程会根据启动节点来开始任务。

四、查看流程状态，可以在“我的任务中”，点“流程图”来查看流程路线图和已经走过的流程节点。

6.1 WinForm 部分

6.1.1 日常业务

“日常业务”模块主要展示了当前登录用户可以启动的业务流程列表。流程引擎会自动根据当前用户可以启动的业务流程自动加载流程列表到“可用业务”树中。“日常业务”模块主界面如下图所示：



图：日常业务主界面

在“可用业务”列表选择一个分类，在右侧选择我们需要启动的业务流程，点击“开始任务”，即可启动一个任务，如下图所示，我们启动了“员工请假流程”。

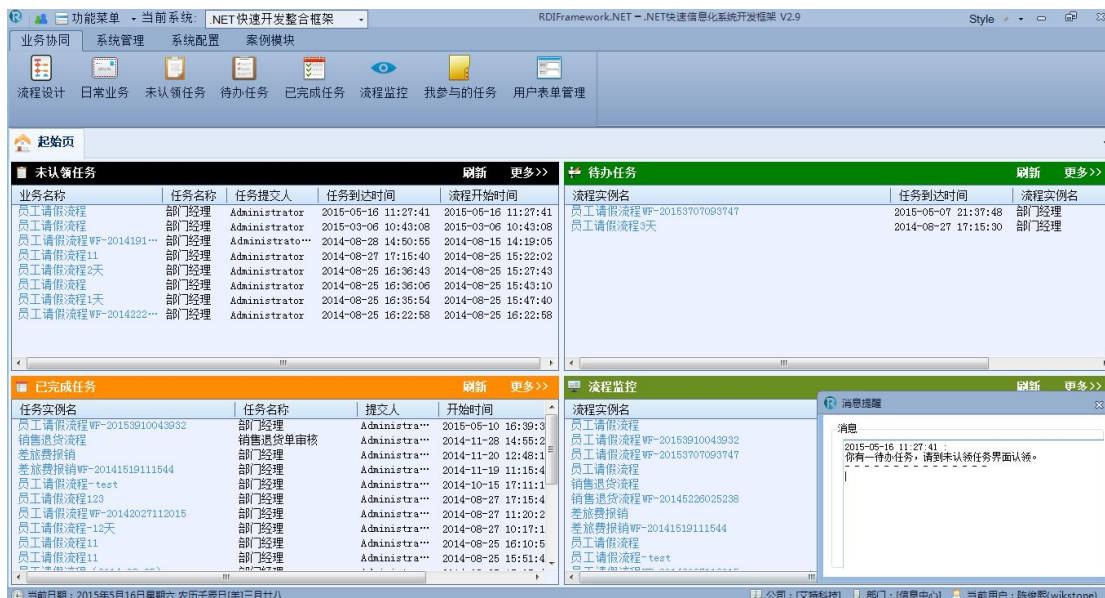


图：启动任务界面

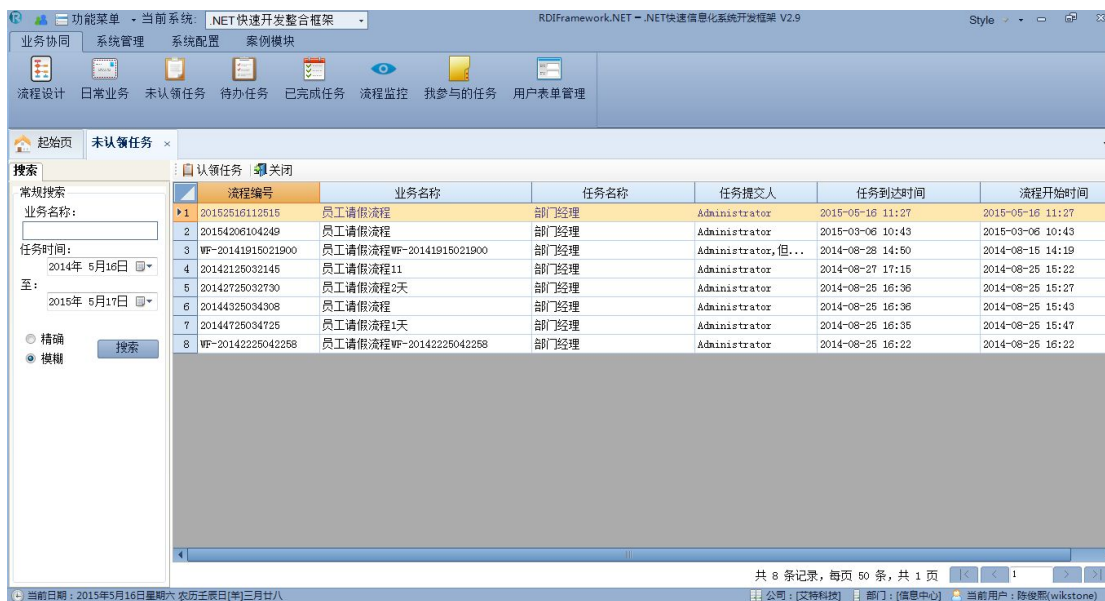
在“启动任务”主界面，表单列表会显示主表单中所有的子表单列表，以供用户输入各表单必要信息。

6.1.2 未认领任务

处理任务前，首先要认领任务。认领任务到“未认领任务”主界面进行认领，如下图所示。选择待认领的任务后点击“认领任务”就可以对所选任务进行认领，认领的任务就进入了“待办任务”列表。同时我们也可以按业务名称以及任务时间进行精确或模糊查询。刚刚的请假申请提交到了“wikstone”用户处，我们登录框架，右下角自动弹出消息提示，同时也可以通过主界面的“示认领任务”区域查看。



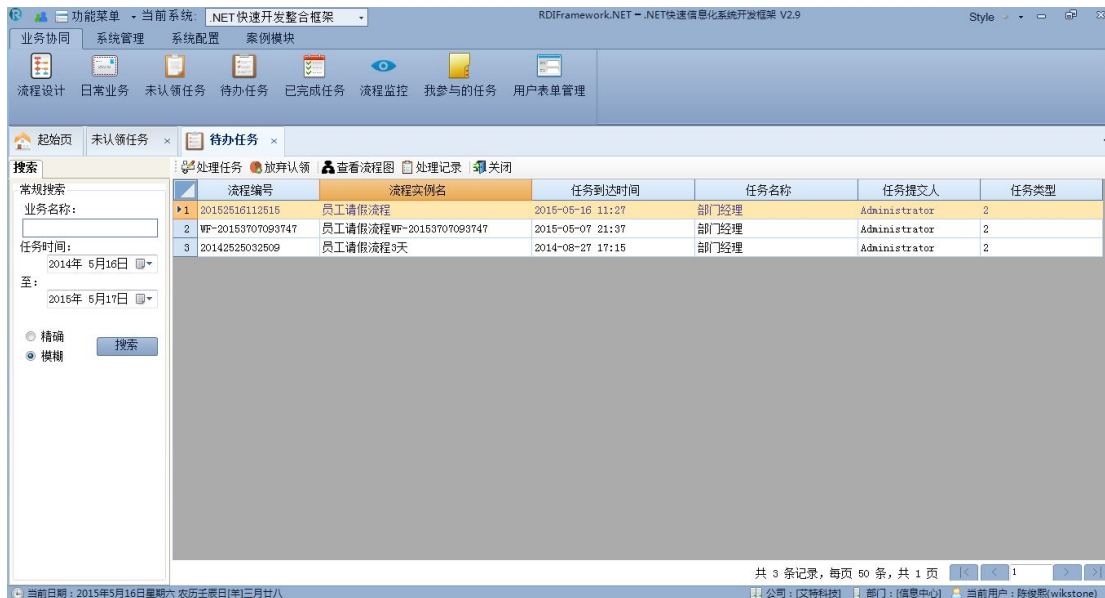
单击主界面“未认领任务”区域的“更多>>”按钮或者单击“业务协同”->“未认领任务”模块，即可打开未认领任务界面，如下图所示：



图：未认领任务界面

6.1.3 待办任务

“待办任务”模块是我们真正处理业务的地方，已认领的任务会进入到“待办任务”列表中，在这儿我们可以按条件搜索待办任务、处理所选任务、放弃认领所选任务、查看所选任务的流程图、查看所选任务的处理记录。待办任务主界面如下图所示：



