

# 组件设计文档

北大 Mozart

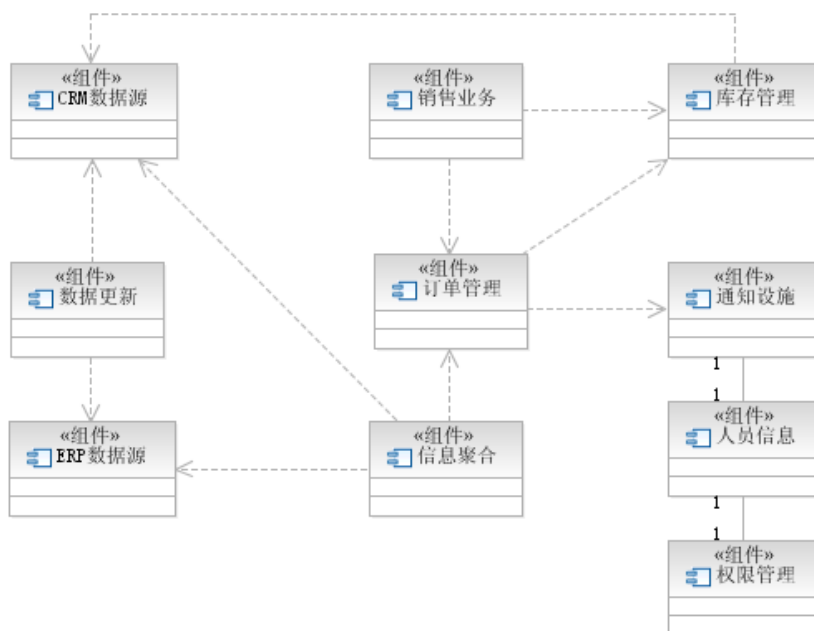
# 凤凰公司整合项目组件设计

**【文档介绍】** 组件设计是SOA方法中的重要开发步骤。它建立在服务建模和系统架构设计的基础之上，对服务建模和系统架构进一步精化，为下一步建立设计实施计划提供蓝图。本文档描述了凤凰公司整合项目中一种可行的组件设计方案，对该方案中每一个组件从多个方面给出了详细的说明。

## 1 系统总体功能模块划分

### 1.1 顶层结构图

根据服务建模和架构设计的分析，可以将目标的集成系统划分成若干模块，这些模块的顶层结构图如下：



上图中，**数据更新组件、信息聚合组件**全部由万合公司开发；而**订单管理组件和销售业务组件**由万合在原有组件功能基础上提供扩展，其它为包装系统已有的接口。

下面在组件描述部分中分别叙述本图中各个组件功能(人员管理和权限管理由 Turbo CRM 和 NC 系统提供，具体描述细节可以从相关文档中得到，在此不再赘述)。

## 2 组件功能描述

本节对上节提到的组件作进一步规约，对于组件的描述内容包括以下几个方面：

- 功能描述
- 组件内部结构
- 组件规约

- 接口定义
- 非功能属性描述
- 实现机制分析
- 数据流程

## 2.1 销售业务组件

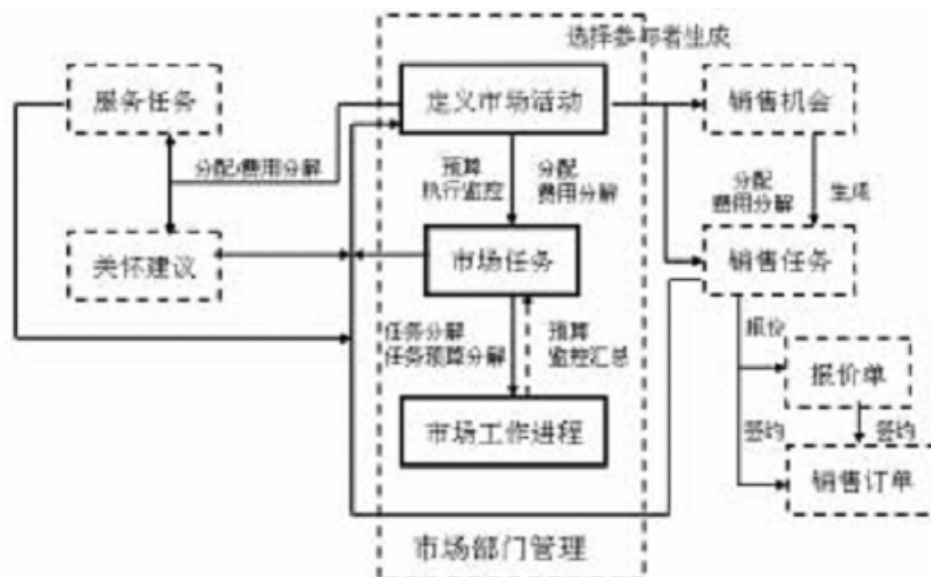
### 2.1.1 功能描述

原有销售业务组件的功能见 Turbo CRM 白皮书。

万合公司在 Turbo CRM 开发的销售业务组件的功能上增加了“生成订单请求”的功能。该功能根据 CRM 系统中的“订单”的内容，生成与 ERP 系统中“预订单”格式、内容一致的“订单请求”，并将此“订单请求”发送给 ERP 系统。

### 2.1.2 组件内部结构

Turbo CRM 白皮书中描述的“销售业务组件”的系统任务如下图。万合公司开发的扩展服务是在原有的订单流转流程的末端加入一个服务，用于将销售订单转化为“订单请求”的格式。



