

什么是面向引擎的软件开发技术?

现有技术:

面对对象的软件开发方法, 面向对象方法(Object-Oriented Method)是一种把面向对象的思想应用于软件开发过程中, 指导开发活动的系统方法, 简称 OO

(Object-Oriented)方法, 是建立在“对象”概念基础上的方法学。对象是由数据和容许的操作组成的封装体, 与客观实体有直接对应关系, 一个对象类定义了具有相似性质的一组对象。而每继承性是对具有层次关系的类的属性和操作进行共享的一种方式。所谓面向对象就是基于对象概念, 以对象为中心, 以类和继承为构造机制, 来认识、理解、刻画客观世界和设计、构建相应的软件系统。

现有方法的步骤:

- (1) 分析确定在问题空间和解空间出现的全部对象及其属性;
- (2) 确定应施加于每个对象的操作, 即对象固有的处理能力;
- (3) 分析对象间的联系, 确定对象彼此间传递的消息;
- (4) 设计对象的消息模式, 消息模式和处理能力共同构成对象的外部特性;
- (5) 分析各个对象的外部特性, 将具有相同外部特性的对象归为一类, 从而确定所需要的类;
- (6) 确定类间的继承关系, 将各对象的公共性质放在较上层的类中描述, 通过继承来共享对公共性质的描述;
- (7) 设计每个类关于对象外部特性的描述;
- (8) 设计每个类的内部实现(数据结构和方法);
- (9) 创建所需的对象(类的实例), 实现对象间应有的联系(发消息)。

现在技术的缺点:

随着管理软件的复杂度要求越来越高, 用原有面向对象的软件开发技术有以下缺点:

- (1) 软件开发效率低。

(2) 软件开发质量低。

(3) 软件开发成本高。

面向引擎的软件开发方法能大大提高软件开发效率、提高软件开发质量、降低软件开发成本。

面向引擎的软件开发不是某种计算机语言，不是某种单一的引擎，是一种能大幅提高软件开发效率和质量的新一代软件开发技术和软件开发方法。

把面向对象、面向组件的概念提升到面向引擎的层面，推出一系列与业务无关性的功能强大的各种引擎，凡是传统方式能实现的功能用引擎解析的方式结合原有技术编程都能高效实现，兼容原有技术、第三方技术。

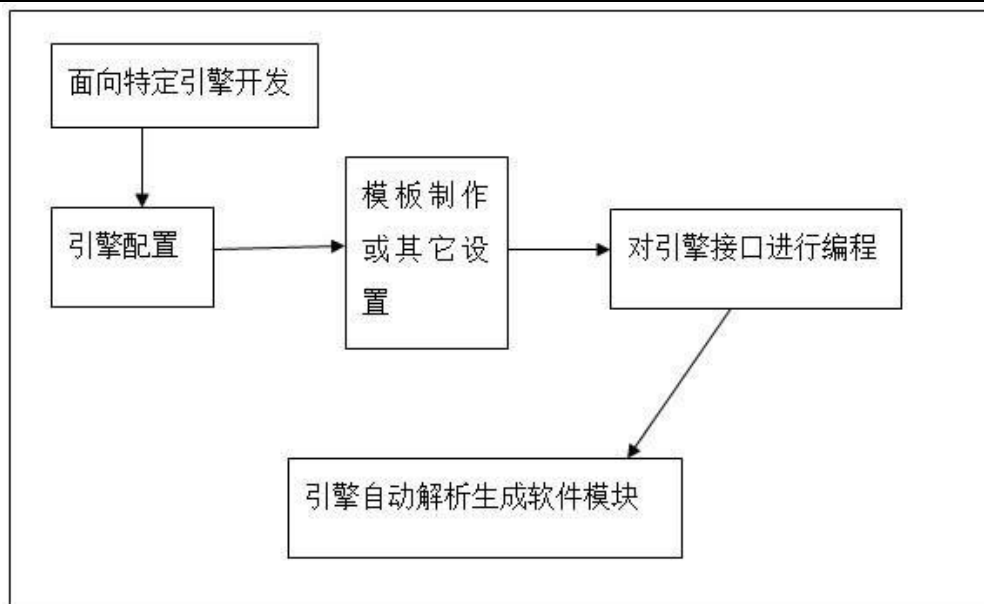
核心理念不是推出一种不用写代码或不懂开发技术就能开发管理软件的系统，也不是推出什么问题都能解决的系统，而是一种大大提高开发效率和质量、降低开发难度的全新软件开发技术和软件开发方法，像当初面向对象的软件开发技术推出一样,相比较结构化程序设计是一次巨大的技术革新,引入了“面向引擎的软

件开发方法”，开发人员只需在引擎中配置各种属性，在引擎的各种事件中编程，

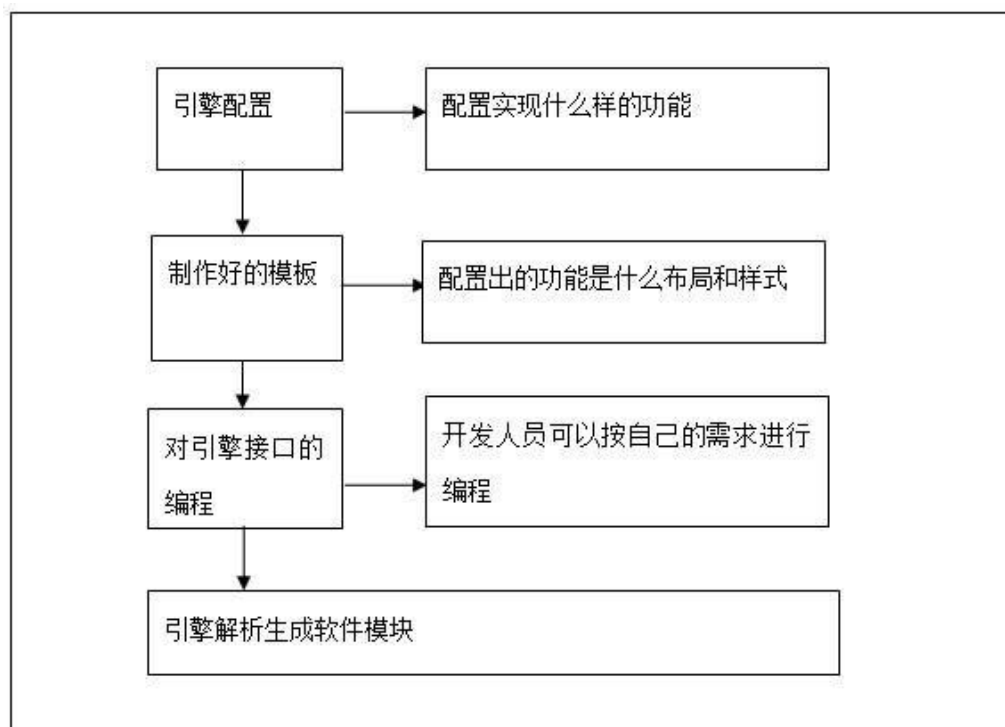
导入通用模板,复杂的事情交给引擎自动解析生成软件功能模块，制作模板和在引擎的事件中编程容纳百家之长，开发人员可以采用擅长的前后端技术来实现。

面向引擎的软件开发方法实现的步骤:

- (1) 根据应用方向采用特定的引擎来开发;
- (2) 对引擎进行配置;
- (3) 制作引擎的模板或其它设置;
- (4) 对引擎的接口进行编程;
- (5) 引擎自动解析生成软件模块。