图：待办任务主界面

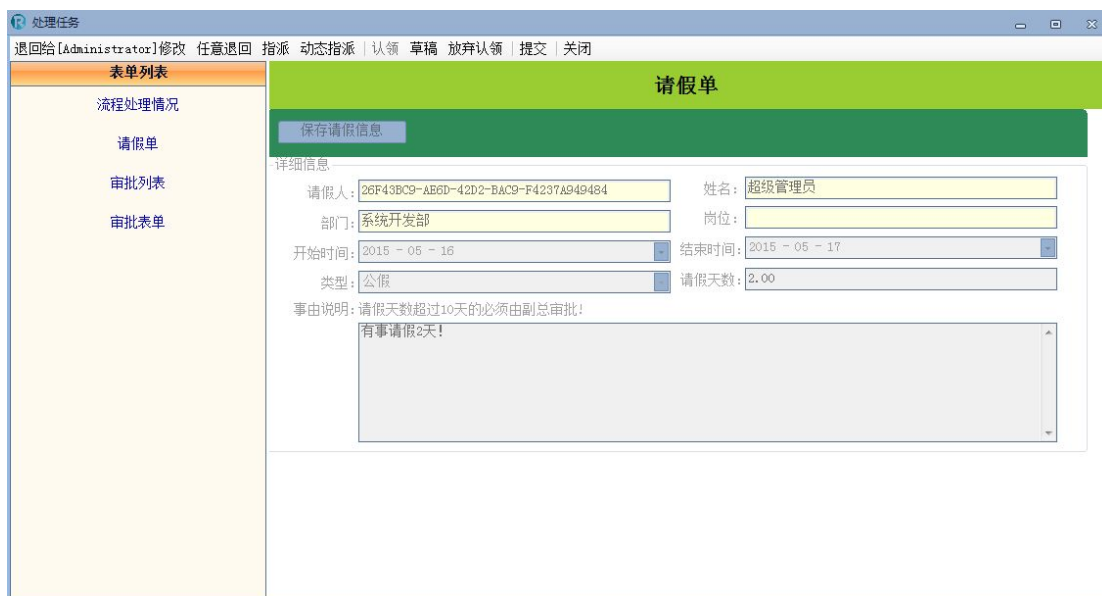
1、处理任务

在“待办任务”窗口，选择一个任务后点击“处理任务”，即可打开“处理任务”主界面，如下图所示，“处理任务”界面上方的命令按钮都是根据当前任务节点的设置动态生成的。



图：处理任务界面-审批表单

同时我们可以查看业务表单，单击左侧的请假单，如下图所示，这儿审批人员是不能修改业务表单信息的，这个都是可以通过流程设计时进行配置的。



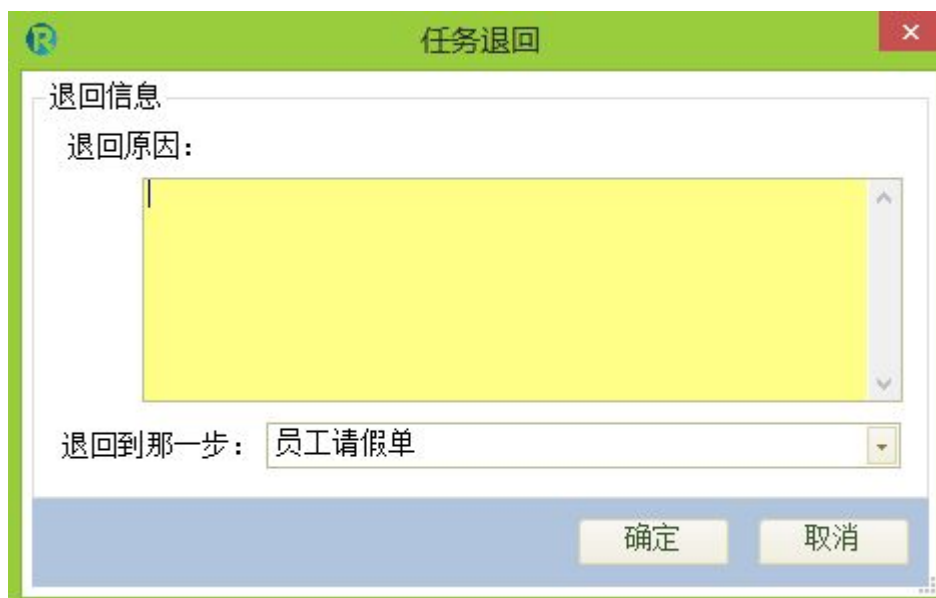
图：处理任务界面-查看业务表单

◆ “退回给...修改”

“退回给...修改”一般是“退回上一步”或者是退回给前一任务的提交者。一般用于发现业务流程不符审批要求，需要上一任务处理者重新处理时，可以单击此按钮。

◆ 任意退回

“任意退回”可以指定退回到已经执行过的某一任务节点，如下图所示，需要填写退回原因。



图：任务退回

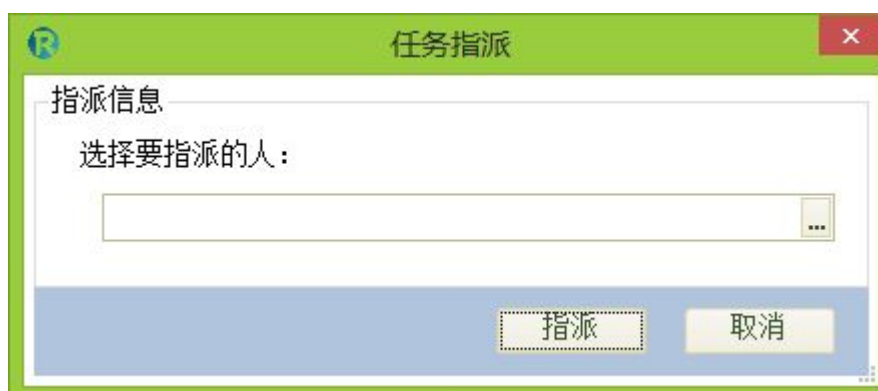
◆ 指派

任务在需要指派给别人处理时使用指派功能，指派规则：

1、只有认领的任务才能指派，通过指派任务就会到达被指派人的未认领任务里，原处理者实例标记为已指派某人处理。

- 2、被指派人如果不接受被指派的任务，可以放弃认领任务。
- 3、只能指派给具体的一个人，指派后原处理人还可以继续处理该任务。
- 4、被指派过的任务，可以再次指派。

点击“指派”命令按钮，即可打开“任务指派”窗口，如下图所示：



图：任务指派

在“选择要指派的人”文本框右边的“...”按钮即可打开用户选择界面，选择要指派的人即可。用户选择界面如下图所示：



图：用户选择

- ◆ 认领
可以在此界面“认领”任务。
- ◆ 草稿
“草稿”命令按钮主要功能就是先不提交到下一任务，只是对当前各表单的填写，审批意思等进行保存，以后再进行提交。
- ◆ 放弃认领

放弃认领的任务，任务重新回到未认领任务中，其他人就可以认领该任务（共享的任务），是认领任务的逆过程。

◆ 提交

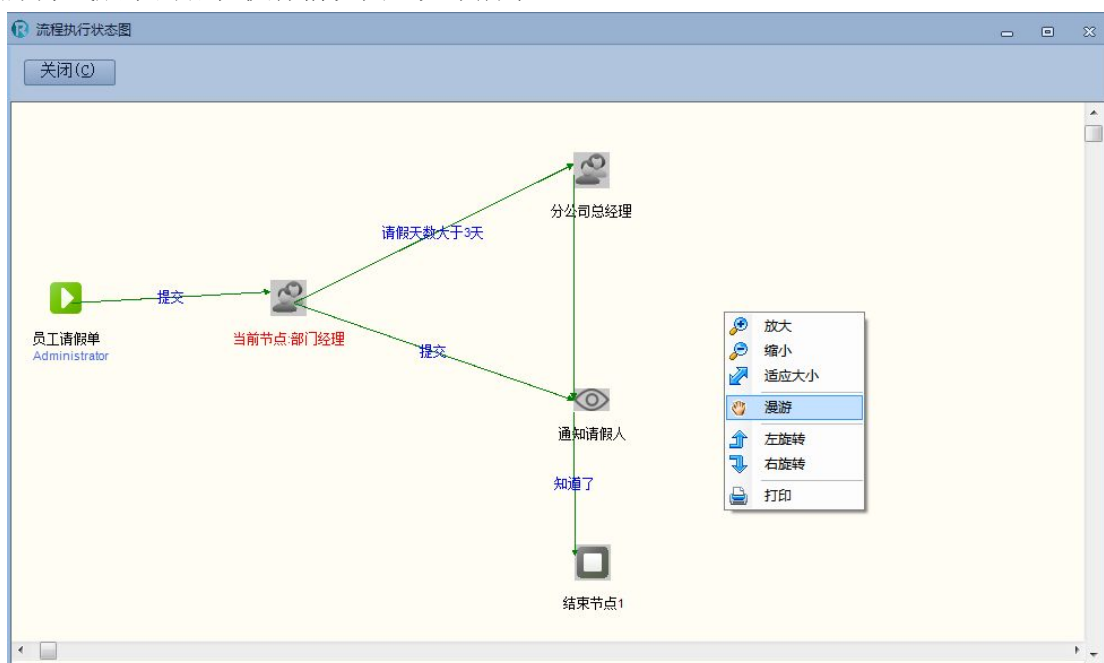
点击“提交”命令按钮，流程引擎即可根据当前流程的定义进行自动流转到下一任务节点。同时保存当前任务节点的业务数据，业务数据中可能含有流程需要的信息，所以在提交时先保存业务信息。业务数据和流程数据的设计尽量分开，业务信息可以经过复杂的处理。业务信息提交成功后就可以进行流程流转信息处理，如果失败根据实际情况判断是否删除已保存的业务数据，提供删除业务数据的接口。

2、放弃认领

在“待办任务”主界面，可以对已认领的任务放弃认领，放弃认领后任务进入未认领任务列表。

3、查看流程图

在“待办任务”主界面，选择一个任务，点击“查看流程图”，即可查看所选任务以图形方式展示的流程执行情况图，如下所示：



图：流程执行状态图

在流程执行状态图上，彩色表示已经走过的流程节点，灰色表示未走过的流程节点，通过流程图可以清晰的看到流程所处位置和后面的路线。

4、处理记录

在“待办任务”主界面，选择一个任务，点击“处理记录”，即可查看所选任务的所处理记录，如下所示：



	流程名称	任务名称	任务处理者	处理时间	处理者ID
1	员工请假流程	员工请假单	Administrator	2015-05-16 11:27:41	76cee5ba-4d00-4b1...
2	员工请假流程	部门经理	wikstone	2015-05-16 11:43:04	e4d1f73d-4c11-4b3...

关闭(C)

图：流程处理记录

6.1.4 已完成的业务

任务处理完成，进入到“已完成任务”列表，如下图所示。在已完成任务模块可以按条件查询已完成的业务，查看所选任务、查看任务流程图等。

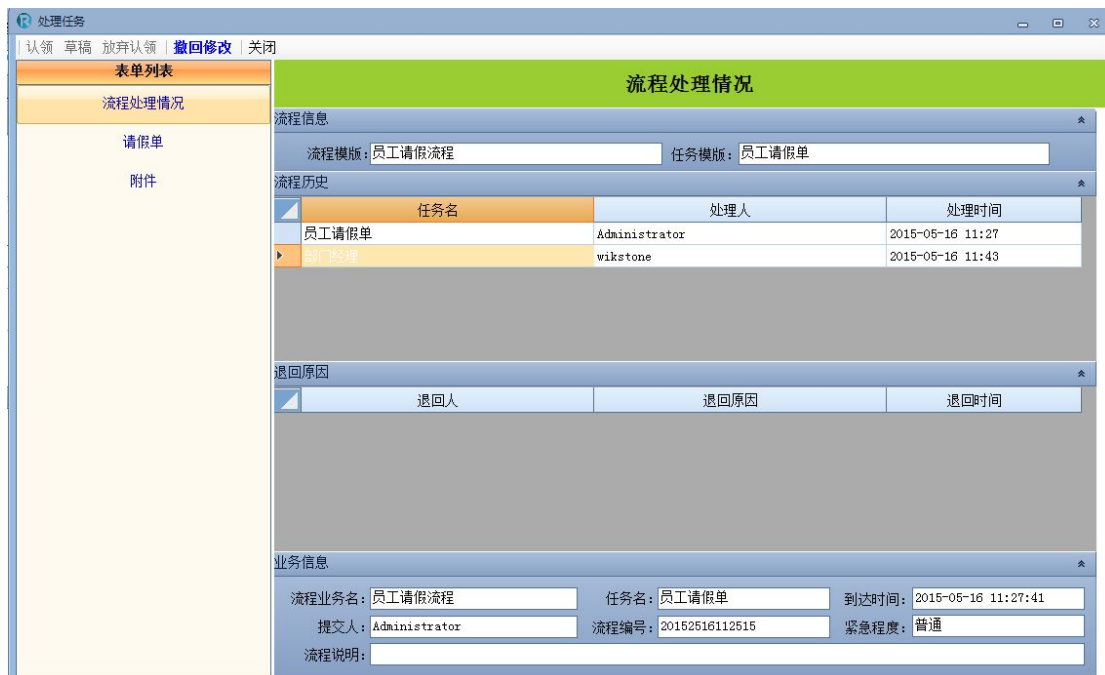


图：已完成任务主界面

1、查看任务

在“已完成任务”列表中选择任务，点击“查看任务”按钮，可以查看所选任务的详细信息，包含任务的业务表单列表、流程处理情况、审批列表、审批表单等。

选择“流程处理情况”可以查看流程的处理信息，如下图所示：



图：已完成任务-流程处理情况

在上面的窗口中,对于已提交的任务,在待处理用户未认领的情况下,可以单击“撤回修改”按钮对已完成任务重新处理。

选择“请假单”可以查看业务表单,如下图所示:

请假单

保存请假信息

详细信息

请假人: 26F43BC9-AB6D-42D2-BAC9-F4237A949484 姓名: 超级管理员

部门: 系统开发部 岗位:

开始时间: 2015 - 05 - 16 结束时间: 2015 - 05 - 17

类型: 公假 请假天数: 2.00

事由说明: 请假天数超过10天的必须由副总审批!