### 2.1.3 组件规约

#### 规则

- 使用本组件的人员必须为 CRM 系统注册人员。
- 使用本组件新增功能的人员必须为 CRM 系统注册人员。

#### 服务

- 生成订单请求

#### 属性

- CRM 系统中的订单号      Integer

|                      |         |
|----------------------|---------|
| ■ 时间                 | Date    |
| ■ CRM 系统中的订单内容（复合属性） |         |
| ◆ 名称                 | String  |
| ◆ 销售人员 ID            | Integer |
| ◆ 客户 ID              | Integer |
| ◆ 产品列表（复合属性）         |         |
| ● 产品 ID              | Integer |
| ● 数量                 | Integer |

## 使用组件

### ■ 订单管理组件

#### 2.1.4 接口定义

##### ■ 生成订单请求

入口参数：CRM 系统中的销售订单代号

返回值：成功或失败代码

异常：订单代号不存在

#### 2.1.5 实现机制分析

由于新增功能是在 Turbo CRM 的原有功能上的扩展，因此直接新开发一个“生成订单请求”的 JSP 页面即可，该页面上有增加订单请求的按钮，点击此按钮可以调用一个 Web 服务。由这个 Web 服务完成全部功能。

由于具体的业务过程与下一节的“订单管理组件”有很强的依赖关系，因此统一在“订单管理组件”中详细叙述。

## 2.2 订单管理组件

### 2.2.1 功能描述

订单管理组件完成对订单信息整合的管理，即完成 CRM 中的订单管理流程与 ERP 系统中的订单管理流程的整合。在上节（销售业务组件）中，我们描述了 Turbo CRM 中的销售业务组件的内部结构，并给出了完成“生成订单请求”服务的方案。

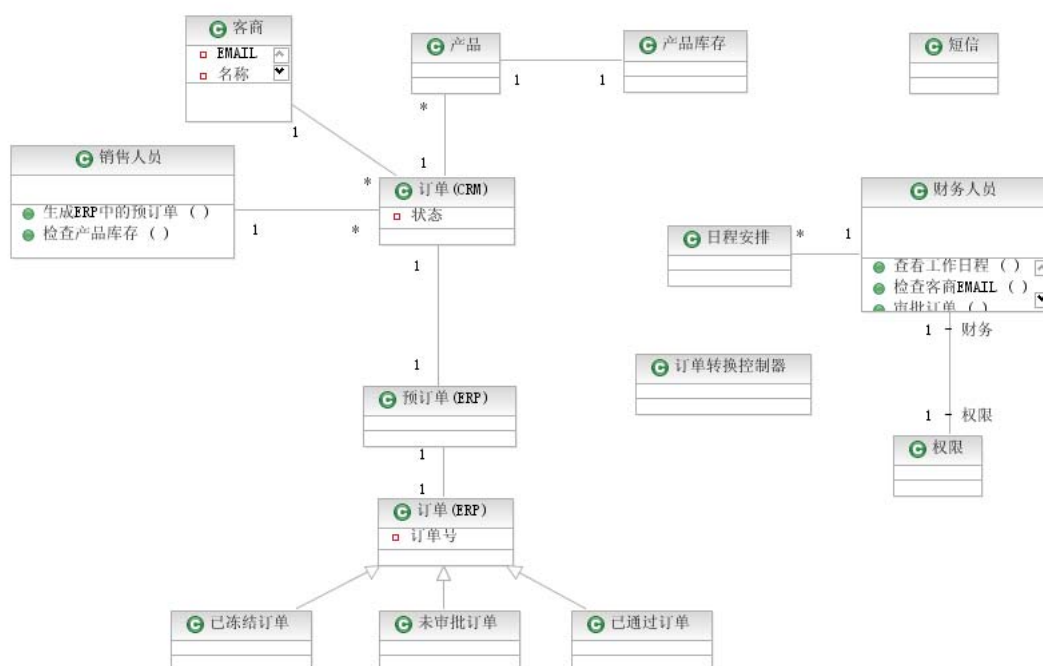
而 NC ERP 系统中的订单管理包括订单（预订单）的新增、修改、审批、修订等一系列功能。根据对两套系统的工作流程的深入分析及其我们前述文档对流程、服务的分析，我们在“订单管理组件”上需要新增的功能包括：

- 维护 CRM 中“销售订单”与 ERP 系统中的“预订单”之间的一一对应关系。
- 在工作日历组件中增加“记录待审批订单请求”的功能，帮助财务人员在第一时间发现新增的、需要审批的订单请求。
- 在 CRM 系统和 ERP 系统中保持订单状态在一定程度上的一致性，即：当财务人员改变订单状态（如审批、冻结等）时，此信息能及时反应到 CRM 系统中。

- 增加一个发送短信功能。在销售人员提交一个订单请求时，可以及时以短信方式通知财务人员。同样，在财务人员改变订单状态时，可以即时以短信方式通知销售人员。

### 2.2.2 组件内部结构

我们以类图形式描述订单管理组件的内部结构。对于这些类的属性和方法，我们仅仅给出了目前我们在系统整合中较为关注的部分。其中类名后标明 CRM 或 ERP 的表示目前已经存在于 CRM 系统或 ERP 系统中、且需要加以区分的类。



### 2.2.3 组件规约

#### 提供的服务

- 提交订单请求
- 通过订单请求
- 退回订单请求

#### 属性

- CRM 系统中的订单内容（复合属性）
  - ◆ 名称 String
  - ◆ 销售人员 ID Integer
  - ◆ 客户 ID Integer
  - ◆ 产品列表（复合属性）
    - 产品 ID Integer
    - 数量 Integer
    - 邮件 String
    - 短信 String
  - ◆ 日程安排（复合属性）