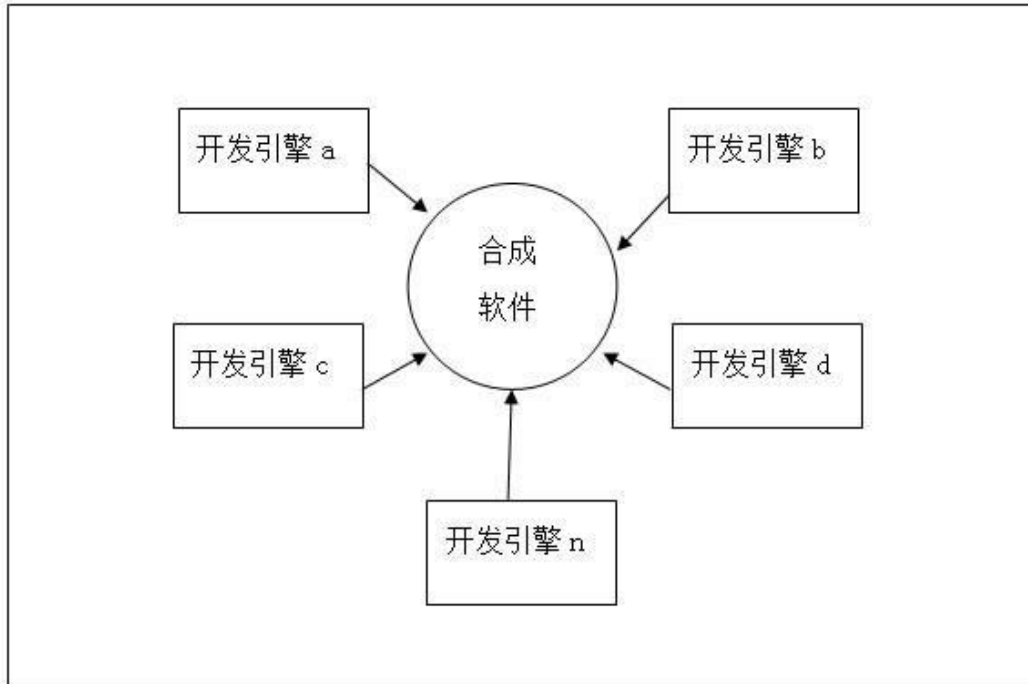


面向引擎的软件开发方法实现原理图-图 1



面向引擎的软件开发方法实现原理图-图 2

从上图可以这样理解，只是软件开发方法与之前的不同，引擎解析生成的软件功能模块与传统软件开发技术开发的功能一样，只是开发效率和开发质量提高了。

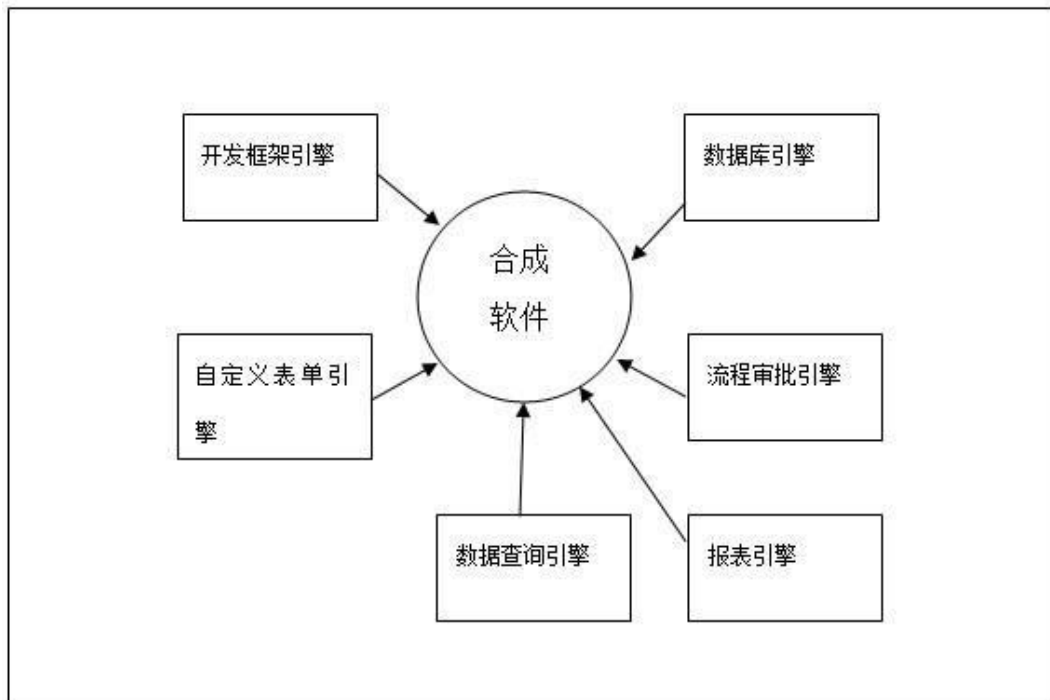


通过各种特定功能的引擎自动合成软件图示-图 3

采用面向引擎的软件开发方法实现方案:

根据应用方向面向特定的引擎开发:

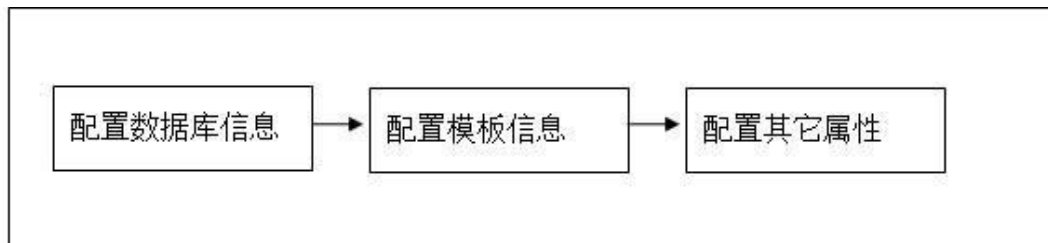
我们把开发引擎分成开发框架引擎、数据库引擎、自定义表单引擎、流程审批引擎、数据查询引擎、报表引擎, 根据特定功能分成不同的开发引擎供开发人使用, 开发人员通过对各引擎的配置和编程生成软件, 如下图所示。



面对各种引擎开发自动合成软件图示-图 4

引擎配置定义:

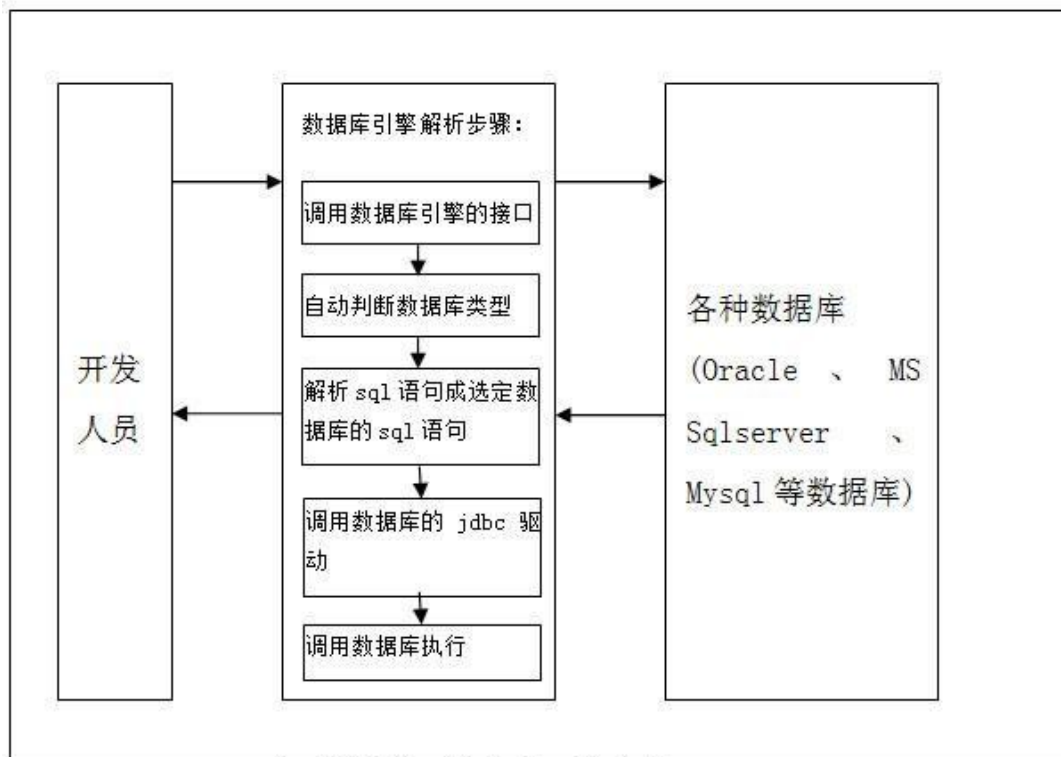
包含配置数据库信息、配置模板信息、配置其它属性。



面向引擎的软件开发方法引擎配置实现原理图-图 5

面向数据库引擎开发软件过程:

面向数据库引擎来建立数据库表和字段、执行 `sql` 语句, 不直接面对数据库, 由数据库引擎来对接各种数据库(Oracle、MS Sqlserver、Mysql 等), 开发人员只需面向数据库引擎调用标准接口编程建立和调用数据库表和字段, 各种数据库中不同的各种复杂的差异性开发人员不用去学习, 简化软件开发时对数据库的管理和调用, 提高数据的执行效率。