有事请假2天!

图: 已完成任务-请假单表单

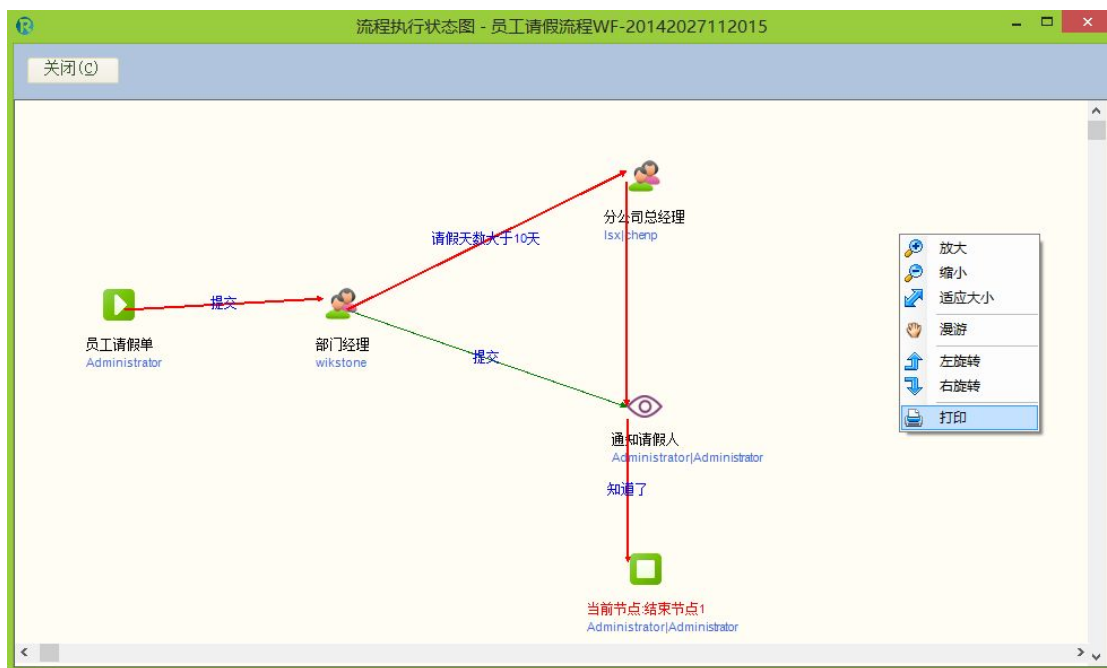
选择“审批列表”可以查看整个审批情况,如下图所示:

审批时间	审批人	部门	职务	审批结果	操作
2014-08-27 11:22	陈鹏	灵通部		同意	同意!
2014-08-27 11:22	刘书仙	总裁办公室		同意	同意!
2014-08-27 11:21	陈俊熙	信息中心		同意	同意, 请

图: 已完成任务-审批列表

2、查看流程图

在“已完成任务”主界面,选择一个任务,点击“查看流程图”,即可查看所选任务以图形方式展示的流程执行情况图,如下所示:



图：已完成任务-流程执行状态图

在上图中，我们可通过快捷菜单对流程图执行相应的操作，如：放大、缩小、漫游、打印等。

6.1.5 流程监控

流程监控模块主要用于监控流程的执行情况，界面上通过不同颜色显示流程的几种状态，如：还未执行、正在办理、正常结束。在流程监控主界面，我们可以组合条件搜索业务流程，同时还可以查看所选业务流程的状态以及处理记录。流程监控主界面如下图所示：

选择	流程编号	流程实例名	开始时间	结束时间	当前状态
<input checked="" type="checkbox"/>	20150516112215	员工请假流程	2015-05-16 11:22:15		正在办理
<input checked="" type="checkbox"/>	WF-20150516043932	员工请假流程WF-20150516043932	2015-05-16 16:39:32		正在办理
<input checked="" type="checkbox"/>	WF-20150516093747	员工请假流程WF-20150516093747	2015-05-16 21:07:48		正在办理
<input checked="" type="checkbox"/>	20150516044240	员工请假流程	2015-05-16 10:43:08		正在办理
<input checked="" type="checkbox"/>	20140528025512	销售退货流程	2014-11-28 14:55:13		正在办理
<input checked="" type="checkbox"/>	WF-20140528025512	销售退货流程WF-20140528025512	2014-11-28 14:55:13	2014-11-28 14:55:13	正常结束
<input checked="" type="checkbox"/>	20140528025512	销售退货流程	2014-11-28 14:55:13	2014-11-28 14:55:13	正常结束
<input checked="" type="checkbox"/>	WF-20141119111844	差旅费报销WF-20141119111844	2014-11-19 11:18:44		正在办理
<input checked="" type="checkbox"/>	20140503033516	员工请假流程	2014-11-03 14:05:28		正在办理
<input checked="" type="checkbox"/>	20140503033516	员工请假流程-1天	2014-10-15 17:00:42	2014-10-15 17:12:50	正常结束
<input checked="" type="checkbox"/>	WF-20142027112015	员工请假流程WF-20142027112015	2014-08-27 11:20:20	2014-08-27 11:30:35	正常结束
<input checked="" type="checkbox"/>	20140503033516	员工请假流程-1天	2014-08-27 10:17:17	2014-08-27 10:17:17	正常结束
<input checked="" type="checkbox"/>	WF-20140228042258	员工请假流程WF-20140228042258	2014-08-25 10:22:58		正在办理
<input checked="" type="checkbox"/>	20140503033516	员工请假流程1天	2014-08-25 15:51:45	2014-08-25 16:12:47	正常结束
<input checked="" type="checkbox"/>	20140503033516	员工请假流程1天	2014-08-25 15:47:40		正在办理
<input checked="" type="checkbox"/>	20140503033516	员工请假流程1天	2014-08-25 15:43:10		正在办理
<input checked="" type="checkbox"/>	20140503033516	员工请假流程1天	2014-08-25 15:41:06		正在办理

图：流程监控

6.1.6 我参与的任务

我参与的任务主界面如下图所示：

常规搜索
业务名称：
任务时间：
2014年11月16日
至：
2015年 6月17日
精确
模糊
搜索

流程编号	业务名称	任务名称	任务提交人	优先级	状态	任务到达时间	
1	20152516112515	员工请假流程	通知请假人	wikestone	普通	执行中	2015-05-16 11:43
2	20152516112515	员工请假流程	员工请假单	Administrator	普通	执行完毕	2015-05-16 11:27
3	WF-20153910043932	员工请假流程WF-20153910043932	通知请假人	wikestone	普通	执行中	2015-05-10 16:40
4	WF-20153910043932	员工请假流程WF-20153910043932	员工请假单	Administrator	普通	执行完毕	2015-05-10 16:39
5	WF-20153910043932	员工请假流程WF-20153910043932	员工请假单	Administrator	普通	执行完毕	2015-05-07 21:37
6	20154206104249	员工请假流程	员工请假单	Administrator	普通	执行完毕	2015-03-06 10:43
7	20145528025512	销售退货流程	销售退货单	Administrator	紧急	执行完毕	2014-11-28 14:55
8	20144220124228	差旅费报销	通知报销人	cs	紧急	执行完毕	2014-11-20 12:58
9	20144220124228	差旅费报销	填写差旅报销单	Administrator	紧急	执行完毕	2014-11-20 12:48
10	WF-20141519111544	差旅费报销WF-20141519111544	通知报销人	cs	普通	执行中	2014-11-19 11:25
11	WF-20141519111544	差旅费报销WF-20141519111544	填写差旅报销单	Administrator	普通	执行完毕	2014-11-19 11:15

共 11 条记录, 每页 50 条, 共 1 页

当前日期: 2015年5月16日星期六 农历壬辰日(甲)三月廿八
公司: (艾特科技) 部门: (系统开发部) 当前用户: 超级管理员(Administrator)

图：我参与的任务

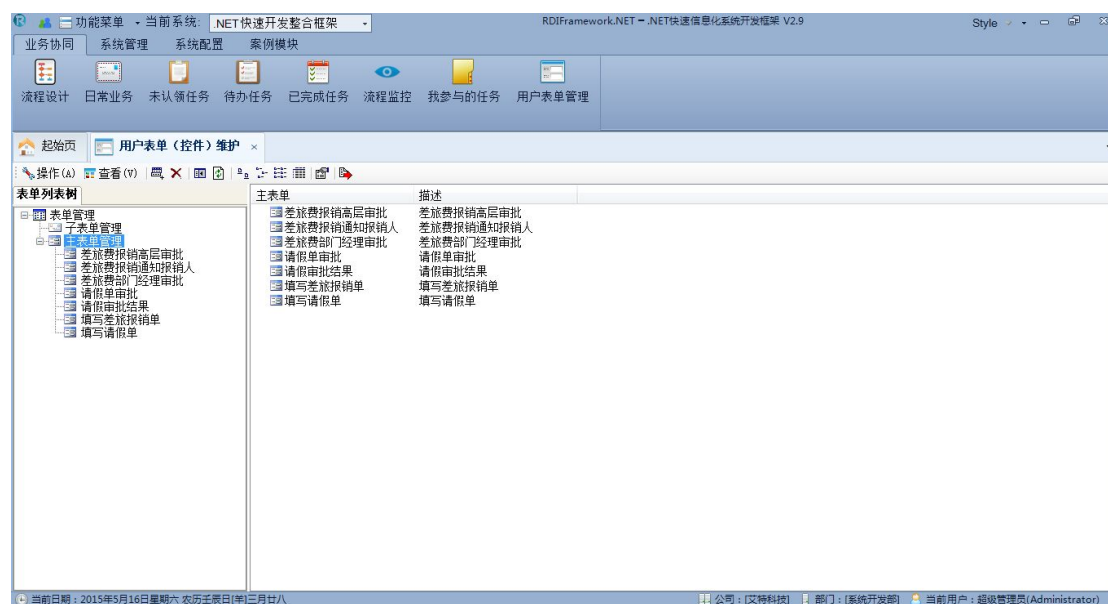
6.1.7 用户表单管理

表单是业务的展现形式，根据 workflow 组件的设计，表单采用用户控件 **UserControl** 来设计，表单有两种形式，主表单和子表单，主表单是由子表单组成的，一个主表单可以包含多个子表单，每个子表单就是一个 **UserControl**，任务节点中的交互节点可以挂接一个主表单，业务平台可以动态加载主表单里的 **UserControl**。