- 提交日期      Date
- 截止日期      Date
- 链接            String
- 对象            String
- ◆ 产品库存信息（复合属性）
  - 产品 ID            Integer
  - 产品名称            String
  - 数量            Integer

### 使用组件

- 库存管理
- 邮件系统
- 短信平台
- 日程管理

### 变化点

- [可配置] 订单相关通知方式，比如将“发送邮件”替换为“发送短信”
- [可配置] 处理订单请求的流程

## 2.2.4 接口定义

根据在服务分析中描述的接口，我们将接口及其细化之后的定义描述如下：

### 提交销售订单请求

- 入口参数： 销售订单代码，销售订单内容
- 返回值： 订单请求提交结果代码
- 异常： 数据库错误等提交失败信息

### 查询销售订单请求审批结果

- 入口参数： 销售订单代号
- 返回值： 审批结果
- 异常： 订单号不存在

### 冻结销售订单请求

- 入口参数： 销售订单代号
- 返回值： 错误代码
- 异常：

### 批准销售订单请求

- 入口参数： 销售订单代号
- 返回值： 错误代码
- 异常信息：

## CRM 系统暴露的销售订单信息相关接口（见 NC ERP 白皮书）

获取销售订单

新增销售订单  
更改销售订单  
删除销售订单

### 2.2.5 非功能属性描述

要求 CRM 和 ERP 之间的订单状态同步时间不超过 1 天。

从 CRM 提交订单请求到 ERP 的延迟时间不超过 1 天。

### 2.2.6 实现机制分析

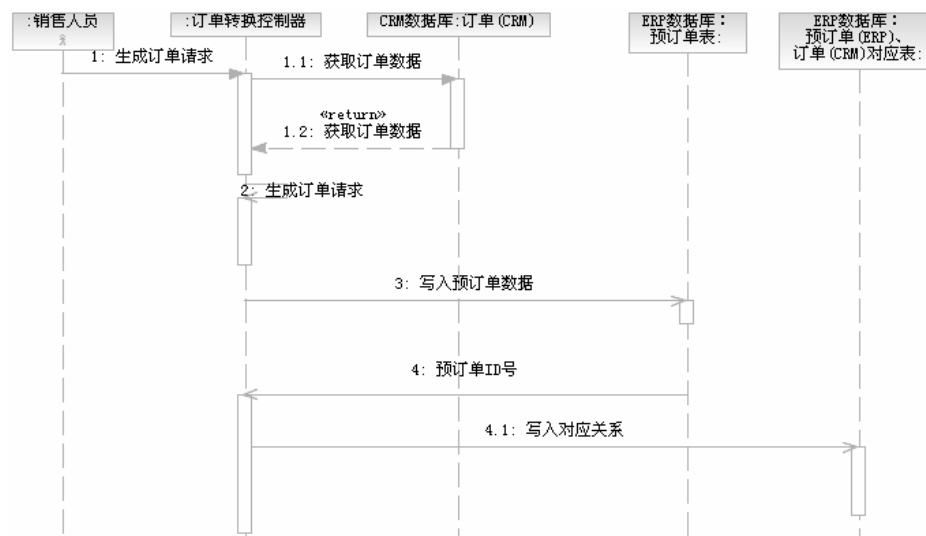
本模块实现上的难点有两点：第一是如何保持 CRM 中生成的“订单请求”与 ERP 中的“预订单”之间的一一对应关系，第二是如何降低单据同步的延迟，满足上面提到的非功能性需求。以下分别加以叙述。（注：其余的关于日历组件、短信平台中的问题，在其相关部分描述）

#### 1. 保持 CRM 中生成的订单请求与 ERP 中的“订单请求”之间的一一对应关系

由于 ERP 系统在新增订单请求（预订单）时会自动产生一个序列号——而不能为 ERP 系统内的订单请求预先分配一个唯一的代号，因此只能在 ERP 系统产生了订单请求之后，才能建立其对应关系。

我们采取的策略是在 ERP 系统内部数据库中增加一张对应表，这张表用于记录 CRM 产生的订单请求号与 ERP 中的预订单号之间的对应关系。

整个操作流程如下：



#### 2. CRM 与 ERP 中的即时信息同步。

需要信息同步的关键点在于，如果在财务人员处理了一条订单请求之后，能够即时的通知 CRM 系统（其它的部分我们可以以手工编码的方式保证他们及时同步）。

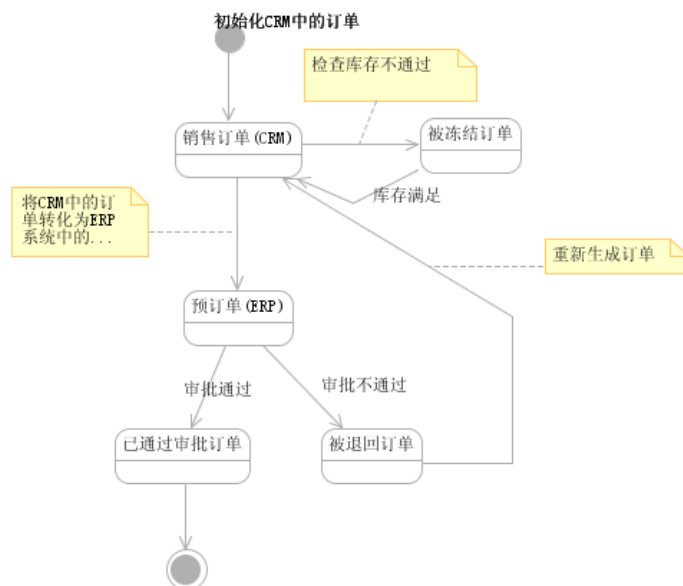
我们的策略包括以下三个部分：

- 使用数据库的触发机制来监视是否产生了数据变化
- 当数据发生变化时先将变化写入临时表
- 一定间隔时间同步一次。

在这样的机制保证下，我们保证了每条订单审批信息及其结果都可以被及时的被销售人员感知，并在一定时间间隔下被反馈到 CRM 数据库。同时，这样的策略考虑了及时性与系统性能（频繁的数据导入、导出会导致系统性能下降）的权衡。