面向引擎的软件开发方法-调用数据库-图 6

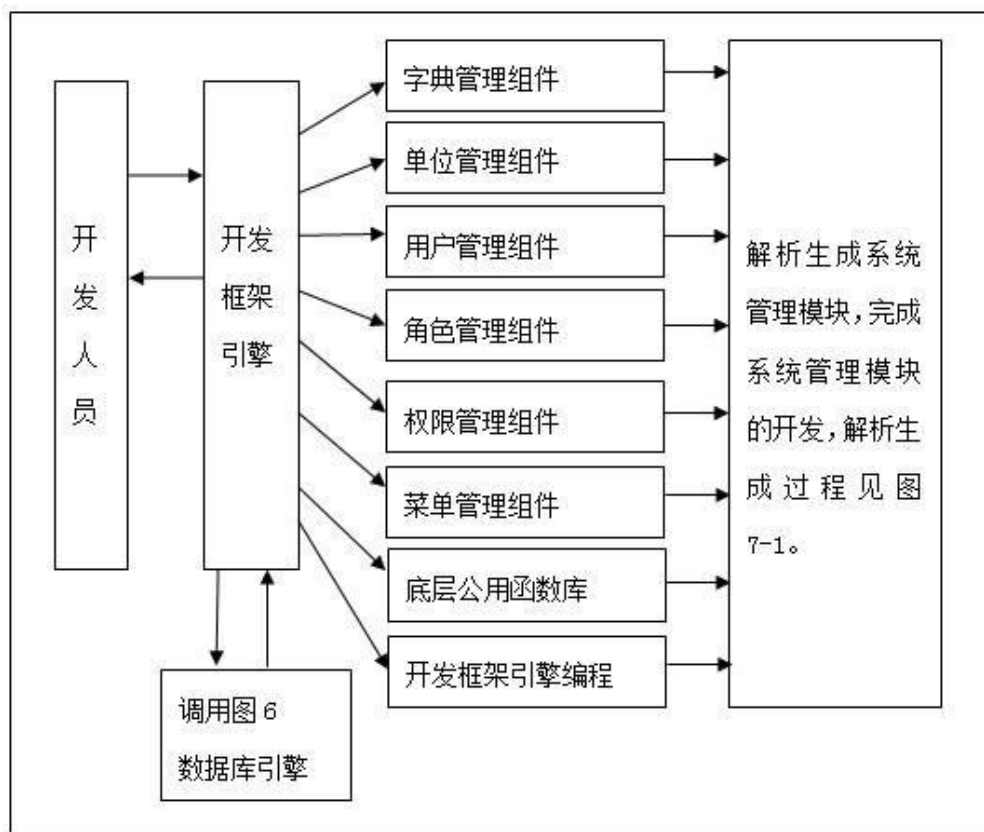
数据库引擎解析步骤:

开发人员只需调用数据库引擎的接口来执行数据库的相关操作，数据库引擎会自动判断采用的数据库类型，解析数据库引擎的接口中的 **sql** 语句成选定数据库的 **sql** 语句，调用数据库的 **jdbc** 驱动，到数据库管理系统中去执行，开发人员只需面向数据库引擎操作，不需要了解复杂的数据库使用和管理技术。

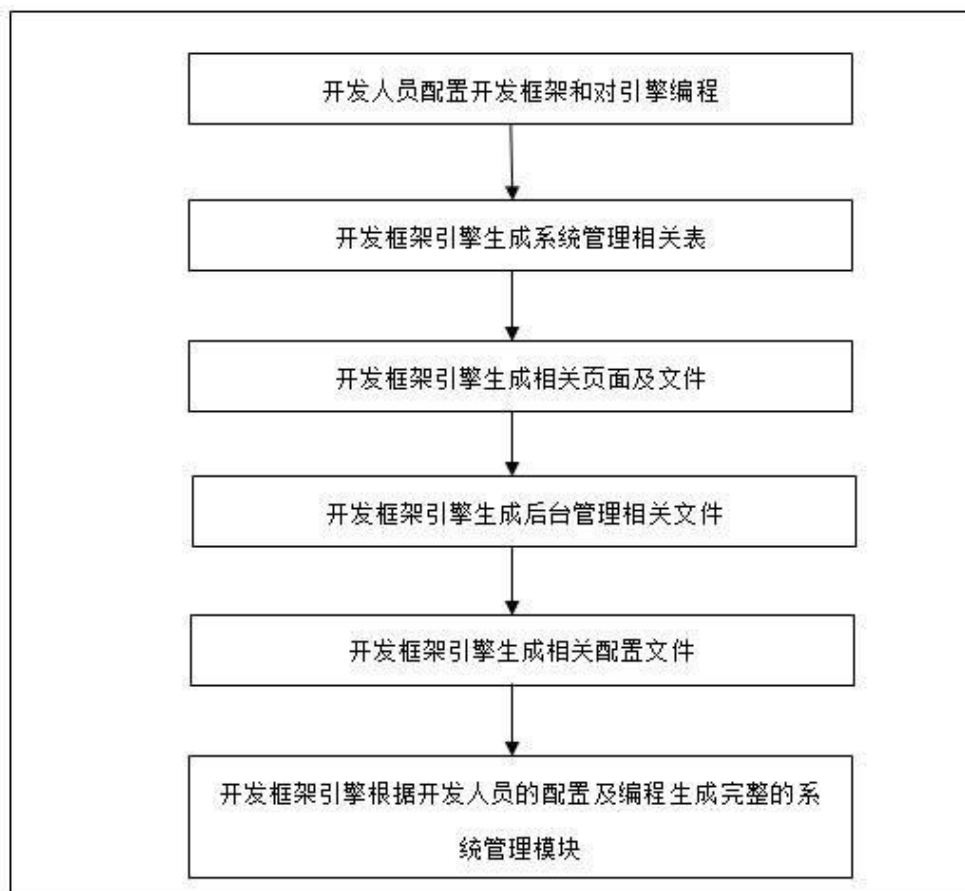
面向开发框架引擎开发软件过程:

本引擎调用到图 6 中的数据库引擎。

开发框架引擎中包括字典管理、单位管理、用户管理、角色管理、权限管理、菜单管理组件、底层公用函数库等，开发人员直接调用编程即可快速生成系统管理功能模块。



面向引擎的软件开发方法-开发框架引擎-图 7



面向引擎的软件开发方法-开发框架引擎解析生成系统管理模块步骤-图 7-1

1.开发人员配置开发框架和对引擎编程:

开发人员通过配置开发框架引擎和对引擎编程, 开发框架引擎解析生成相应的系统管理功能模块。

2.开发框架引擎生成系统管理相关表:

3.开发框架引擎生成相关页面及文件:

以下相关文件生成到开发人员的项目中。

ZRpowerWeb: Web 项目的相关 **web** 文件。

开发时需要加入的 **lib**:开发时需要加入的 **jar** 文件。

运行时所需的第三方 **jar** 文件。

4.开发框架引擎生成后台管理相关文件:

生成文件夹 **ZRpowerWeb-scr**: **Web** 项目的后台管理源代码文件。

5. 开发框架引擎生成相关配置文件:

生成文件夹 **Properties**:配置文件, 可以更改里面的文件设置属性。

6. 开发框架引擎根据开发人员的配置及编程生成完整的系统管理模块:

生成可以运行的系统管理功能模块, 开发人员可以在此基础上进一步深入开发系统管理功能。

面向自定义表单引擎开发软件过程:

开发人员调用自定义表单引擎执行以下步骤即可开发出一个数据录入模块。

后台会调用到图 6 中的数据库引擎。

增加数据录入模块基础配置:配置数据录入模块对应什么表和什么数据录入模板。

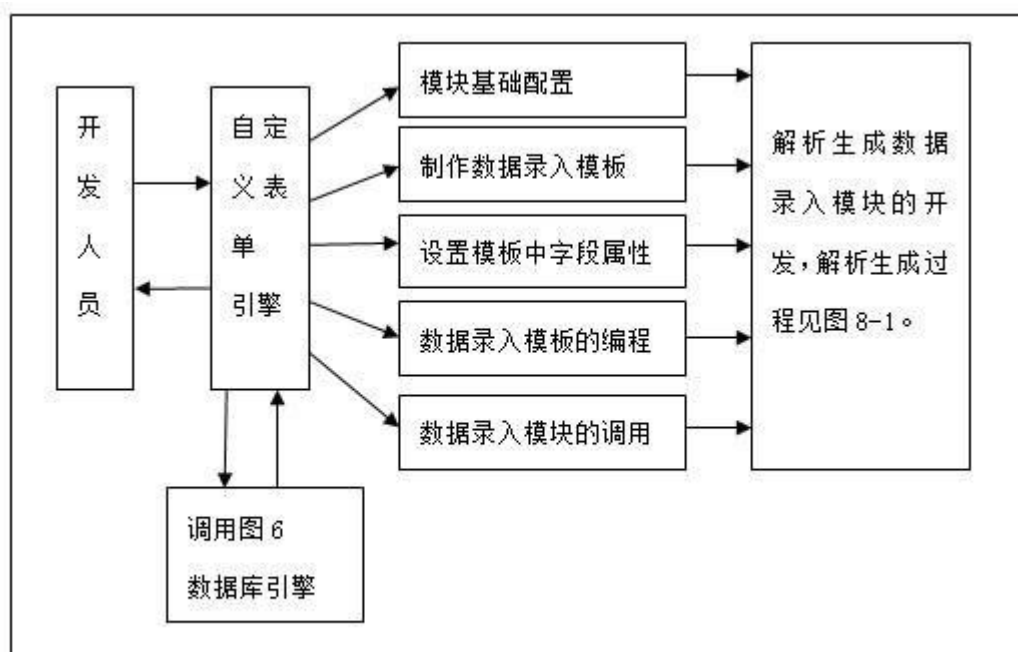
制作数据录入模板:用常用的静态网页工具制作的数据录入模板, 制作时标签名称与数据表字段名称保持一致。