导航菜单中选择“业务协同”→“用户表单管理”，即可打开用户表单管理主界面。主表单与子表单的管理都在同一界面上展示。

1、主表单管理

在“表单列表树”中选择“主表单管理”即可打开主表单列表，我们可以增加主表单、修改主表单、删除主表单以及设置主子表单关联。



图：主表单管理主界面

◆ 增加主表单

主表单信息维护

主表单:

描述:

子表单:

子表单	所在程序集
-----	-------

查看
修改
新建

添加(A) 删除(D) 上移 下移

保存(S) 关闭(E)

图：增加主表单

◆ 修改主表单

主表单信息维护

主表单: 填写请假单

描述: 填写请假单

子表单:

子表单	所在程序集
请假单	RDIFramework.WorkFl
附件	RDIFramework.WorkFl

查看
修改
新建

添加(A) 删除(D) 上移 下移

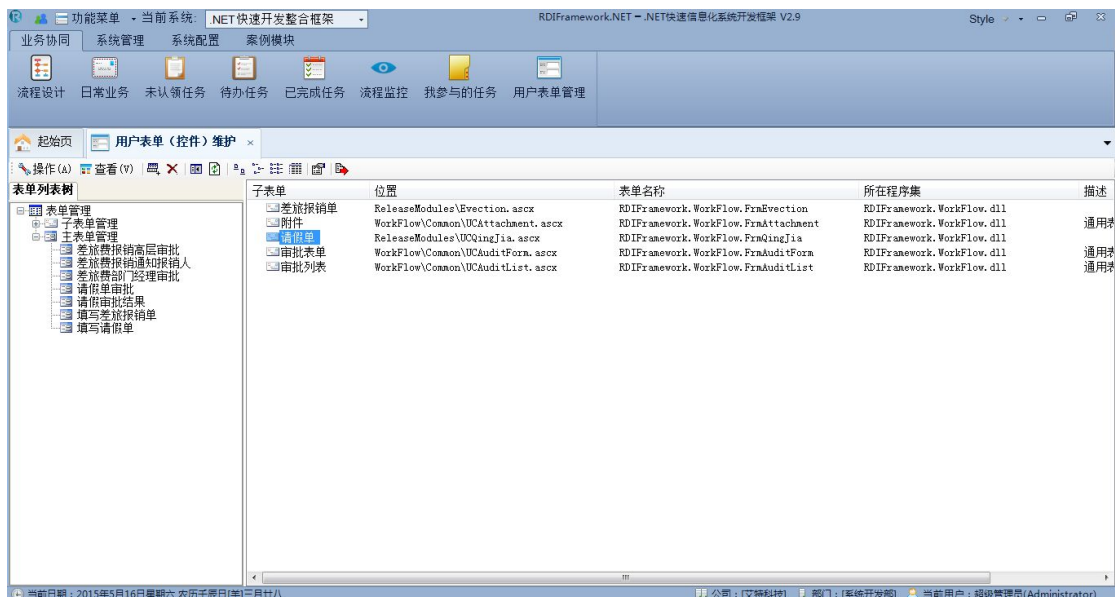
保存(S) 关闭(E)

图：修改主表单

◆ 删除主表单

2、子表单管理

在“表单列表树”中选择“子表单管理”即可打开子表单列表，我们可以增加子表单、修改子表单、删除子表单。



图：子表单管理主界面

◆ 增加子表单

子表单维护

基本信息

子表单名:

表单类型: ☒ WinForm ☐ Web ☐ WinForm/Web

位置: ...

控件ID:

表单名称:

所在程序集:

描述:

保存(S) 关闭(E)

图：增加子表单

◆ 修改子表单



子表单维护

常规 隶属于

子表单名: 请假单

表单类型: ☐ WinForm ☐ Web ☒ WinForm/Web

位置: ReleaseModules\UCQingJia.ascx ...

控件ID: UCQingJia.ascx

表单名称: RDIFramework.WorkFlow.FrmQingJia

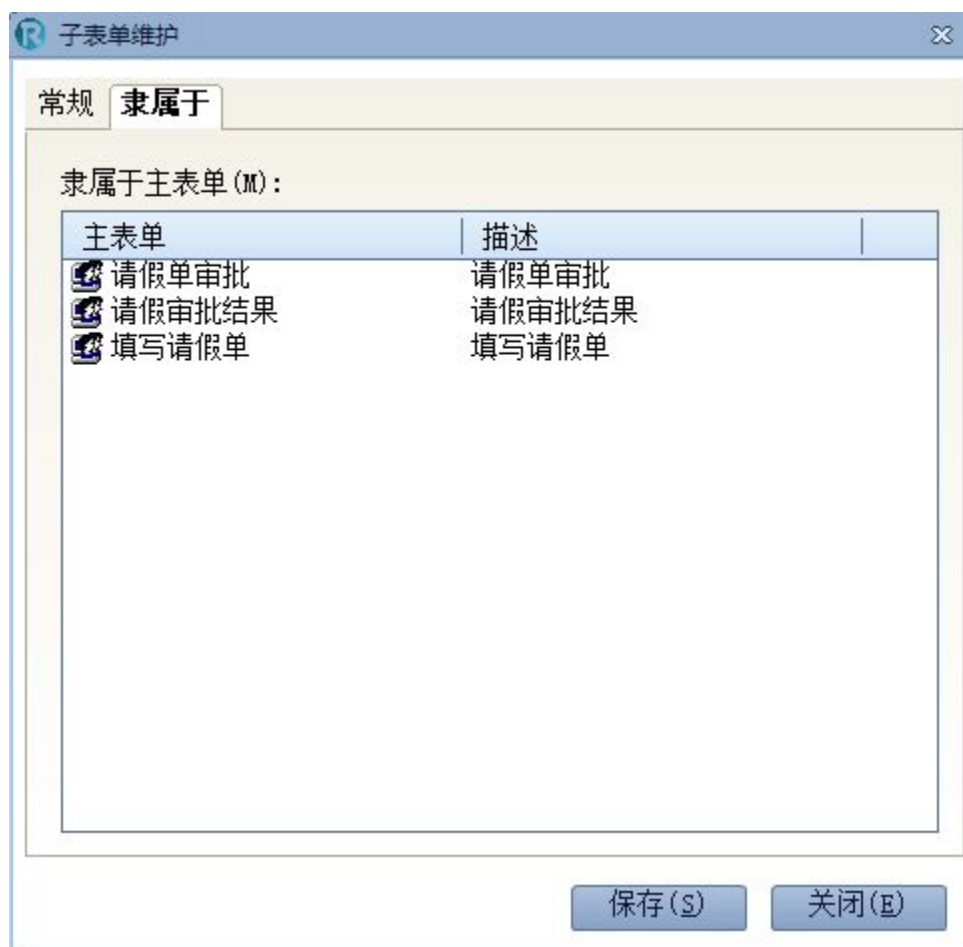
所在程序集: RDIFramework.WorkFlow.dll

描述:

保存(S) 关闭(E)

图：修改子表单

在修改子表单主界面，我们还可以查看子表单隶属关系（子表单隶属于那些主表单），如下图所示：



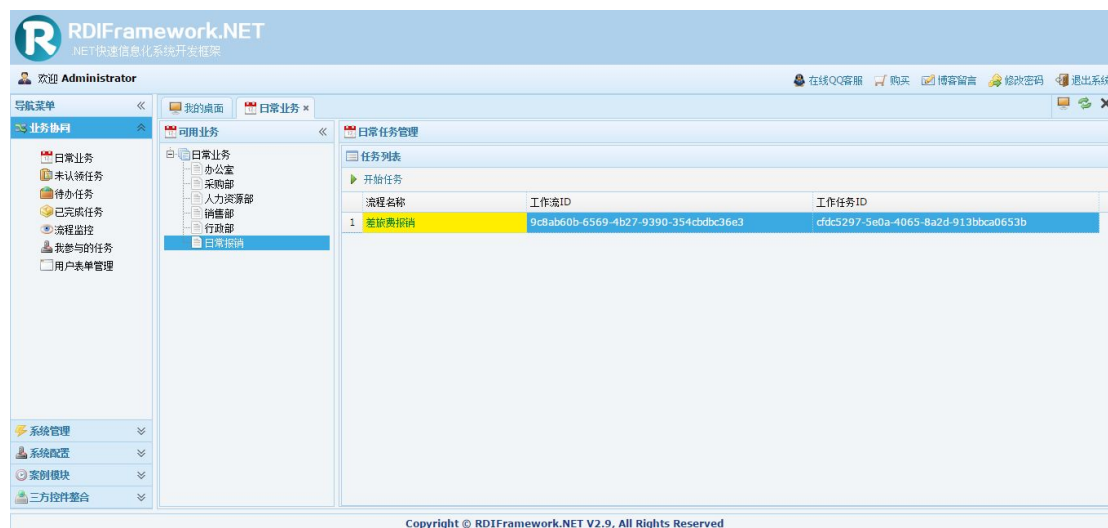
图：子表单隶属关系

◆ 删除子表单

6.2 WebForm 部分

6.2.1 日常业务

“日常业务”模块主要展示了当前登录用户可以启动的业务流程列表。流程引擎会自动根据当前用户可以启动的业务流程自动加载流程列表到“可用业务”树中。“日常业务”模块主界面如下图所示：