### 2.2.7 数据流程

订单管理组件的核心就是订单在该组件中的流转过程，我们下面给出订单在该组件中的状态变换过程。



## 2.3 数据更新组件

### 2.3.1 功能描述

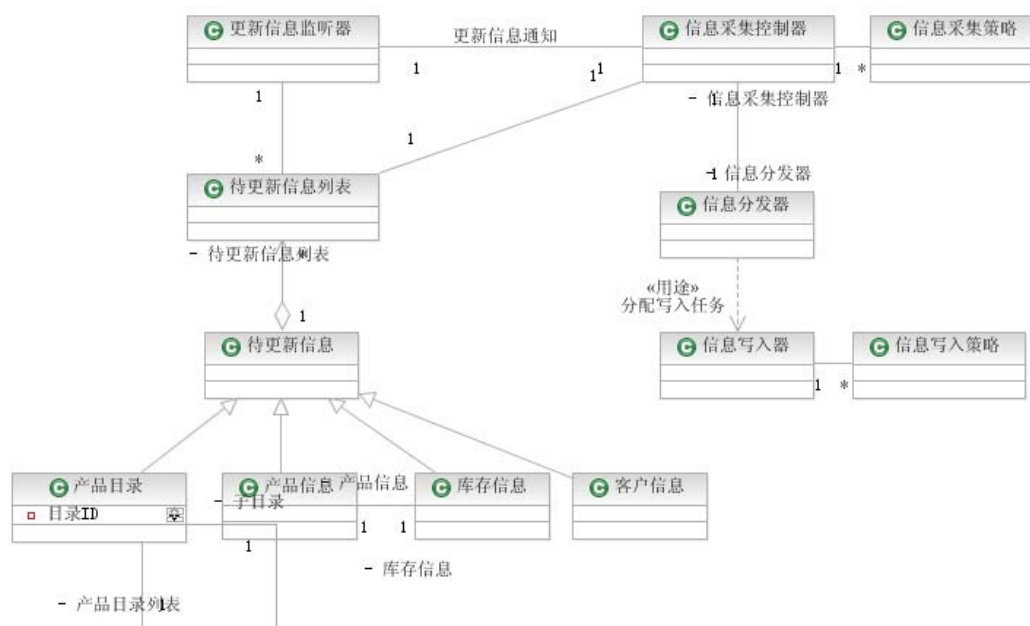
数据更新组件主要完成 ERP 和 CRM 系统的数据同步工作。需要同步的数据包括：产品目录、产品信息、库存信息、产品价格信息和客户信息等。

数据更新组件提供可定制的数据更新策略。用户可以根据数据不同的数据性质定义不同的更新策略。例如对于产品目录，可以采用 1 天更新一次的策略；对于库存信息，可以采用触发器监控更新的策略。

同样，对于信息分发，也可以由用户定义写入策略。写入策略包括即时写入、读失败时写入、时间段写入等等。



## 2.3.2 组件内部结构



从上图我们可以看出，信息采集控制器根据定制的信息采集策略采集数据。采集到信息之后，使用信息分发器将数据写入不同目标数据存储地。

## 2.3.3 组件规约

### 服务

- 产品目录信息定时更新
- 产品信息定时更新
- 产品价格信息实时更新
- 产品库存信息实时更新
- 订单信息触发更新
- 客户信息触发更新
- 启动实时更新服务

### 属性

- 数据类型 枚举（产品目录/产品信息/产品价格/产品库存/客户信息/订单信息）
- 日期 Date

### 使用组件

- ERP 信息源
- CRM 信息源

### 变化点

[可配置] 可配置策略的信息采集

[可配置] 可配置策略的数据分发

### 2.3.4 接口定义

#### 启动数据实时更新服务

输入：数据类型

输出：成功失败代码

#### 信息读取器

##### ■ 查询产品分类/产品/价格信息

输入：数据类型，起始日期，终止日期

输出：全部产品分类/产品信息

##### ■ 查询产品库存信息

输入：起始日期，终止日期，包含产品属性信息的查询条件

输出：输出所有满足条件的全部库存信息

#### 信息写入器

##### ■ 产品分类/产品/价格信息

输入：数据类型，需要写入的数据结合

输出：成功/失败代码

##### ■ 查询产品库存信息

输入：包含产品属性信息的查询条件

输出：成功/失败代码

### 2.3.5 非功能属性描述

若定制的策略为即时更新，则需要能保证更新的即时性。

若定制策略为定时更新，则每次更新时间可配置，配置设置为 1 天。

### 2.3.6 实现机制分析

根据不同的更新策略，采用不同的实现机制。以下分别讨论。

#### 实时更新

采用数据库提供的触发器机制，保证在数据库发生更新时即时通知数据更新控制器。

#### 定时更新

更新信息监听器每隔一段时间（用户定制）监听数据库是否发生更新。若更新发生则通知

同样，根据不同的数据写入策略，也采用用户定义的机制实现。

#### 即时写入

在数据更新的同时写入。

#### 延迟+读失败 写入

将数据缓冲，在需要的时候才写入数据。

#### 延迟+定时 写入

每隔一段时间将数据写入数据库中。

## 2.4 信息聚合组件(扩展需求)

### 2.4.1 功能描述

在需求分析中,我们已经概述了该模块的基本需求:即集成这些来自不同数据源的信息,并以直观、清晰的方式提供给需要的企业人员。

我们的信息聚合方案试图对以下问题给出初步的解决手段:

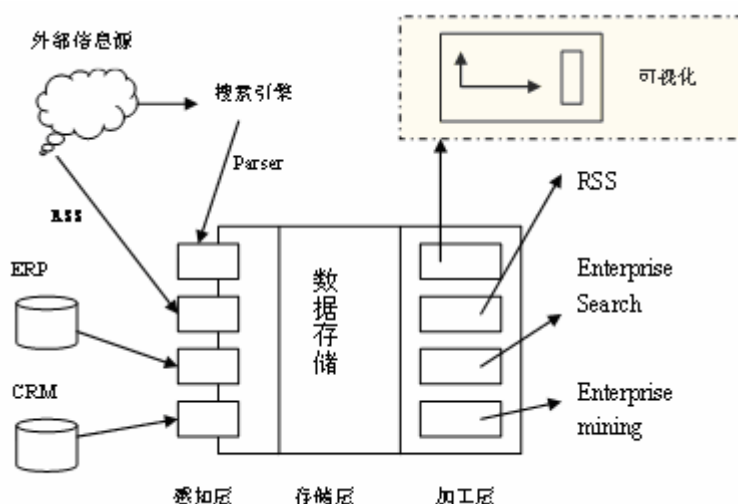
- 从不同的异构数据源（包括不同的企业内部系统以及企业外部信息源）中集成所需的信息，并保证数据的完整性、实时性。
- 支持基于角色的数据访问控制机制，为不同的企业人员（决策者、销售人员、财务人员）提供不同的数据集；
- 支持用户对显示数据项的定制：对于自己所能访问的数据集，用户可以按需选择显示哪些数据项，从而定制自己需要的分析视图。

### 2.4.2 组件内部结构

由于该组件结构较为复杂，我们将从至上而下的不同层次展现该组件的设计方案。

#### ■ 信息聚合组件概览

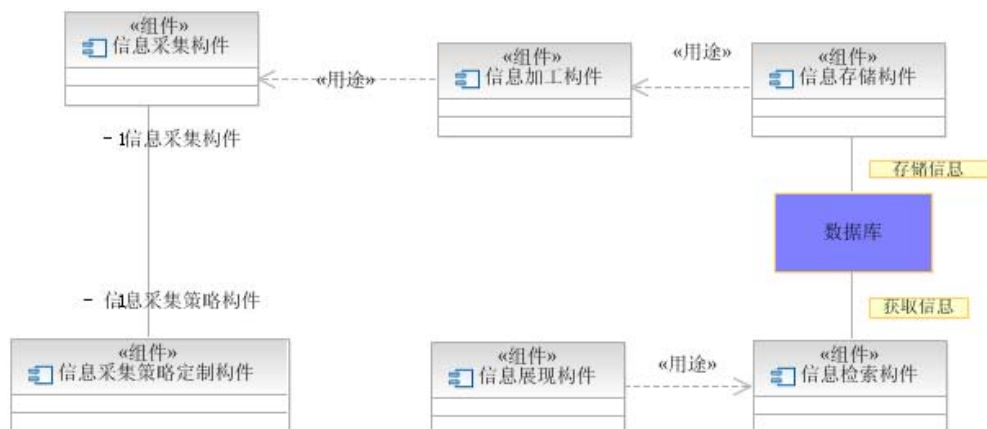
在相关整合架构上，我们将整合系统分为“感知层”、“存储层”和“加工层”。如下图所示：



其中，**感知层**主要是自企业内不同的业务系统和外部信息源获取数据；**存储层**采用同一的格式进行存储；基于这些数据集，可以在**加工层**开发增值应用，比如：实现可视化的信息集成表示、提供 RSS 消息发布源、支持企业内部的信息搜索、支持企业数据挖掘等等。在本扩展需求中，我们主要集中于可视化信息聚合方面。