设置模板中字段属性:设置各个录入项是输入框、选择框、单选、多选、是否是单位项、是否是人员项等属性。

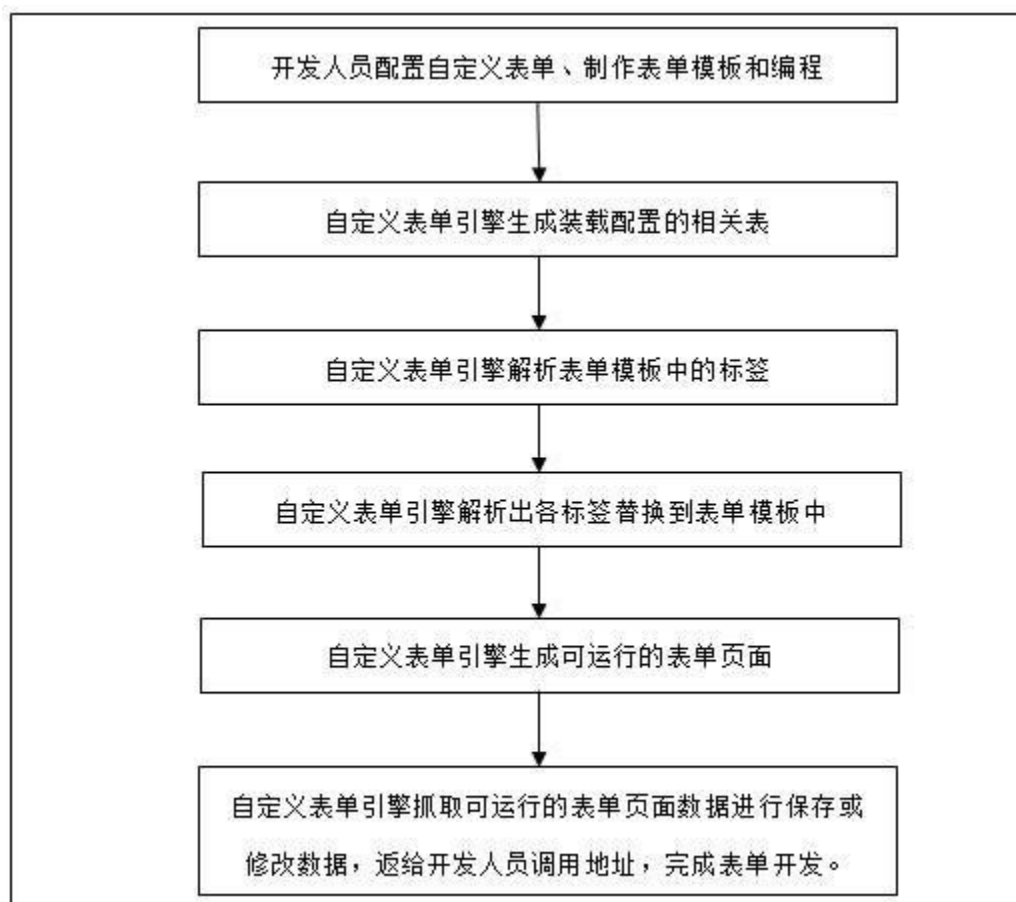
数据录入模板的编程:可以直接对自定义表单引擎进行编程, 后台会自动解析编程内容执行。

数据录入模块的调用:采用特定地址+ID 调用即可生成一个完整的数据录入模块。

采用面向引擎的软件开发方法开发数据录入模块, 非常简单和高效。



面向引擎的软件开发方法-自定义表单引擎-图8



面向引擎的软件开发方法-自定义表单引擎-解析生成数据录入模块步骤-图8-1

1. 开发人员配置自定义表单、制作表单模板和编程:

开发人员配置自定义表单生成需要的所有属性和自定义编程。

2. 自定义表单引擎生成装载配置的相关表:

自定义表单引擎生成装载配置的相关表, 为后台的表单解析提供支撑。

3. 自定义表单引擎解析表单模板中的标签

表单模板是标准的 **html** 文件模板, 支持标准 **html** 的所有技术, 表单引擎解析抓取表单模板中的标签内容, 根据配置表中的各种属性生成各种可运行的并且有值的表单中的标签代码。

4. 自定义表单引擎解析出各标签替换到表单模板中

解析生成的代码替换到原来模板对应的标签中, 生成像传统软件开发方式开发出来的页面。

5. 自定义表单引擎生成可运行的表单页面

模板解析完后自定义表单引擎进一步生成可运行的表单页面, 为后面数据保存或修改提供支撑。

6. 自定义表单引擎抓取可运行的表单页面数据进行保存或修改数据, 返给开发人员调用地址, 完成表单开发。

表单的最终目的是要能实现数据的录入保存或修改数据, 最后给开发人员返回可调用的地址, 通过特定地址调用集成到开发项目中, 完成各种表单的开发。

面向流程审批引擎开发软件过程:

开发人员调用流程审批引擎执行以下步骤即可开发出一个流程审批模块。

后台会调用到图 6 中的数据库引擎和图 8 中的自定义表单引擎。

设置流程基础配置: 设置流程的功能按钮和所执行的代码。

设置流程基本属性: 设置流程对应的数据录入模块、流程监控权限和流程审批权限。

设置流程步骤: 设置一个审批流程包含多少步, 每步审批的名称是什么。

设置流程步骤间的关系: 设置每一步审批同其它步骤间的关系是什么。

设置流程步骤的审批属性:包括流程步骤的基础属性、策略属性、超时属性、操作属性、可控字段属性、表单属性、附件属性、参考者属性等。

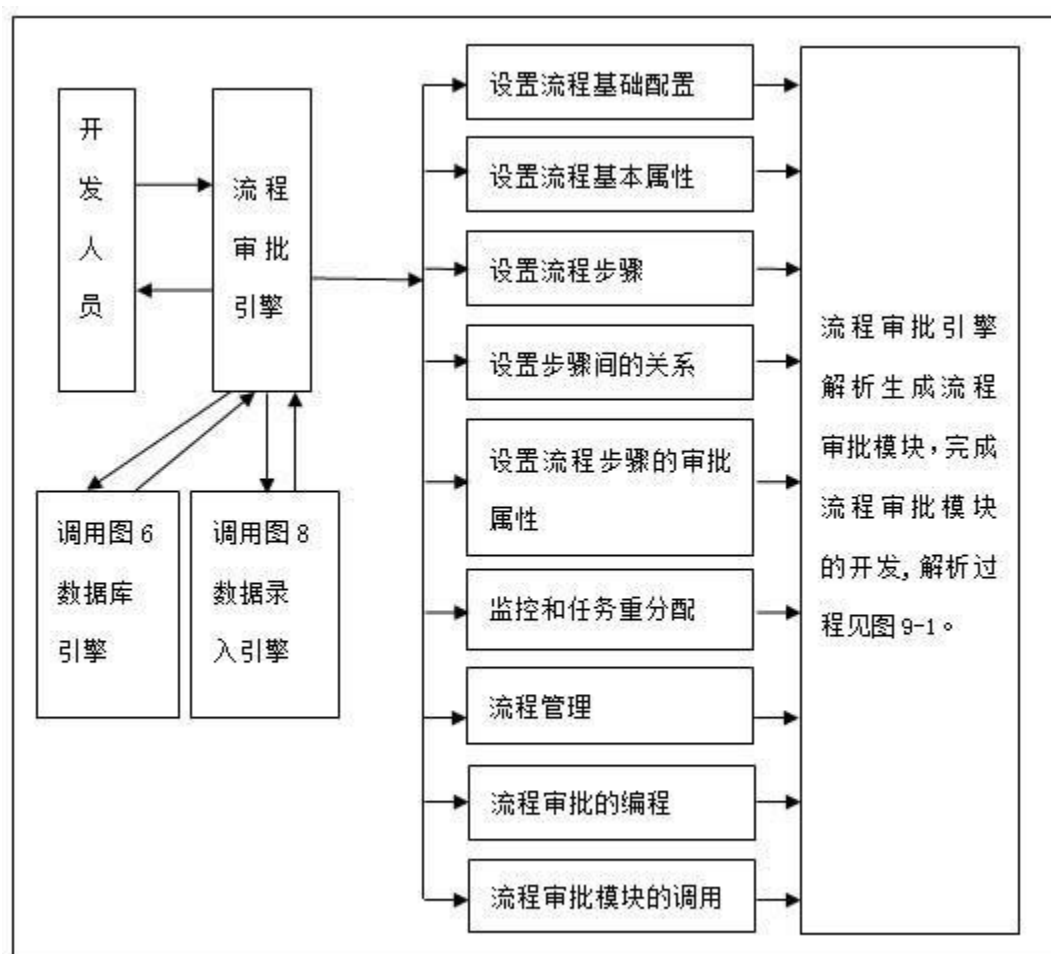
流程监控和任务重分配:执行过程中的流程可以进行监控和任务重分配。

流程管理:管理流程、更改流程、初始化流程、删除流程等。

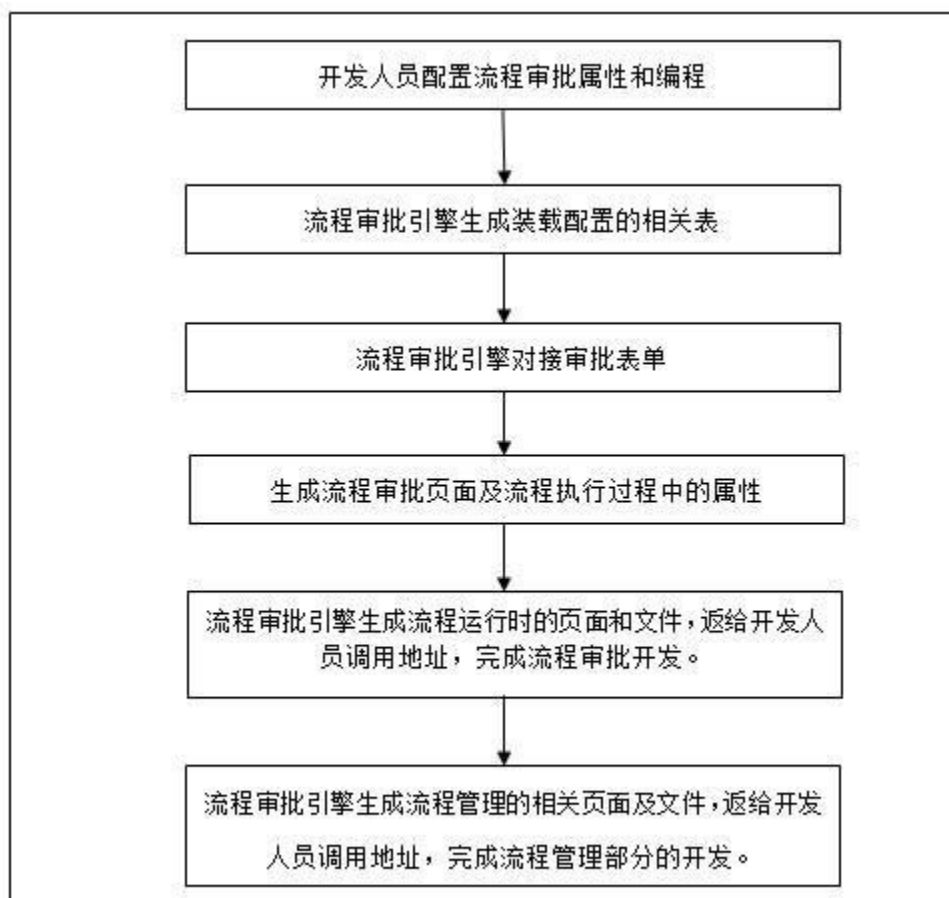
流程审批编程:针对流程审批特殊性进行编程。

流程审批模块的调用:采用特定地址+ID 调用即可生成一个完整的流程审批模块。

采用面向引擎的软件开发方法开发流程审批模块,非常简单和高效。



面向引擎的软件开发方法-流程审批引擎-图9



面向引擎的软件开发方法-流程审批引擎-解析生成流程审批模块步骤-图 9-1

1. 开发人员配置流程审批属性和编程:

开发人员配置流程审批生成需要的所有属性和自定义编程。

2. 流程审批引擎生成装载配置的相关表:

流程审批引擎生成装载配置的相关表, 为后台的流程审批解析提供支撑。

3. 流程审批引擎对接审批表单

流程引擎需要同自定义表单引擎对接, 让配置的表单提供给流程审批使用, 也就是流程的审批对象包含有审批表单的信息。

4. 生成流程审批页面及流程执行过程中的属性

生成流程审批页面及流程执行过程中的属性, 这个过程相当于传统软件开发方式经过长期复杂的开发才能实现的流程审批基本功能。

生成的流程审批模块在流程运行过程中根据以下各种属性来判断执行相应的流程操作。

5. 流程审批引擎生成流程运行时的页面和文件, 返给开发人员调用地址, 完成流程审批开发。

流程审批引擎生成流程运行时的页面和文件, 返给开发人员调用地址, 开发人员集成到项目中, 完成流程审批的开发。

6. 流程审批引擎生成流程管理的相关页面及文件, 返给开发人员调用地址, 完成流程管理部分的开发。

流程审批引擎生成流程管理的相关页面及文件, 返给开发人员调用地址, 开发人员集成到项目中, 完成流程管理部分的开发。

面向数据查询引擎开发软件过程:

开发人员调用数据查询引擎执行以下步骤即可开发出一个数据查询模块。

后台会调用到图 6 中的数据库引擎。

设置查询基础配置: 设置操作按钮和程序编码。

设置查询基本属性: 设置查询对应的主表及查询方式等。

设置查询关联表属性: 多表关联查询时配置。

设置初始条件属性: 设置数据的筛选条件。

设置查询列: 设置数据表中的哪些字段用作查询条件。

设置结果列: 查询出表格数据中有哪些字段。

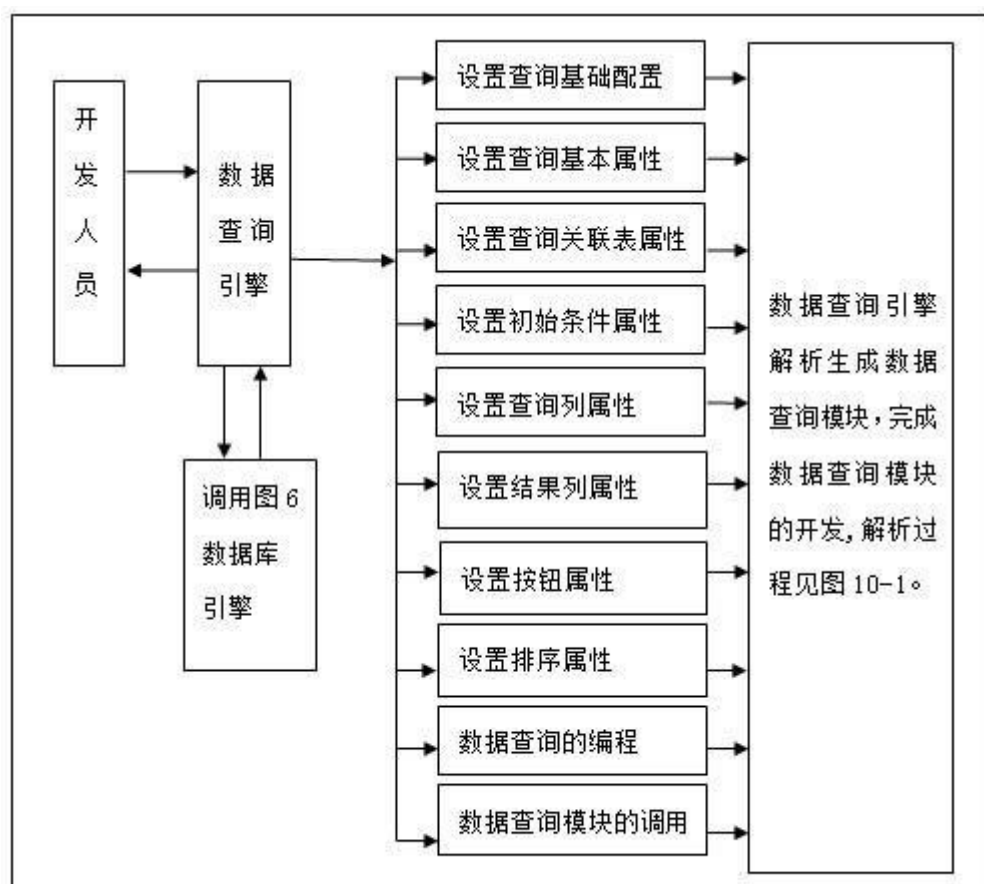
设置按钮属性: 设置对查询结果进行进一步的功能操作。

设置排序属性: 设置对查询结果按什么字段来排序展示。

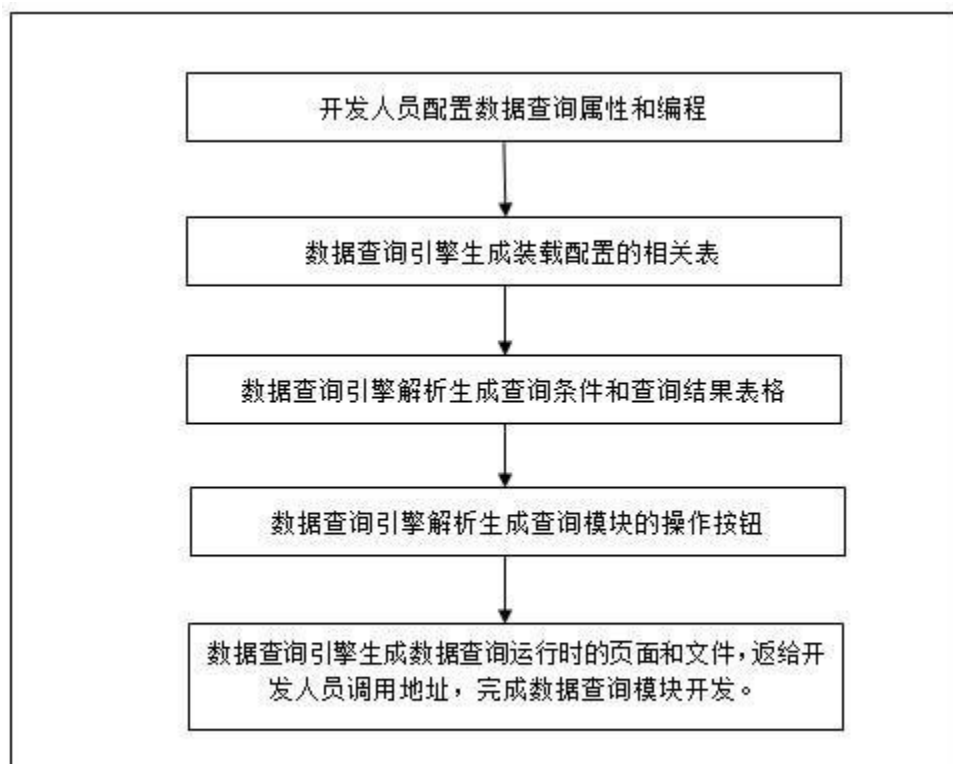
数据查询的编程: 针对数据查询特殊性进行编程。

数据查询模块的调用: 采用特定地址+ID 调用即可生成一个完整的数据查询模块。

采用面向引擎的软件开发方法开发数据查询模块, 非常简单和高效。



面向引擎的软件开发方法-数据查询引擎-图 10



面向引擎的软件开发方法-数据查询引擎-解析生成数据查询模块步骤-图 10-1

1. 开发人员配置数据查询属性和编程:

开发人员配置数据查询模块生成需要的所有属性和自定义编程。

2. 数据查询引擎生成装载配置的相关表:

数据查询引擎生成装载配置的相关表, 为后台的数据查询模块解析提供支撑。

3. 数据查询引擎解析生成查询条件和查询结果表格

查询引擎解析生成查询模块中的查询条件和查询结果表格。

查询条件中包括配置时定义的可用于查询的查询字段, 查询结果表格中包括结果的字段内容、顺序、对齐方式、宽度等各种定义。

4. 数据查询引擎解析生成查询模块的操作按钮

生成的数据查询模块中进一步扩展操作功能, 通过按钮的方式解析出来, 实现与其它引擎生成的模块互动或调用第三方技术。

5. 数据查询引擎生成数据查询运行时的页面和文件, 返给开发人员调用地址, 开发人员集成到项目中, 完成数据查询模块开发。

面向报表计算引擎开发软件过程:

开发人员调用报表计算引擎执行以下步骤即可开发出一个统计报表模块。

后台会调用到图 6 中的数据库引擎。

设置统计基础属性: 设置报表名称、用什么表来做统计、统计对应的模板等基础属性。

设置统计特殊属性: 设置自定义统计前选择的条件、自动计算及报表形式等。

制作统计报表模板: 定义统计出来的报表的展示风格和样式。

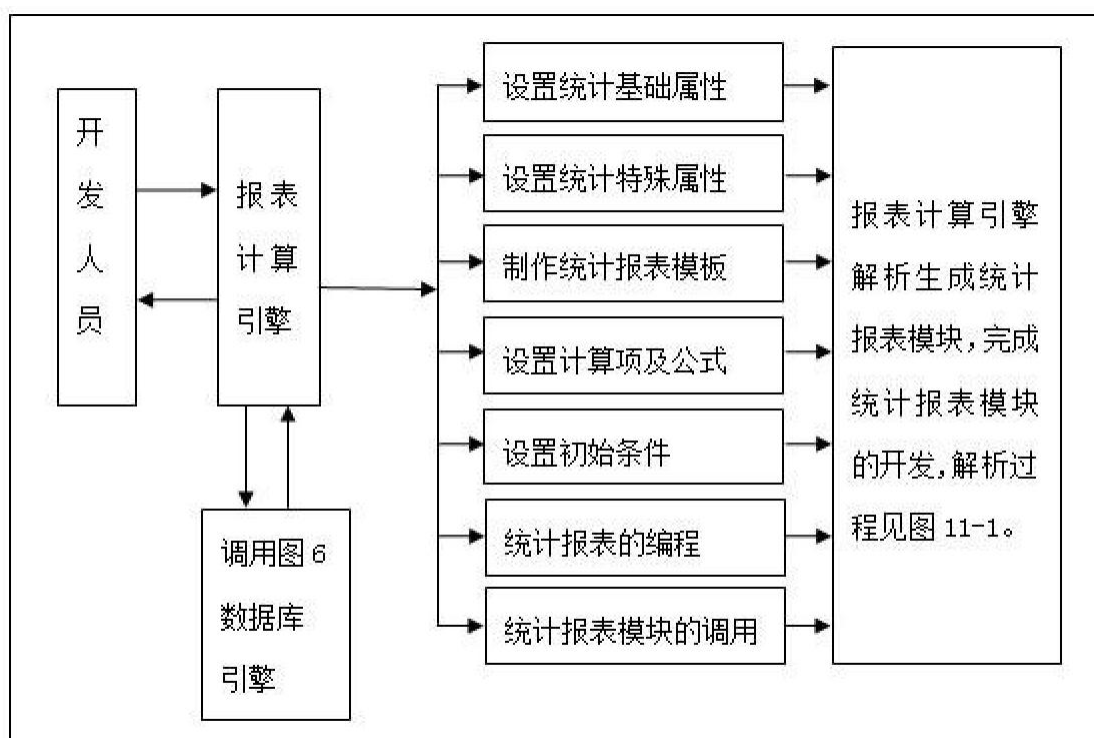
设置计算项及公式: 定义报表有哪些计算项及每个计算项的计算公式。

设置初始条件: 对统计的数据进行筛选。

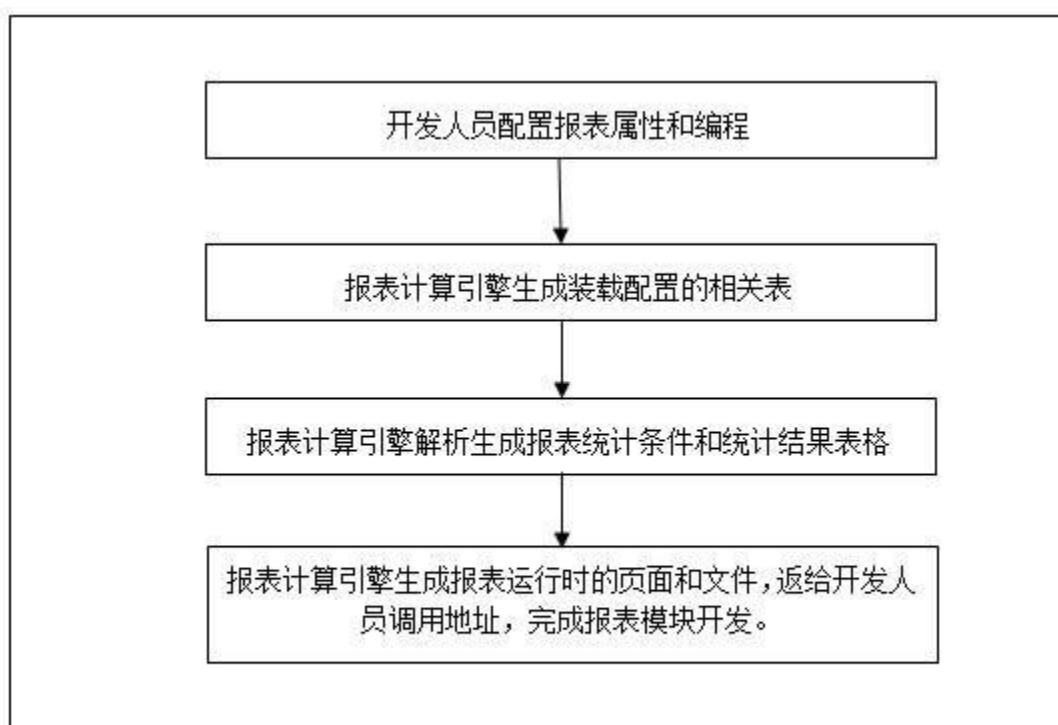
统计报表的编程: 针对统计报表特殊性进行编程。

统计报表模块的调用: 采用特定地址+ID 调用即可生成一个完整的统计报表模块。

采用面向引擎的软件开发方法开发统计报表模块, 非常简单和高效。



面向引擎的软件开发方法-报表计算引擎-图 11



面向引擎的软件开发方法-报表计算引擎-解析生成报表模块步骤-图 11-1

1. 开发人员配置报表属性和编程:

开发人员配置报表模块生成需要的所有属性和自定义编程。

2. 报表计算引擎生成装载配置的相关表:

报表计算引擎生成装载配置的相关表, 为后台的报表模块解析提供支撑。

3. 报表计算引擎解析生成报表统计条件和统计结果表格

报表计算引擎解析生成报表模块中的统计条件和统计结果表格。

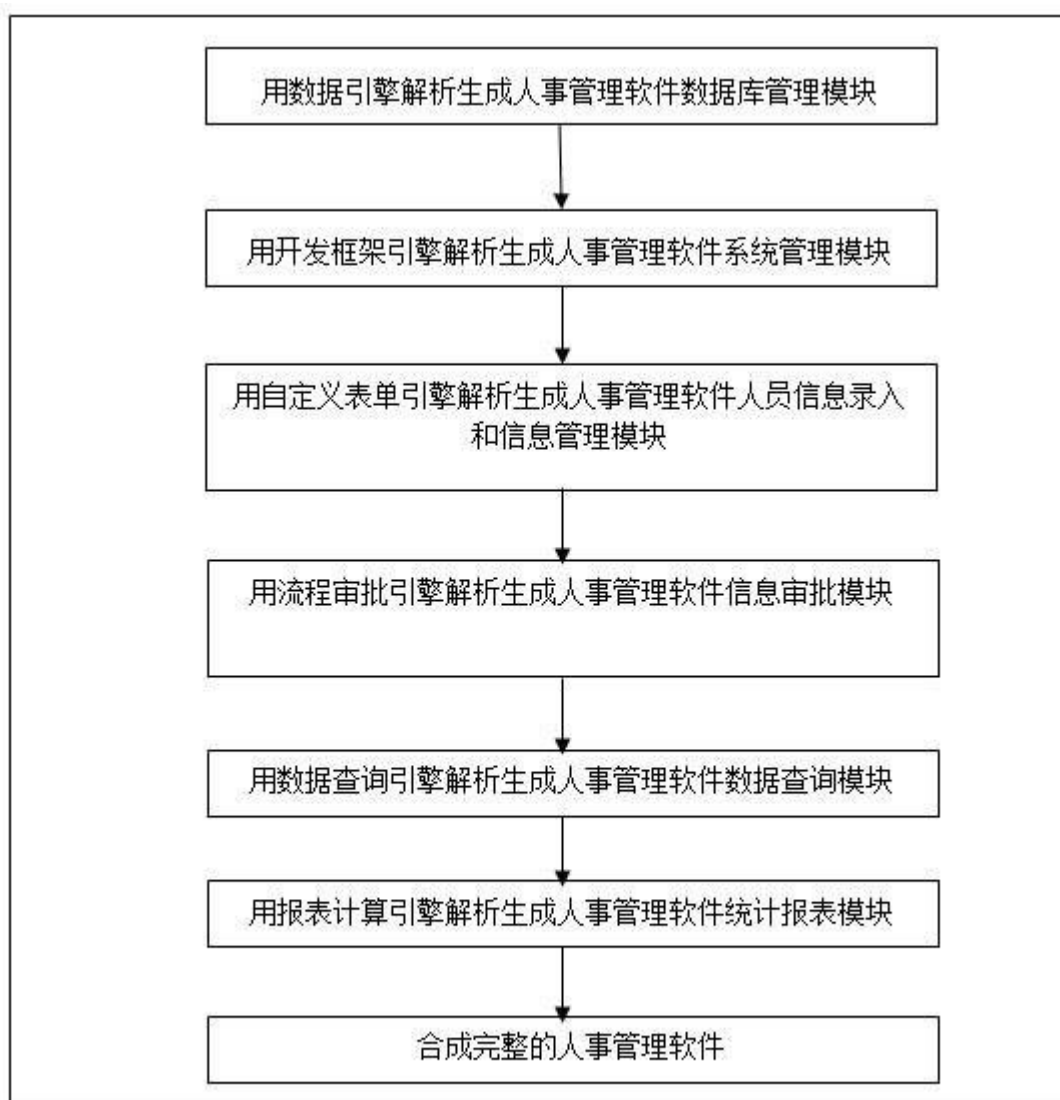
统计条件中包括配置时定义的可用于统计的统计条件, 报表结果根据配置的计算项目和公式自动计算生成报表的统计表格。

4. 报表计算引擎生成报表运行时的页面和文件, 返给开发人员调用地址, 完成报表模块开发。

报表计算引擎生成报表运行时的页面和文件, 返给开发人员调用地址, 开发人员集成到项目中, 完成报表模块开发。

总之, 管理软件开发中的各个方面都可以定义成不同的引擎, 开发人员只需面向引擎配置和编程即可更简单、更高效、更高质量开发各种管理软件功能模块。

前面的面向引擎的软件开发方法开发过程描述比较抽象, 以下以开发一个人事管理系统为例, 运用上面各引擎开发过程最终实现软件开发的效果图来更形象地分析各引擎解析效果, 这样能更容易理解面向引擎的软件开发方法。



面向引擎的软件开发方法-开发人事管理软件过程-图 12

数据库引擎最终要的实现效果图:

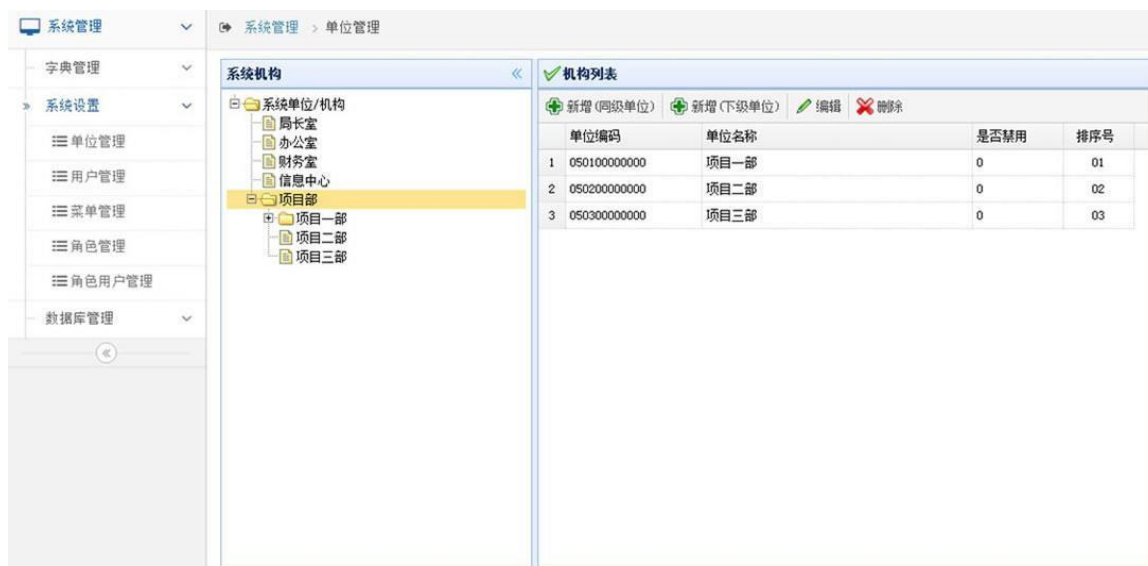
ExecuteDelete (java.lang.String TableName, java.lang.String Condition)
删除表中的记录
ExecuteEdit (xr.zrpower.common.db.dbRow dbrow)
编辑记录保存到数据库
调用前提: 必须设置主键和主键值
ExecuteEdit (xr.zrpower.common.db.dbRow dbrow, java.lang.String Condition)
编辑记录保存到数据库
ExecuteInsert (xr.zrpower.common.db.dbRow dbrow)
插入记录到数据库
ExecuteSQL (java.lang.String SQL)
执行单SQL语句操作
ExecuteSQL (java.lang.String SQL, java.lang.String[] value)
执行单SQL语句操作, 类似Hibernate执行sql语句的参数传递。
ExecuteSQLs (java.lang.String[] SQLs)
执行多SQL语句操作
ExecuteSQLs (java.lang.String[] SQLs, java.lang.ArrayList value)
执行多SQL语句操作, 类似Hibernate执行sql语句的参数传递。
ExecuteSQLs (java.lang.String SQLs, java.lang.String[][] value)
执行多SQL语句操作, 类似Hibernate执行sql语句的参数传递。
initialize ()
初始化函数
QueryColumn (java.lang.String TableName, java.lang.String[] ColumnName)
数据库查询操作
QueryCondition (java.lang.String TableName, java.lang.String[] ColumnName, java.lang.String Condition)
数据库查询操作
QuerySQL (java.lang.String strSQL)
执行查询 (多态2)
QuerySQL (java.lang.String strSQL, java.lang.String TableName)
执行查询 (多态1)
QueryTable (java.lang.String TableName)
查询表
UpdateAll (xr.zrpower.common.db.dbRow[] dArr)
更新多个数据行
UpdateAll (xr.zrpower.common.db.dbRow[] dArr, java.lang.String[] strBeforeSql, java.lang.String[] strAfterSql)