图：日常业务主界面

在“可用业务”列表选择一个分类，在右侧选择我们需要启动的业务流程，点击“开始任务”，即可启动一个任务，如下图所示，我们启动了“员工请假流程”。

启动任务 - Google Chrome

localhost:8090/WorkFlow/StartWorkFlow.aspx?workFlowId=1f8632c4-cfa5-4948-8460-66e281678633&work1

流程信息			
流程模板:	员工请假流程	任务模板:	员工请假单

业务信息			
流程业务名:	员工请假流程WF-20142808012807 *	流程编号	WF-20142808012807
流程说明:		紧急程度	普通 ▼ *

业务表单			
请假单			
请假人	26F43BC9-AE6D-42D2-BAC9-F4237A949484	姓名	超级管理员
部门	系统开发部	岗位	
开始时间	2014/11/8	结束时间	2014/11/8
类型	病假 ▼	请假天数	1
事由说明	请假天数超过10天的必须由副总审批!		
附件			
附件信息			
选择文件 未选择文件 确认上传			

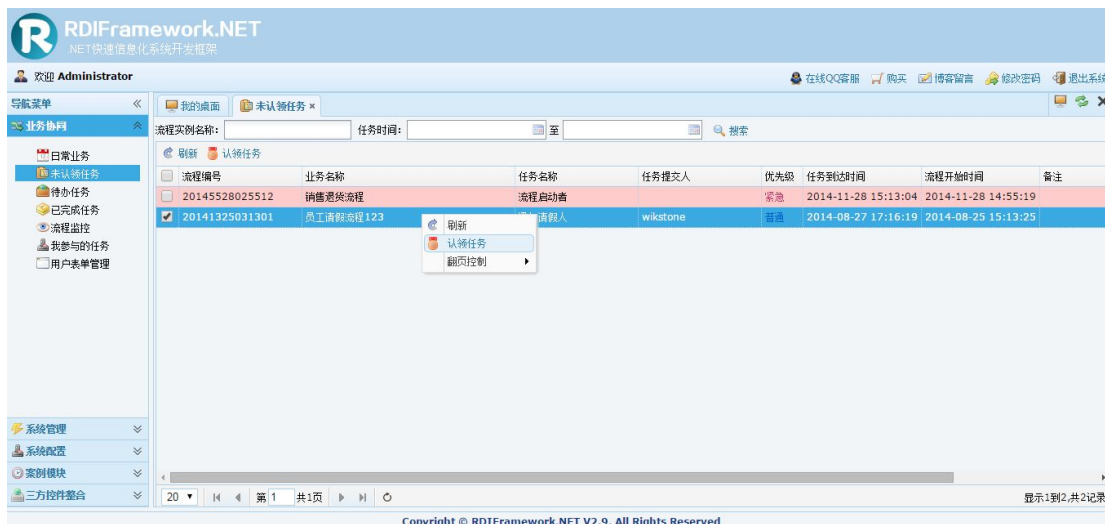
流程操作	
提交	取消

图：启动任务界面

在“启动任务”主界面，表单列表会显示主表单中所有的子表单列表，以供用户输入各表单必要信息。

6.2.2 未认领任务

处理任务前，首先要认领任务。认领任务到“未认领任务”主界面进行认领，如下图所示。选择待认领的任务后点击“认领任务”就可以对所选任务进行认领。同时我们也可以按业务名称以及任务时间进行精确或模糊查询。



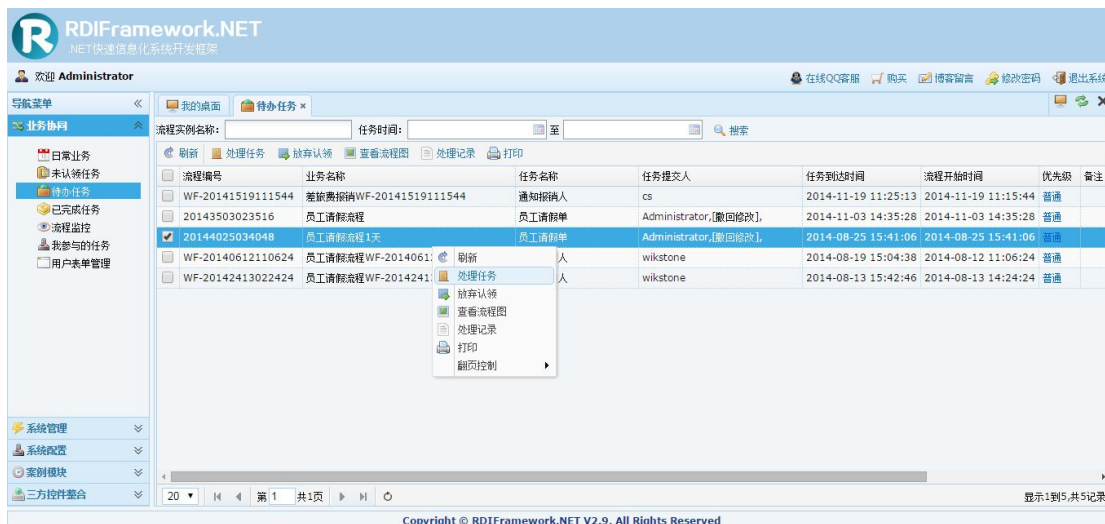
图：未认领任务界面

当前在进入框架后会默认在右下角有提示当前用户有多少未认领的任务，或者在“我的桌面”的“待办任务”工作区域显示有当前用户有多少个待办任务还没有处理，如下图所示：



6.2.3 待办任务

“待办任务”模块是我们真正处理业务的地方，已认领的任务会进入到“待办任务”列表中，在这儿我们可以按条件搜索待办任务、处理所选任务、放弃认领所选任务、查看所选任务的流程图、查看所选任务的处理记录。待办任务主界面如下图所示：



图：待办任务主界面

1、处理任务

在“待办任务”窗口，选择一个任务后点击“处理任务”，即可打开“处理任务”主界面，如下图所示，“处理任务”界面上方的命令按钮都是根据当前任务节点的设置动态生成的。

处理任务 - Google Chrome

localhost:8090/WorkFlow/ProcessingWorkFlowTask.aspx?operatorInsId=9c3fd81d-2aa3-459d-ae21-b97cb4

审批列表

审批信息

审批人	部门	职务	审批结果	审批意见	审批时间
陈俊熙	信息中心		同意	同意申请, 请做好工作的交接。	2014/11/8 10:32:03

审批表单

审批信息

审批人Id:	41DD6B83-0784-4A8C-823F-EE5C300554EF	审批人:	陈俊熙
部门:	信息中心	职务:	
审批结果:	<input checked="" type="radio"/> 同意 <input type="radio"/> 拒绝	审批时间:	2014/11/8 10:32:03
审批意见:	<div>同意申请, 请做好工作的交接。 </div>		

流程操作

认领 草稿 放弃认领 指派 提交 退回给[Administrator]修改

指定下一任务处理人: 选择用户

取消 预览并打印

图：处理任务界面

“指派”命令按钮：

任务在需要指派给别人处理时使用指派功能，指派规则：

- 1、只有认领的任务才能指派，通过指派任务就会到达被指派人的未认领任务里，原处理者实例标记为已指派某人处理。
- 2、被指派人如果不接受被指派的任务，可以放弃认领任务。
- 3、只能指派给具体的一个人，指派后原处理人还可以继续处理该任务。
- 4、被指派过的任务，可以再次指派。

点击“指派”命令按钮，即可打开“任务指派”窗口，如下图所示，选择一个用户后确定即可。

任务指派

User Name: Real Name: Search

<input type="checkbox"/>	编号	登录名	用户名	部门	
<input type="checkbox"/>	1	Administrator	Administrator	超级管理员	系统开发部
<input type="checkbox"/>	2	yonghu	yonghu	胡勇	系统开发部
<input type="checkbox"/>	3	rdiframework	rdiframework	EricHu	质量管理本部
<input type="checkbox"/>	4	jxChen	wikstone	陈俊熙	信息中心
<input type="checkbox"/>	5	mjLi	mjLi	李明杰	采购部
<input type="checkbox"/>	6	Admin	Admin	超级管理员	质量管理本部
<input type="checkbox"/>	7	tyLuo	tyLuo	罗天祐	采购部
<input type="checkbox"/>	8	ytLiang	ytLiang	梁雨婷	信息中心
<input type="checkbox"/>	9	cdx	cdx	陈大贤	移动终端事业部
<input checked="" type="checkbox"/>	10	chenp	chenp	陈鹏	灵通部
<input type="checkbox"/>	11	cg	cg	陈高	无线增值事业部
<input type="checkbox"/>	12	chf	chf	陈海芳	质量管理部
<input type="checkbox"/>	13	chy	chy	陈海英	信息中心
<input type="checkbox"/>	14	chenxy	chenxy	陈小叶	无线增值事业部
<input type="checkbox"/>	15	luoy	luoy	罗燕	信息中心
<input type="checkbox"/>	16	fanwei	fanwei	凡卫	质量管理本部