## 内部结构

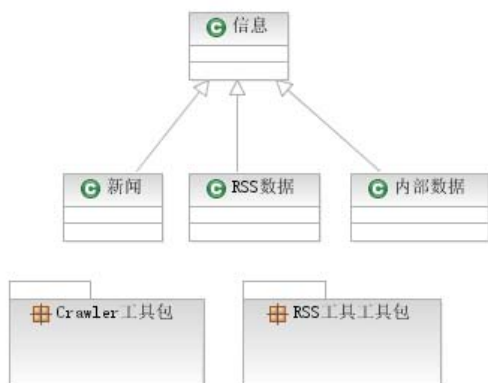
信息聚合部分的内部结构，我们以一个组件图的形式给出：



本图中包含的组件如下：

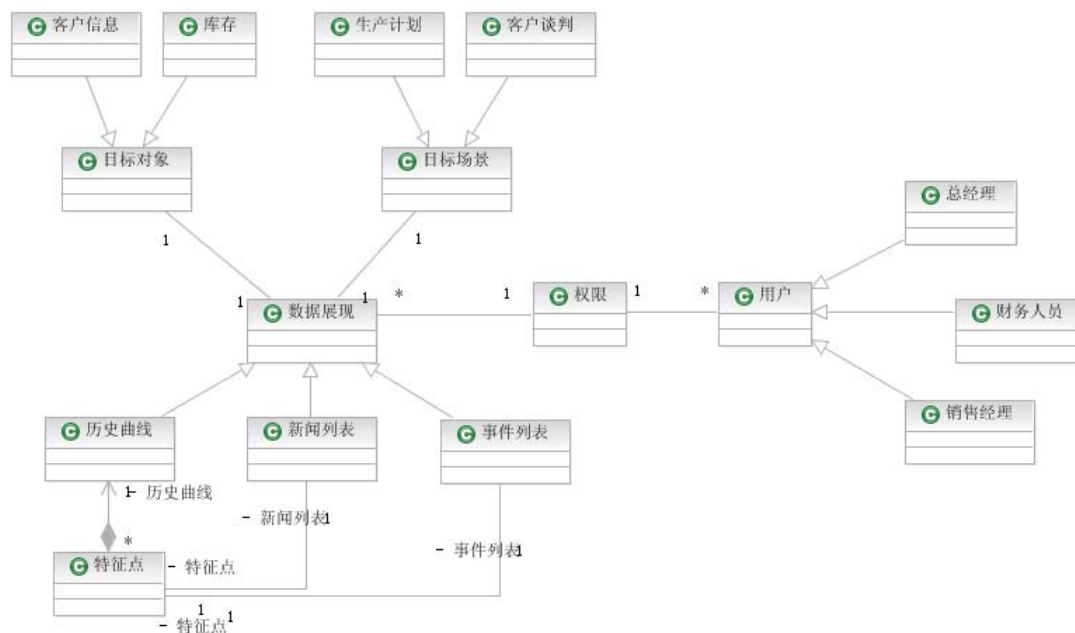
### ■ 信息采集组件

提供多种数据采集方式，包括搜索引擎、RSS、信息加工、信息存储、策略定制、信息展现、信息检索几个内部组件。由于上述组件中大部分我们都可以使用已有的工具支持实现，因此本节着重分析数据展现组件和信息采集组件的内部结构。



### ■ 数据展现组件

数据展现组件的功能是提供用户访问信息聚合组件的图形界面。我们以类图形式展现该组件的内部结构。



该图中，有以下的类需要加以叙述：

**数据展现类** 提供给用户的视觉接口，它有三种具体的实现形式，即历史曲线、新闻列表、事件列表。在历史曲线中，存在一个特征点，该点表示了曲线中值得关注的时间点，它很可能隐含了一个重大事件或消息，而我们的信息聚合组件则可以根据这个特征点关联到相关的新闻和事件记录上。

## ■ 数据加工组件



数据加工组件包括数据清理、切词、文本聚类三个工具包。而这三个工具包我们可以从已有开源组件和现有工具支持中得到，在此不详述其功能与设计。

## 2.4.3 组件规约

### 信息采集组件

#### ■ 规则

每天自动从 internet 获取客户、产品相关的新闻、信息  
由系统定制获取 internet 数据的站点、关键词

#### ■ 服务

定制获取的新闻信息、时间

#### ■ 属性

时间            Date  
关键词         String

#### ■ 变化点

[可配置] 摄取 internet 信息的时间间隔

[可配置] 摄取 internet 信息的内容

### 聚合信息展现组件

#### ■ 服务

查询最新曲线图

查询曲线图上关键点对应的新闻信息

查询曲线图上关键点对应的事件信息

#### ■ 属性

请求类别（产品、客户等） String

起始时间、终止时间         Date

### 信息加工组件

#### ■ 服务

信息加工

#### ■ 变化点

[可配置] 信息存储格式

[可配置] 摄取 internet 信息的内容

## 2.4.4 接口定义

#### ■ 查询最新曲线图

入口参数：查询对象(客户，产品等)类型，查询对象代码，起始时间，终止时间

返回值：    曲线数据点数据列表

异常：        入口参数不合法，图形不存在

#### ■ 查询曲线图上关键点对应的新闻信息

入口参数：员工ID号，查询对象(客户，产品销售，产品库存等)类型，查询对象代码，时间点

返回值：    新闻列表

异常：

#### ■ 查询曲线图上关键点对应的事件信息

入口参数：员工 ID 号，查询对象(客户，产品销售，产品库存等)类型，查询对象代码，时间点

返回值：    事件列表

异常

#### 2.4.5 非功能属性描述

- 一次获取最新数据及数据处理、存储必须在 3 小时之内完成
- 数据展现要求能在 1 秒钟之内绘制成图片

#### 2.4.6 实现方式

- 数据整合与存储由万合开发
- 信息采集
  - 互联网数据使用 Google 或 Yahoo 提供的服务
  - RSS 数据采集使用开源工具
- 数据加工
  - 数据加工内部使用的各组件均使用免费工具
  - 由万合整合这些免费工具

#### 2.4.7 实现机制分析

从实现方式一节我们可以看到，大多数的组件都使用的是开源的工具（符合尽量减少开支的要求），因此本节仅仅分析信息存储组件与信息展现的实现机制。

#### 信息存储组件

信息存储组件的关键在于存储格式与索引机制。

##### 存储格式

- 曲线图预计存储为一个图形数据列表
  - 新闻、事件存储为文本
- 索引机制
  - 采用倒排表索引文件

#### 信息展现组件

信息展现组件实现的难点主要在于高效、平滑的图形展现，对于这一点，我们预计采用 Ajax 技术，提高图形转换的效率。

另一个难点在于特征点的识别。特征点的识别基于两个重要条件，第一个条件是特征点所在的时间区域的新闻数量较多——这说明发生了较为令人关注的事件；另一个条件是特征点的数值的变化速率较大。