字段ID	字段名	中文名称	字段类型	字段长度	是否可空	特殊属性	扩展属性	对应字典表	对应字段1或	对应字段2	对应字段3	描述
1	0000000 ID	ID	VARCHAR2	23								
2	0000000 PCODE	编号	VARCHAR2	50								
3	0000000 NAME	姓名	VARCHAR2	50								
4	0000000 DEPT	部门	VARCHAR2	12								
5	0000000 XB	性别	VARCHAR2	6								
6	0000000 SPELL	姓名简拼	VARCHAR2	50								
7	0000000 NL	年龄	NUMBER	3								
8	0000000 ZW	职务	VARCHAR2	3								
9	0000000 GS	归属	VARCHAR2	3								
10	0000001 CODEORDER	排序号	VARCHAR2	3								
11	0000001 FGH	工号	VARCHAR2	5								
12	0000001 JG	籍贯	VARCHAR2	200								
13	0000001 FKSZD	户口所在地	VARCHAR2	300								
14	0000001 XL	第一学历	VARCHAR2	6								
15	0000001 ZC	职称	VARCHAR2	50								
16	0000001 BLYS	毕业院校	VARCHAR2	200								
17	0000001 ZL	专业	VARCHAR2	100								
18	0000001 CBZST	参加工作日期	DATE	0								

编辑
字段名: ZW
中文名称: 职务
字段类型: VARCHAR2
字段长度: 3
是否可空: ☒ 是 ☐ 否
是否主键: ☒ 是 ☐ 否
默认值:
查询关联字段:
说明: 多个字段间用、号分隔; 在使用查询引擎时, 查询当前字段一起将关联字段中的符合条件的记录查询出来。
描述:
特殊属性: 代码表标记
扩展属性: 下拉控件
对应字典表: 职务字典表

开发框架引擎解析生成的人事管理系统的系统管理模块效果图:



系统管理
字典管理
系统设置
单位管理
用户管理
菜单管理
角色管理
角色用户管理
数据库管理

系统机构
系统单位/机构
局长室
办公室
财务室
信息中心
项目部
项目一部
项目二部
项目三部

机构列表
新增(同级单位) 新增(下级单位) 编辑 删除

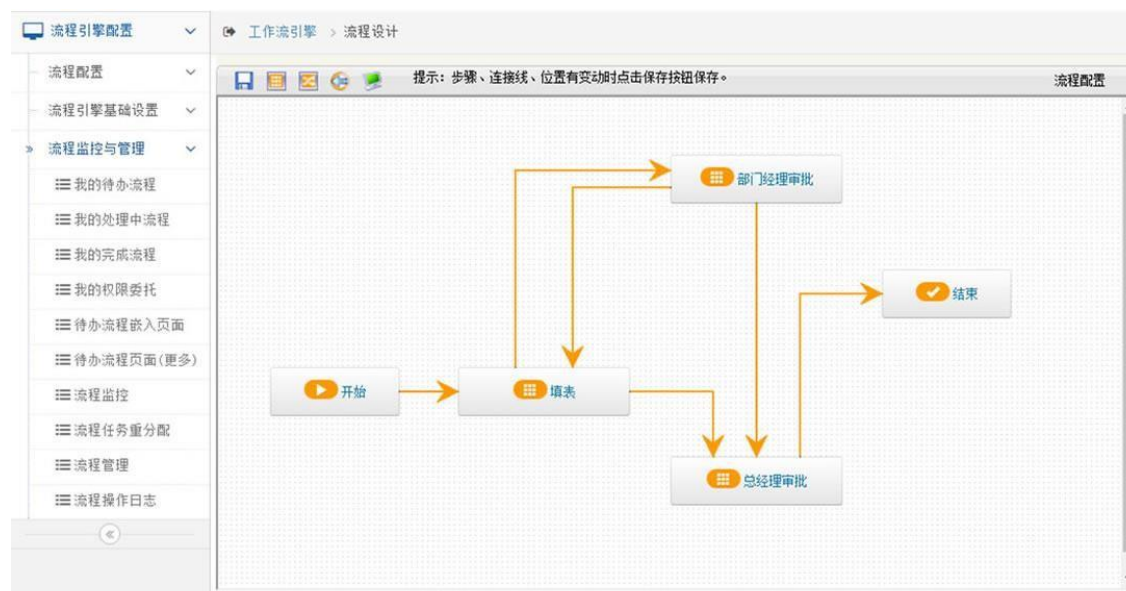
单位编码	单位名称	是否禁用	排序号
1 050100000000	项目一部	0	01
2 050200000000	项目二部	0	02
3 050300000000	项目三部	0	03

自定义表单引擎解析生成的人事数据录入模块效果图:

✓ 人事档案管理

姓名:	徐谢垠		
工号:	00187		
部门:	办公室		
性别:	男		
出生日期:	1988-08-08		
照片上传:	选择文件 (gif/jpeg格式, 最大 500 k)		
姓名简拼:	xk	归属:	综合柜员
身份证号码:	52010119885523111x	民族:	汉族
婚姻状况:		政治面貌:	群众
籍贯:		户口所在地:	
第一学历:	本科	最高学历:	
毕业院校:	第二炮兵工程学院	毕业时间:	
参加工作时间:		进入本单位时间:	
用工类型:		职务:	

流程审批引擎解析生成的人事流程审批模块效果图:



数据引擎解析生成的人事数据查询模块效果图:

部门:

性别:

归属:

毕业院校:

专业:

是否在职教育:

姓名:

年龄, 从: 至

工号:

参加工作时间, 从:

用工类型:

身份证号码:

职务:

职称:

到:

手机:

查询

查看

导出Excel

显示/隐藏查询

工号	姓名	部门	性别	年龄	职务	归属	第一学历	职称	毕业院校	专
2 00001	张林	办公室	男		理事长	领导班子	研究生	会计师		
3 00003	李琴	办公室	女		监事长	领导班子	本科	助理政工师		
4 00004	陈敏	办公室	女		副主任	领导班子	本科	助理会计师		
5 00009	许敏	办公室	女		副主任	领导班子	本科			
6 00006	刘庆雷	办公室	男		主任助理	领导班子	研究生			
7 00005	王国恒	办公室	男		主任助理	领导班子	本科	会计员		
8 00007	杨五婷	办公室	女		主任助理	领导班子	本科			
9 00008	李小炫	办公室	男		主任助理	领导班子	大专			
10 00011	敬晓洪	办公室	男		主任	中层干部(部门)			中共中央党校	注
11 00012	吴刘	办公室	男		综合柜员	工作人员	中专		空军第一工程学校、贵州大学	牙
12 00014	邓干美	办公室	女		综合柜员	工作人员	大专	会计员	自学考试	行

28

第 1 页 共 7 页

当前显示: 1到28 条记录 共 185 条记录

报表计算引擎解析生成的人事数据统计报表模块效果图:

导出Excel

学历	人数	年龄					
		22岁以下	22-25岁	26-30岁	31-40岁	41-50岁	51岁以上
小学	0	0	0	0	0	0	0
初中	0	0	0	0	0	0	0
高中	10	0	0	0	5	5	0
中专	24	0	0	4	10	8	2
大专	41	1	1	5	24	10	0
本科	91	1	22	32	24	12	1
研究生	4	0	1	0	2	1	0
硕士	1	0	0	0	1	0	0
博士	0	0	0	0	0	0	0
博士后	0	0	0	0	0	0	0
大学	3	0	0	0	2	1	0
其它	0	0	0	0	0	0	0
合 计	185	2	24	42	73	42	3

应用效果:

采用面向引擎的软件开发方法来开发软件, 将某些特定功能的模块定义成不同的引擎, 软件开发时只需对这些引擎进行配置、制作模板、面向引擎编程、引擎能快速解析生成软件, 提高软件开发效率和质量, 降低软件开发成本。