50 Page 1 of 3 Displaying 1 to 50 of 123 items

确定 关闭

图：任务指派

“指定下一任务处理人”命令按钮，点击选择用户按钮，即可打开“用户选择”窗口，如下图所示：



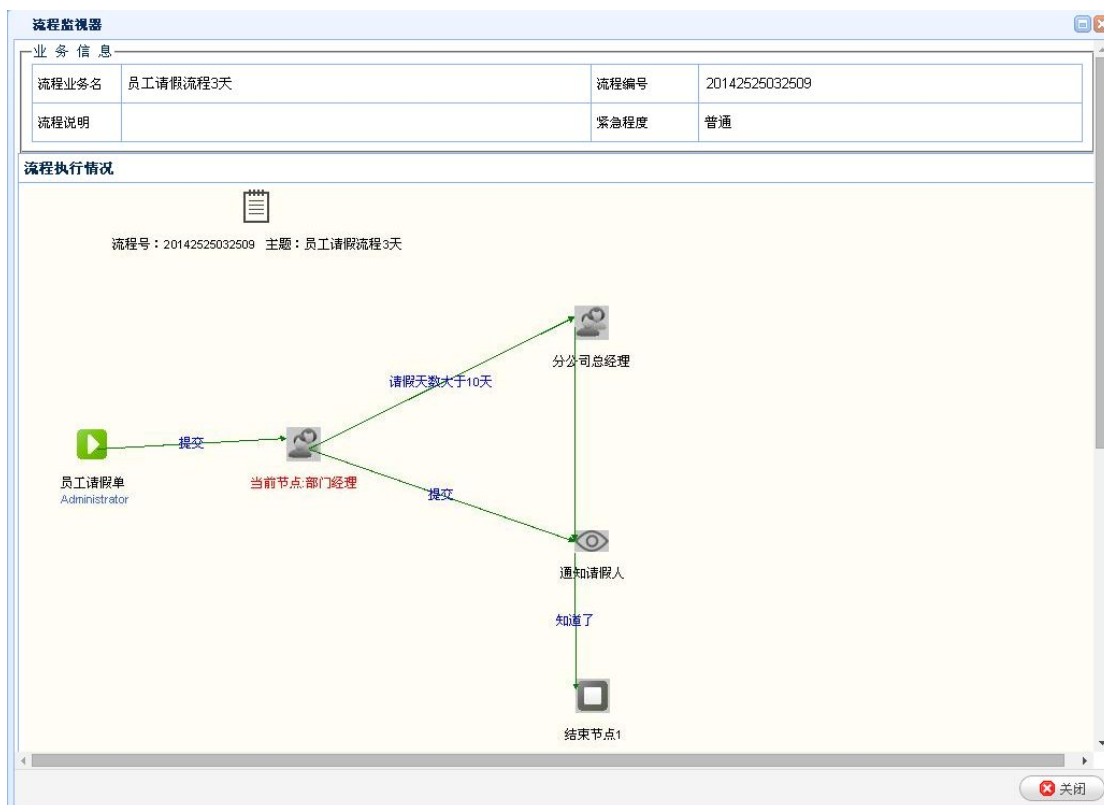
图：用户选择

2、取消认领

在“待办任务”主界面，可以对已认领的任务放弃认领，放弃认领后任务进入未认领任务列表。

3、查看流程图

在“待办任务”主界面，选择一个任务，点击“查看流程图”，即可查看所选任务以图形方式展示的流程执行情况图，如下所示：



图：流程监视器

在流程执行状态图上，彩色表示已经走过的流程节点，灰色表示未走过的流程节点，通过流程图可以清晰的看到流程所处位置和后面的路线。

4、处理记录

在“待办任务”主界面，选择一个任务，点击“处理记录”，即可查看所选任务的所处理记录，如下所示：



图：流程处理记录（流程轨迹）

5、打印

打印功能可以打印当前待办任务列表，如下图所示：



图：打印待办任务列表

6.2.4 已完成任务

任务处理完成，进入到“已完成任务”列表，如下图所示。在已完成任务模块可以按条件查询已完成的任务，查看所选任务、查看任务流程图等。

The screenshot displays the 'Completed Tasks' (已完成任务) module in the RDIFramework.NET application. The interface includes a sidebar with navigation options like 'Business Category' (业务分类), 'System Management' (系统管理), and 'Case Module' (案例模块). The main area shows a table of completed tasks. A task is selected, and a context menu is open, allowing users to 'View Task' (查看任务) or 'View Flowchart' (查看流程图). The table columns include 'Task ID' (流程实例编号), 'Task Name' (任务名称), 'Task Submitter' (任务提交人), 'Priority' (优先级), 'Task Start Time' (任务开始时间), 'Task End Time' (任务结束时间), and 'Task Start Flow Time' (流程开始时间).

流程实例编号	业务名称	任务名称	任务提交人	优先级	任务开始时间	任务结束时间	流程开始时间
WF-20153707093747	员工请假流程 WF-20153707093747	员工请假单	Administrator	普通	2015-05-07 21:37:48	2015-05-07 21:37:49	2015-05-0
20154206104249	员工请假流程	员工请假单	Administrator	普通	2015-03-06 10:43:08	2015-03-06 10:43:08	2015-03-0
20145528025512	销售退货流程	销售退货单	Administrator	紧急	2014-11-28 14:55:19	2014-11-28 14:55:23	2014-11-2
20144220124228	差旅费报销	通知报销人	cs	紧急	2014-11-20 12:58:40	2014-11-20 13:11:06	2014-11-2
20144220124228	差旅费报销	填写差旅报销单	Administrator	紧急	2014-11-20 12:48:09	2014-11-20 12:48:13	2014-11-2
WF-20141519111544	差旅费报销 WF-20141519111544	填写差旅报销单	Administrator	普通	2014-11-19 11:15:44	2014-11-19 11:15:45	2014-11-1
20140915050926	员工请假流程-test	通知请假人	wikstone	普通	2014-10-15 17:12:25	2014-10-15 17:12:58	2014-10-1
20140915050926	员工请假流程-test	员工请假单	Administrator.[撤回修改]	普通	2014-10-15 17:09:52	2014-10-15 17:11:18	2014-10-1
WF-20142027112015	员工请假流程 WF	通知请假人	chenp	普通	2014-08-27 11:31:57	2014-08-27 11:32:35	2014-08-2
WF-20142027112015	员工请假流程 WF	通知请假人	lsx	普通	2014-08-27 11:22:14	2014-08-27 11:30:36	2014-08-2
WF-20142027112015	员工请假流程 WF	通知请假人	Administrator	普通	2014-08-27 11:20:21	2014-08-27 11:20:23	2014-08-2
20141627101644	员工请假流程-1	通知请假人	chenp	紧急	2014-08-27 11:11:03	2014-08-27 11:12:17	2014-08-2
20141627101644	员工请假流程-12天	通知请假人	Administrator	紧急	2014-08-27 10:17:17	2014-08-27 10:17:17	2014-08-2
WF-20142225042258	员工请假流程 WF-20142225042258	通知请假人	Administrator	普通	2014-08-25 16:22:58	2014-08-25 16:22:58	2014-08-2

图：已完成任务主界面

1、查看任务

在“已完成任务”列表中选择一个任务，点击“查看任务”按钮，可以查看所选任务的详细信息，包含任务的业务表单列表、流程处理情况、审批列表、审批表单等。如下图所示：

处理任务 - Google Chrome

localhost:8090/WorkFlow/ProcessingWorkflowTask.aspx?operatorInsId=63e52e6d-aa78-4dbf-90ec-6e616c

审批列表

审批信息

审批人	部门	职务	审批结果	审批意见	审批时间
陈鹏	灵通部		同意	同意!	2014/8/27 11:11:03
陈俊熙	信息中心		同意	同意申请, 请分公司总经理审批~!	2014/8/27 11:07:38

审批表单

审批信息

审批人Id: 41DD6B83-0784-4A8C-823F-EE5C300554EF

审批人: 陈俊熙

部门: 信息中心

职务:

审批结果: ☒ 同意 ☐ 拒绝

审批时间: 2014/8/27 11:07:38

审批意见:

同意申请, 请分公司总经理审批~!

流程操作

撤回修改 认领 草稿 放弃认领

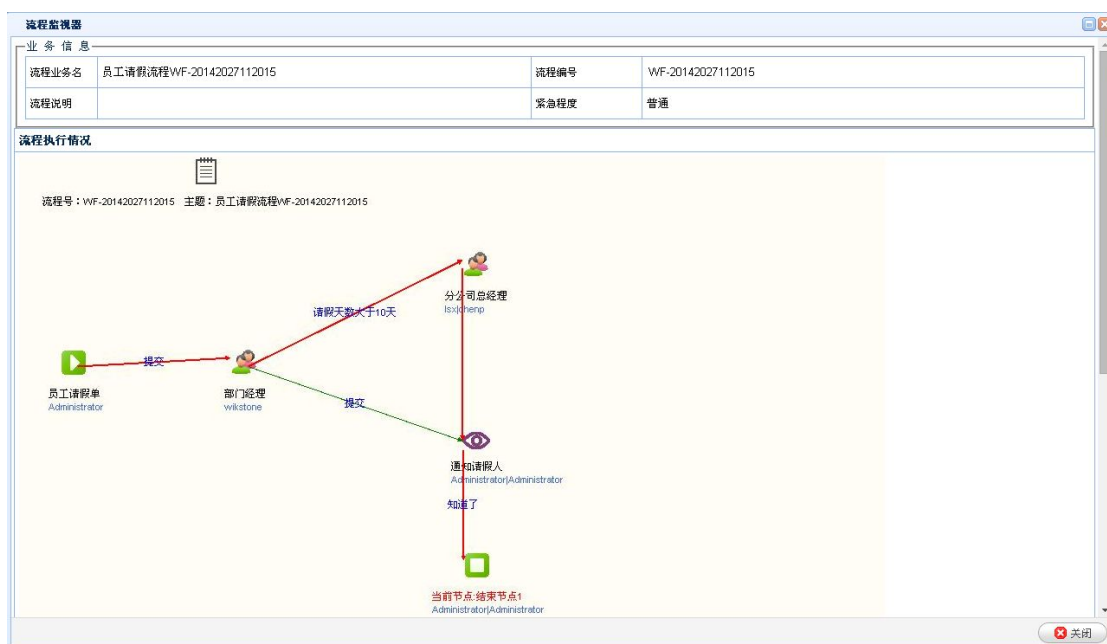
取消 预览并打印

图：已完成任务-查看任务

在上面的窗口中, 对于已提交的任务, 在待处理用户未认领的情况下, 可以单击“撤回修改”按钮对已完成任务重新处理。

2、查看流程图

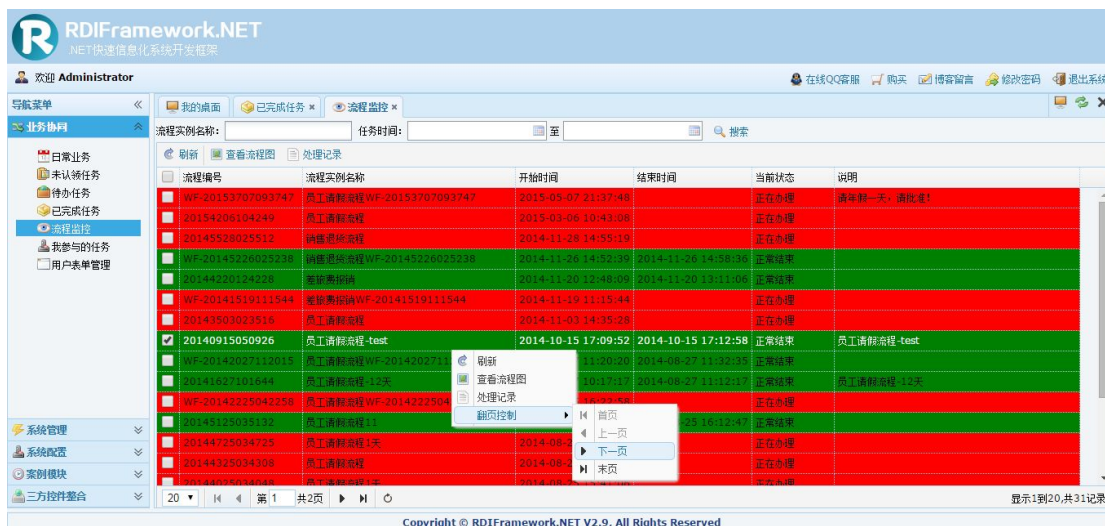
在“已完成任务”主界面, 选择一个任务, 点击“查看流程图”, 即可查看所选任务以图形方式展示的流程执行情况图, 如下所示:



图：已完成任务-流程执行状态图

6.2.5 流程监控

流程监控模块主要用于监控流程的执行情况，界面上通过不同颜色显示流程的几种状态，如：还未执行、正在办理、正常结束。在流程监控主界面，我们可以组合条件搜索业务流程，同时还可以查看所选业务流程的状态以及处理记录。流程监控主界面如下图所示：



图：流程监控

6.2.6 我参与的任务

我参与的任务主界面如下图所示：

RDIFramework.NET
NET快速信息化系统开发框架

欢迎 Administrator

在线QQ客服 购买 博客留言 修改密码 退出系统

导航菜单 << 我的桌面 已完成任务 × 流程监控 × 我参与的任务 ×

业务协同

流程实例名称: 任务时间: 至 搜索

刷新 查看任务 查看流程图 处理记录

流程编号	业务名称	任务名称	任务提交人	状态	任务到达时间	任务结束时间	处理时间
WF-20153707093747	员工请假流程WF-20153707093747	员工请假单	Administrator	执行完毕	2015-05-07 21:37:48	2015-05-07 21:37:49	2015-05-07
20154206104249	员工请假流程	员工请假单	Administrator	执行完毕	2015-03-06 10:43:08	2015-03-06 10:43:08	2015-03-06
20145528025512	销售退货流程	销售退货单	Administrator	执行完毕	2014-11-28 14:55:19	2014-11-28 14:55:23	2014-11-28
20144220124228	差旅费报销	通知报销人	cs	执行完毕	2014-11-20 12:58:40	2014-11-20 13:11:06	2014-11-20
20144220124228	差旅费报销	填写差旅报销单	Administrator	执行完毕	2014-11-20 12:48:09	2014-11-20 12:48:13	2014-11-20
WF-20141519111544	差旅费报销WF-20141519111544	填写报销单	cs	进行中	2014-11-19 11:25:13		
WF-20141519111544	差旅费报销WF-20141519111544	报销单	Administrator	执行完毕	2014-11-19 11:15:44	2014-11-19 11:15:45	2014-11-19
20143503023516	员工请假流程	请假单	Administrator,[撤回修改]	进行中	2014-11-03 14:35:28	2014-11-03 14:35:29	2014-11-03
20140915050926	员工请假流程-test	请假单	wikstone	执行完毕	2014-10-15 17:12:25	2014-10-15 17:12:58	2014-10-15
20140915050926	员工请假流程-test	请假单	Administrator,[撤回修改]	执行完毕	2014-10-15 17:09:52	2014-10-15 17:11:18	2014-10-15
WF-20142027112015	员工请假流程WF-20142027112015	通知请假人	enp	执行完毕	2014-08-27 11:31:57	2014-08-27 11:32:35	2014-08-27
WF-20142027112015	员工请假流程WF-20142027112015	通知请假人	enp	执行完毕	2014-08-27 11:22:14	2014-08-27 11:30:36	2014-08-27
WF-20142027112015	员工请假流程WF-20142027112015	员工请假单	Administrator	执行完毕	2014-08-27 11:20:21	2014-08-27 11:20:23	2014-08-27

20 第 1 共 3 页 显示 1 到 20 共 46 记录

Copyright © RDIFramework.NET V2.9, All Rights Reserved

图：我参与的任务

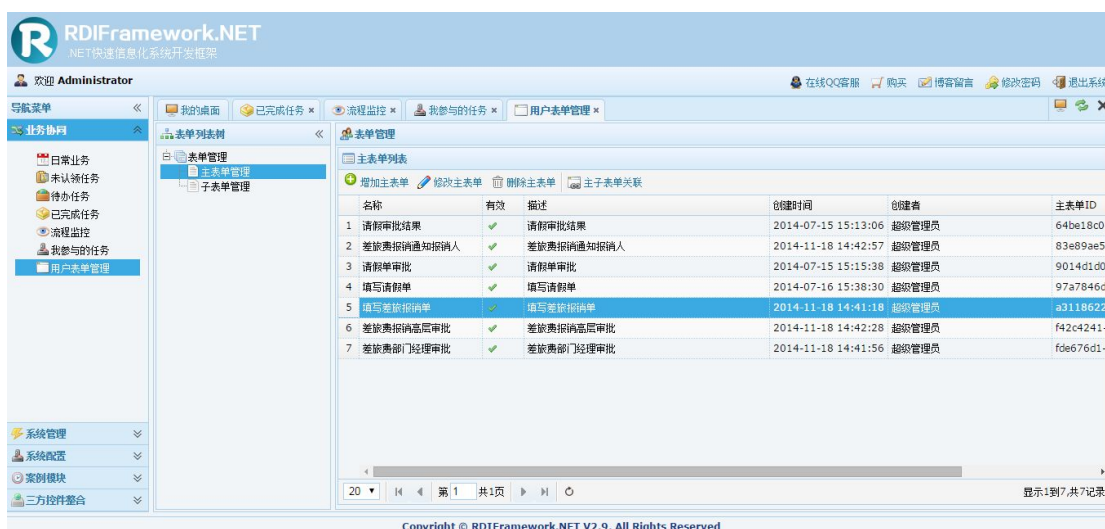
6.2.7 用户表单管理

表单是业务的展现形式，根据工作流组件的设计，表单采用用户控件 UserControl 来设计，表单有两种形式，主表单和子表单，主表单是由子表单组成的，一个主表单可以包含多个子表单，每个子表单就是一个 UserControl，任务节点中的交互节点可以挂接一个主表单，业务平台可以动态加载主表单里的 UserControl。

导航菜单中选择“业务协同”→“用户表单管理”，即可打开用户表单管理主界面。主表单与子表单的管理都在同一界面上展示。

1、主表单管理

在“表单列表树”中选择“主表单管理”即可打开主表单列表，我们可以增加主表单、修改主表单、删除主表单以及设置主子表单关联。



◆ 增加主表单

◆ 修改主表单



修改主表单

名称: 填写请假单

有效: ☒ 有效

备注:

确定 关闭

◆ 删除主表单

◆ 主子表单关联



名称: 填写请假单

所有用户表单		
	名称	
1	附件	
2	请假单	
3	审批表单	
4	审批列表	

➡

⬅

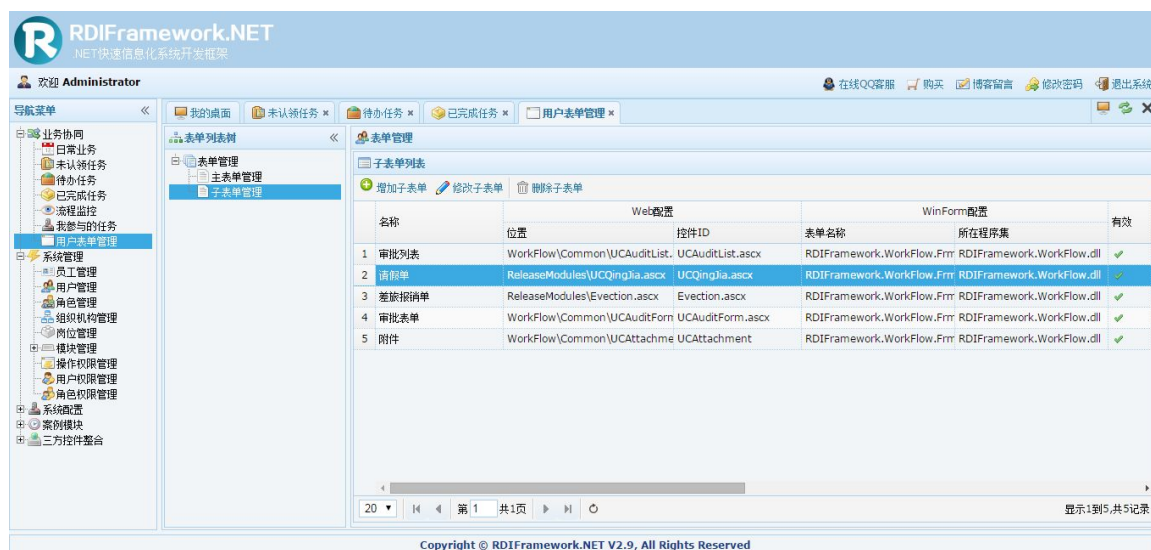
已选用户表单		
	名称	状态
1	请假单	新建
2	附件	修改

双击即可添加或删除，右侧可以使用右键快捷菜单进行相应的操作！

确定 关闭

2、子表单管理

在“表单列表树”中选择“子表单管理”即可打开子表单列表，我们可以增加子表单、修改子表单、删除子表单。



◆ 增加子表单

增加子表单

子表单名:

表单类型:

☐ WinForm ☒ Web ☐ WinForm/Web

位置:

控件ID:

表单名称:

所在程序集:

有效:

☒ 有效

描述:

确定

关闭

◆ 修改子表单

 修改子表单

子表单名:	请假单
表单类型:	<input type="radio"/> WinForm <input type="radio"/> Web <input checked="" type="radio"/> WinForm/Web
位置:	ReleaseModules\UCQingJia.ascx
控件ID:	UCQingJia.ascx
表单名称:	RDIFramework.WorkFlow.FrmQingJia
所在程序集:	RDIFramework.WorkFlow.dll
有效:	<input checked="" type="checkbox"/> 有效
描述:	

 确定  关闭

◆ 删除子表单

第 7 章 开发业务流程

平台能够方便的开发业务界面，提供与数据库的接口，不需要太多编程知识就可开发业务流程，基本步骤如下：



第一步：利用建模工具画好业务流程图，配置好每一个任务节点的处理者和处理方式。

第二步：利用 **Visual Studio** 制作工作流业务表单。

第三步：利用建模工具在流程图的每一个节点上配置业务表单。

第四步：把业务表单的界面文件和生成的 **dll** 发布到服务器。

做以上基本配置即可完成工作流开发，工作流平台不需要重新编译。

7.1 创建流程图

创建流程图是业务流程的基础，只有创建好了流程图，业务才能按照我们的要求进行流转，对于如何创建流程图可以参考我们已有的业务流程。下面我们来重点说一下业务表单的制作。