### 2.5 库存管理组件

#### 2.5.1 功能描述

记录产品分类信息、产品信息、产品价格、产品库存信息。除了对外提供访问这些信息的接口之外，该组件还提供检查产品库存是否满足需求的接口。

## 2.5.2 组件内部结构

ERP,CRM 提供的组件内部结构见用友 NC 系统白皮书和 Turbo CRM 系统白皮书，万合包装过后的组件结构见数据更新部分的组件结构图。

## 2.5.3 组件规约

### 服务

- 检查库存是否满足需求
- 冻结订单
- 对产品分类/产品信息/产品价格的添加、删除、查询

### 属性

- 产品 ID Integer
- 产品分类 ID Integer
- 数量 Integer

### 变化点

## 2.5.4 接口定义

### 冻结订单

输入：订单代号

输出：成功/失败信息

### Turbo CRM 提供的接口

| 产品分类信息相关调用 |             |              |                 |
|------------|-------------|--------------|-----------------|
| 获取产品分类信息   | 获取符合条件的产品分类 | GetProdClass |                 |
| 新增产品分类信息   | 新增一条/多条产品分类 |              | NewProdClass    |
| 更改产品分类信息   | 修改一条/多条产品分类 |              | UpdateProdClass |
| 删除产品分类信息   | 仅限于删除未使用的信息 |              | DelProdClass    |
|            |             |              |                 |
| 获取产品信息     | 获取符合条件的产品   | GetProd      |                 |
|            |             |              |                 |
| 新增产品信息     | 新增一条/多条产品   |              | NewProd         |
| 更改产品信息     | 修改一条/多条产品   |              | UpdateProd      |
| 删除产品信息     | 仅限于删除未使用的信息 |              | DelProd         |
|            |             |              |                 |
| 盘点单信息相关调用  |             |              |                 |
| 获取盘点单信息    | 获取符合条件的盘点单  | GetCheckBill |                 |
| 新增盘点单信息    | 新增一条/多条盘点单。 |              | NewCheckBill    |
| 更改盘点单信息    | 修改一条/多条盘点单， |              | UpdateCheckBill |
| 删除盘点单信息    | 仅限于删除未使用的信息 |              | DelCheckBill    |

### NC ERP 系统提供的接口

- 查询全部产品分类/产品/产品价格信息

输入：



输出：全部产品分类/产品/产品价格信息

#### ■ 查询产品库存信息

输入：包含产品属性信息的查询条件

输出：输出所有满足条件的全部库存信息

#### ■ 查询某项产品是否缺货

输入：产品 ID 号，需求数量

输出：当前库存是否能满足需求数量

### 2.5.5 实现机制分析

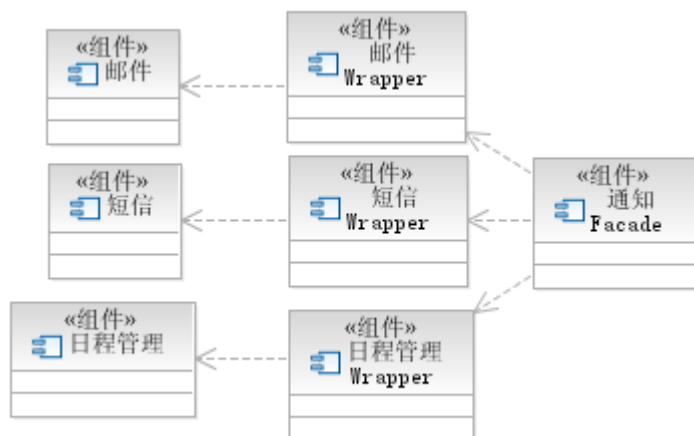
参考（ERP/CRM）数据源组件的结构图。我们使用 Wrapper（Adaptor）模式包装该 ERP,CRM 系统提供的接口，然后将该接口包装为 Web Services，发布到 ESB。

## 2.6 通知设施组件

### 2.6.1 功能描述

提供通知相关的服务，包括短信服务、Email 服务、日程管理等等。日程管理相关内容我们已经在日程管理组件中予以描述。而短信服务组件、Email 服务组件则是在已有系统中已有的组件。

### 2.6.2 组件内部结构



### 2.6.3 组件规约

#### 规则

外部构建调用此接口时，必须指明通知的类型是短信、邮件还是日程安排。

#### 服务

发送一条通知。

查询员工通知信息。

#### 属性

---

|       |              |
|-------|--------------|
| 日期    | Date         |
| 通知类型  | 枚举（短信/邮件/日程） |
| 人员 ID | Integer      |
| 通知内容  | String       |

### 变化点

[可配置] 消息模版机制

## 2.6.4 接口定义

### 发送通知

■ 入口参数：通知类型，发送对象 ID，发送内容

返回值： 发送结果代码

异常： 发送对象（人员）ID 不存在

### 查询个人通知

■ 入口参数：查询对象（短信/邮件/日程），起始时间，终止时间

返回值： 发送结果代码

异常：

## 2.6.5 非功能属性描述

要求通知设施组件能够在 1 秒之内响应请求，如果发送失败（如短消息发送超时）则应在 5 秒之内给出提示。

## 2.6.6 实现机制分析

由于短信服务、Email 服务、日程安排组件则是在已有系统中已有的组件，万合使用 Adaptor（Wrapper）模式包装这些接口，然后使用 Façade 模式提供统一的接口。此接口最后包装成 Web 服务，并发布到 ESB。

## 2.7 日程管理组件

在业务分析过程中，日程管理组件是作为订单管理的一部分存在的。由于在组件规约的分析过程中，我们发现日程管理实际是 NC ERP 系统中已经存在，因此我们将日程管理组件单独拿出来作为一个组件，并给出其设计。

### 2.7.1 功能描述

该组件提供对员工日程任务安排的管理，为员工维护一个日程—任务对应表，并列表当前需要完成的任务。

### 2.7.2 组件内部结构

由于目前 Turbo CRM 和 NC ERP 系统均提供了日程表功能，我们可以集成它们当前提供的功能。但是 Turbo CRM 和 NC ERP 系统目前并没有提供详细的文档说明。我们只能在它们提供了详细说明后才能确定具体的内部结构。

### 2.7.3 组件规约

#### 规则

- 不可写入重复的日程安排
- 人员可要求对日程安排重新分配
- 人员可要求取消日程安排
- 对于已经超负荷的人员可要求重新分配日程安排

#### 服务

- 增加、修改、删除一条日程安排
- 查看日程安排

#### 属性

- 日期     Date
- 人员 ID   Integer
- 日程安排(复合属性)
  - ◆ 日期     Date
  - ◆ 人员 ID   Integer
  - ◆ 事件 ID   String

#### 变化点

[可配置] 日程写入规则

### 2.7.4 接口定义

向员工工作日历中增加、修改一条待审批订单请求

输入：员工代码，订单请求代码

输出：成功或错误代码

从员工工作日历中删除一条订单请求

输入：员工代码，订单请求代码

输出：成功或错误代码

从员工工作日历中查看待审批订单请求列表

输入：员工代码，订单请求代码

输出：成功或错误代码

### 2.7.5 实现机制分析

由万合包装 NC 和 Turbo CRM 系统提供的接口为 Web Services，发布到 ESB。

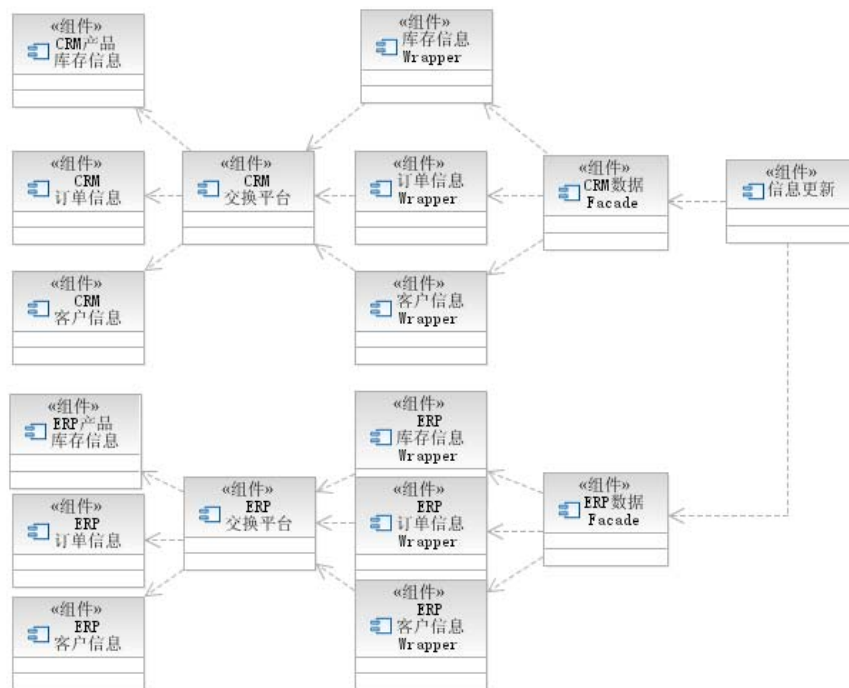
## 2.8 (CRM/ERP)数据源组件

### 2.8.1 功能描述

本部分为数据更新组件的基础设施，提供全部访问 CRM/ERP 数据库中需要同步更新的数据的接口。

## 2.8.2 组件内部结构

下图为包装之后的组件接口及其与数据更新组件的关系。



## 2.8.3 接口定义

库存信息、订单信息、客户信息以及外部交换平台的接口全部由 NC、Turbo CRM 提供。下面是 CRM 数据 Façade 组件的接口定义（ERP 数据 Façade 的接口与此类似）。

### 读取信息

- 入口参数：读取信息类型（产品/产品分类/产品价格/产品库存/订单/客户信息），起始时间，终止时间
- 返回值： 读取的记录条数
- 异常： 读取信息类型不存在，起始/终止时间不合法，数据库读写错误

### 写入信息

- 入口参数：导入信息类型（产品/产品分类/产品价格/产品库存/订单/客户信息）
- 返回值： 写入信息的条数
- 异常： 写入信息类型不存在，起始/终止时间不合法，数据库读写错误

## 2.8.4 非功能属性描述

要求能提供事务性支持，在数据读出、写入过程中的失败时支持回滚。

## 2.8.5 实现机制分析

我们使用 Wrapper（Adaptor）模式包装该 ERP,CRM 系统提供的接口，然后将该接口包装为 Web Services，发布到 ESB。