7.2 制作业务表单

工作流平台开发的主要任务是开发表单，要在工作流平台上展现丰富多彩的业务，都依靠表单，所以能开发出什么样子的表单直接关系到业务系统的成败。下面介绍如何进行表单开发。

◆ 表单组成

表单分为主表单和子表单，主表单是由若干子表单组成的，一个主表单可以包含多个子表单，一个子表单可以属于多个主表单，主表单和子表单在系统中都是唯一的。子表单在 Visual Studio 中体现为 UserControl，而主表单则是很多 UserControl 的集合。所以只要会开发 UserControl 就会开发表单，门槛比较低。只要遵循表单开发规则就能开发出功能强大的表单，当然我们主要的任务就是开发子表单了。

◆ 表单开发规范

表单开发规则是比较固定的，也是比较严格的，只有遵循这些规则才能开发出平台可以识别的表单，平台才能很好的加载它，只要遵循这些规则其他功能可以自由发挥。只要做到系统界面协调即可。

7.3 Web 业务表单

7.3.1 表单开发步骤

Web 业务表单的开发，我们可以在单独的项目中进行添加业务表单，也可以在我们现有的 Web 项目的基础上添加，在这儿为了方便，我就以 RDIFramework.NET Web 项目为基础创建业务表单。

第一步：添加业务表单用户控件，如下图所示：



然后编写表单文件的后台代码，具体可见我们项目下表单文件的源码。

第二步：发布表单

发布表单与常规的 Web 发布方式一样。

第三步：

发布后把生成的 Dll 和页面文件复制到框架 Web 项目的相应目录下。Dll 一定要复制到 Bin 目录下，只复制新生成的 Dll，Bin 目录里已经有的系统需要的 DLL 不能覆盖，以防出现版本不兼容问题。把页面文件和表单文件复制到 ReleaseModules 目录下，平台以相对路径的方式调用。

7.3.2 表单界面开发

首先了解一下，什么是表单文件，表单文件就是一个个具体的 UserControl 即 *.ascx 文件，所有的业务功能都是在表单上展现的。如下图：

请假人	26F43BC9-AE6D-42D2-BAC9-F4237A949484	姓名	超级管理员
部门	系统开发部	岗位	
开始时间	2014/11/8	结束时间	2014/11/8
类型	病假	请假天数	1
事由说明	请假天数超过10天的必须由副总审批!		
<div></div>			

在框架中表单有 3 个状态：新建、修改、查看，分别表示表单交互数据在表单提交时的处理方式，新建表示数据要插入到数据库，修改表示数据要更新到数据库，查看表示只是浏览不做更新。因为我们的设计目的是尽量使用一个表单来完成这三种状态的展现（当然也可以根据三种状态分别制作 3 个表单），尽量在一个表单上来处理业务避免重复开发。具体说明一下三种状态：

1、新建

值: `WorkConst. STATE_ADD`

表示该表单在该任务节点实例中是以新建方式出现的，用户通过平台打开表单的时候表单处于新建状态，此时如果点击提交按钮的时候，表单上的数据一般需要更新到数据库。

2、修改

值: `WorkConst. STATE_MOD`

表示该表单在该任务节点实例中是以修改方式出现的，用户通过平台打开表单的时候表单处于修改状态，此时如果点击提交按钮的时候，表单上的数据一般需要更新到数据库。

3、查看

值: `WorkConst. STATE_VIEW`

表示该表单在该任务节点实例中是以查看方式出现的，用户通过平台打开表单的时候表单处于查看状态，此时如果点击提交按钮的时候，表单上的数据一般不需要更新到数据库。

说明:以上参数是平台为表单开发提供的接口参数，可以利用这些参数来控制表单数据的展现，例如初始化，控制界面元素的显示和隐藏等。具体参见差旅报销表单 demo。

7.3.3 表单开发重点说明

一、表单数据如何展现

表单数据展现是指表单显示数据表中的数据，根据表单状态的不同一个表单有不同的展现，读取表单数据的代码一般写在 UserControl 的 Page_Load 事件中。如下图一段代码：

```
protected void Page_Load(object sender, EventArgs e)
{
    base.Page_Load();
    Panel2.Visible = CtrlState != WorkConst.STATE_VIEW; //控制编辑区是否显示
    if (!IsPostBack)
    {
        InitData();
    }
}
```

注意以下问题：

1、表单配置的状态，新建、修改、查看。例如如果配置为新建状态，那么相应的代码就要支持新建，初始化新建的默认值。如果配置为修改和查看，那么相应代码就要支持修改和查看，从数据表中读取已有的信息。

2、特别注意的是表单退回重新处理和保存草稿的情况，这两种虽然该表单配置了新建状态，但是数据表里已经有数据了，这个时候就不能以新建 WorkConst.STATE_ADD 作为首要的判断了，而是需要先查询数据表中是否已经有数据了，如果有那么显示已有的数据。

3、实际项目应用中都要考虑表单退回处理和草稿的情况。

二、表单数据如何提交

表单数据提交是指流程提交时对表单数据的处理，如果以草稿方式提交，那么只会提交表单数据，流程不会流转。该过程是通过重载 SaveUserControl() 方法实现的，在 SaveUserControl(bool isDraft)里的代码在流程提交或者保存草稿时都会被执行。

注意以下问题：

- 1、根据表单状态编写相应处理代码，以提供平台展现表单的支持。
- 2、特别注意的是如果表单是退回处理和草稿时，避免数据重新提交。

7.3.4 表单继承的基类

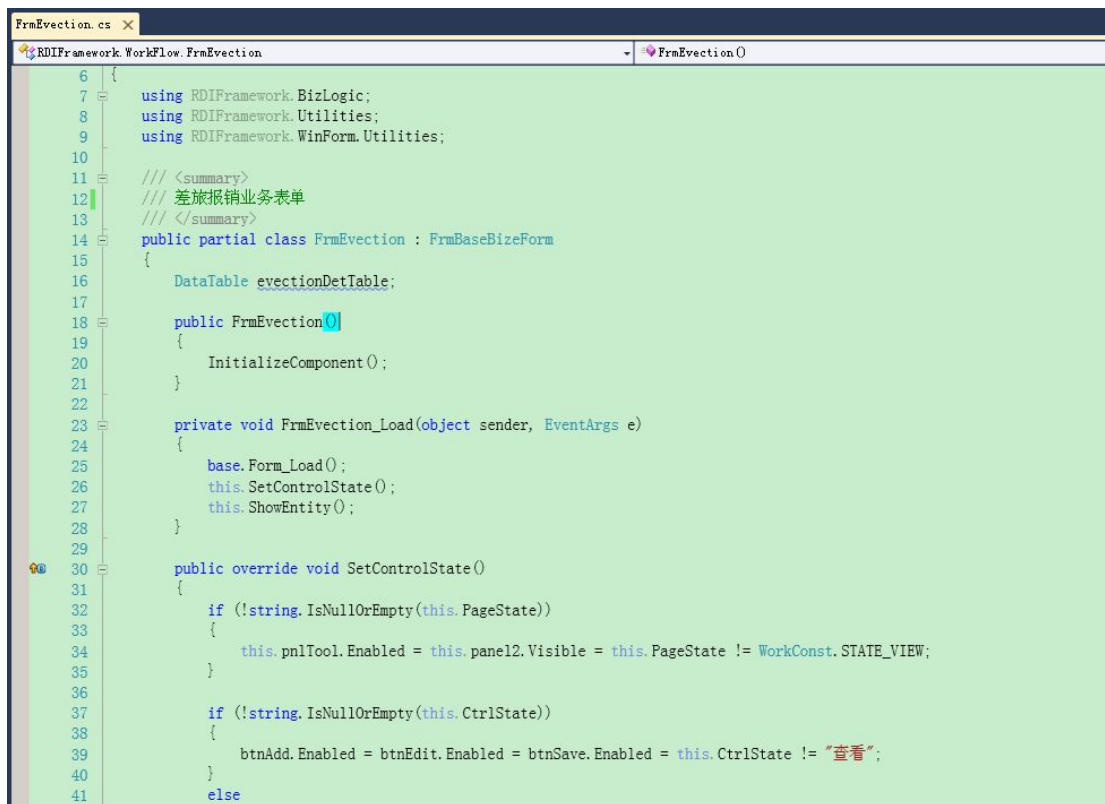
把 System.Web.UI.UserControl 修改为 BaseUserControl。

下面是一个表单代码的模板

```
using RDIFramework.BizLogic;
using RDIFramework.Utilities;
/// <summary>
/// 差旅报销表单
/// </summary>
public partial class Evection : BaseUserControl
```

7.4 WinForm 业务表单

WinForm 业务表单的开发与常规的表单开发类似，只是要注意是继承自“FrmBaseBizForm”基窗体。可以参见我们的相关案例业务表单。如下图所示：



界面如下：

启动任务

提交 | 关闭

表单列表

差旅报销单

附件

流程信息

流程模版: 差旅费报销 任务模版: 填写差旅报销单

业务信息

流程业务名: 差旅费报销 流程编号: 20141827011832

流程说明: 紧急程度:

差旅报销单

保存差旅报销单

单据编号: Evection-c536747e-015f 提交日期: 2014 - 11 - 27 报销人: 超级管理员 所属部门: 系统开发部

出差事由:

报销明细

起点: 终点: 交通工具: 飞机 车/船/机票费: 市内交通费: 住宿-天数: 住宿-房价/单价: 出差-天数: 出差津贴/每天: 其他费用:

起点	终点	交通工具	车/船/机票费	市内交通费	住宿-天数	住宿-房价(单价)	出差-天数	出差津贴(每天)	其他费用
----	----	------	---------	-------	-------	-----------	-------	----------	------

费用合计: 减:借款: 支付金额: 大写:

备注:

第 8 章 业务流程实例

RDIFramework.NET, based on .NET, rapid information system development, integration framework, gives users and developers the best .NET framework deployment solution.

Author: [EricHu](#)

Email: 406590790@qq.com

QQ: 406590790

QQ group: 185568405

Framework blog:

<http://www.cnblogs.com/huyong>

<http://blog.csdn.net/chinahuyong>

Email exchange: 406590790@qq.com

Purchase address: <http://yonghu86.taobao.com>

About the author:

About the author: Senior Engineer, Information System Project Manager, DBA. Specialized in Microsoft platform project architecture, management and enterprise solutions, many years of project development and management experience, has organized and developed many large projects, in object-oriented, service-oriented and database field has a certain造诣. Currently mainly engaged in RDIFramework.NET framework technical development, consulting work, mainly serving financial, medical health, railway, telecommunications, logistics, Internet of Things, manufacturing, retail and other industries.

If there are any questions or suggestions, please feel free to contact me!

