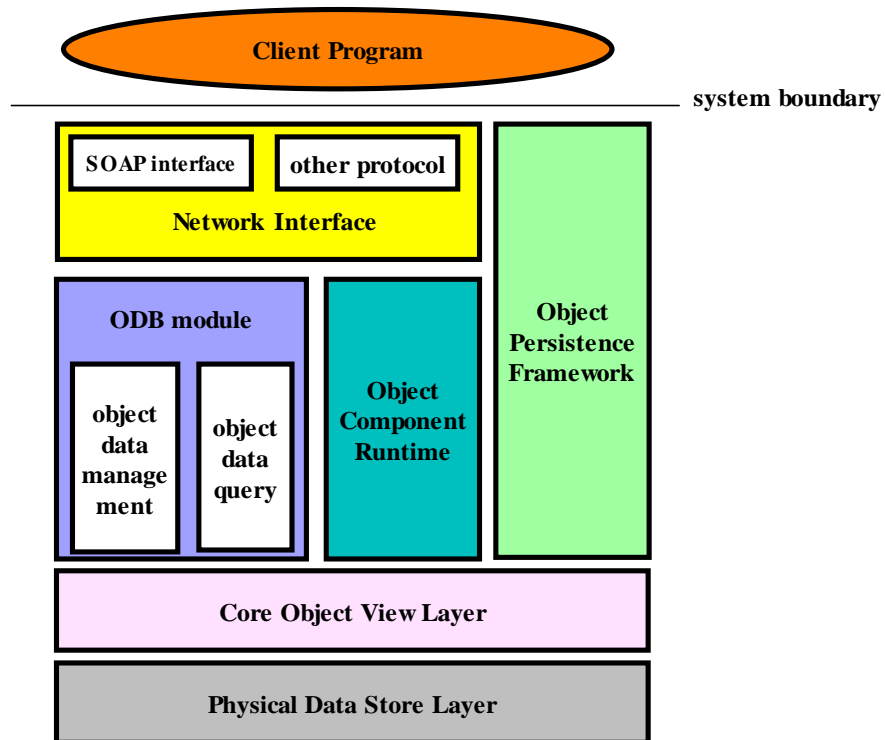


# 设计架构及技术报告

## (Detailed Design Specification)

### 系统设计架构概述

系统结构图如下：



系统可以划分为 3 个子系统，涵盖的模块和功能如下

#### I. 存储子系统

涵盖的模块：

Physical Data Store Layer module

Core Object View Layer module

## Object Persistence Framework)

### 功能:

负责将对象在持久存储介质的存储

## II. 对象数据库子

### 涵盖的模块:

#### ODB module

### 功能:

系统提供包括对象数据的存储管理、对象查询

## III. 组件运行子系统

### 涵盖的模块:

#### Object Component Runtime module

#### Network Interface module

### 功能:

对象组件运行环境, 包括识别组件、管理组件状态、执行组件计算, 提供远程访问借口

## 各程序模块功能描述如下:

### Physical Data Store Layer

负责数据在持久存储介质的存储, 提供低层的数据存贮

### Core Object View Layer

核心的对象视图, 将物理层的数据转换为对应的对象

### Object Persistence Framework

嵌入式的对象持久化框架

### Object Component Runtime

对象组件运行环境, 包括识别组件、管理组件状态、执行组件计算

### ODB module

对象数据库模块, 提供包括对象数据的存储管理、对象查询服务。

## Network Interface

通过网络访问系统的接口，他包括访问对象数据库的连接协议，和分布式计算的远程调用协议。

## Client Program

在此系统上编写的应用程序

## 主要功能模块描述及针对每个主要模块的结构图

### Physical Data Store Layer

#### 存储结构:

该模块是一个 Key-Value 模式的数据存储管理模块。将数据存贮定义以 Record ,Cluster, Type Set ,User Bank , Storage.

Record : 纪录, 代表一个数据流的物理存贮。

Cluster : 簇. 由多个记录组成。

Type Set: 类型数据集。由多个 Cluster 组成, 标示同一个类型的数据集合。

User Bank: 用户数据集合。由多个 Type Set 组成, 表示一个用户存贮的数据的集合。

Storage : 所有存储数据的集合。由多个 User Bank 组成。

在逻辑上, 上层管理下层。

还包括一个垃圾收集子模块。

#### 性能:

TBD

#### 输入项目:

二进制数据流.

#### 输出项目:

存储数据的记录集合

#### 程序逻辑:

如图一

#### 算法:

Garbage Collector Algorithm

### Object View Layer

#### 功能

将程序域中的对象与可存储的数据流相互转换。

#### 对象的状态和事务(Transaction)

所有需要存贮的对象都由此模块缓存并管理其状态。

对于托管的对象定义了以下几种状态:

NewClean : 新创建的干净数据

OldClean : 从存储中取出或经过修改后的干净数据

NewDirty : 新创建的、被修改过的脏数据

OldDirty : 从存储中取出或经过修改后的脏数据

NewDeleted : 新创建的、被删除的数据

OldDeleted : 从存储中取或经过修改后的被删除的数据

NotInit : 不可识别的]状态

只有干净的数据才能被写入存储中, 从存储中读取的数据的状态均为 OldClean 状态。

通过对对象的状态标示与识别, 实现事务。

当事物提交时, 将所有干净数据和脏数据写入存储中, 删除的数据也写入写入存储; 当



事务会滚时，除 OldClean 数据外，清空其他状态数据的缓存。

性能

TBD

输入项目

程序域中的对象

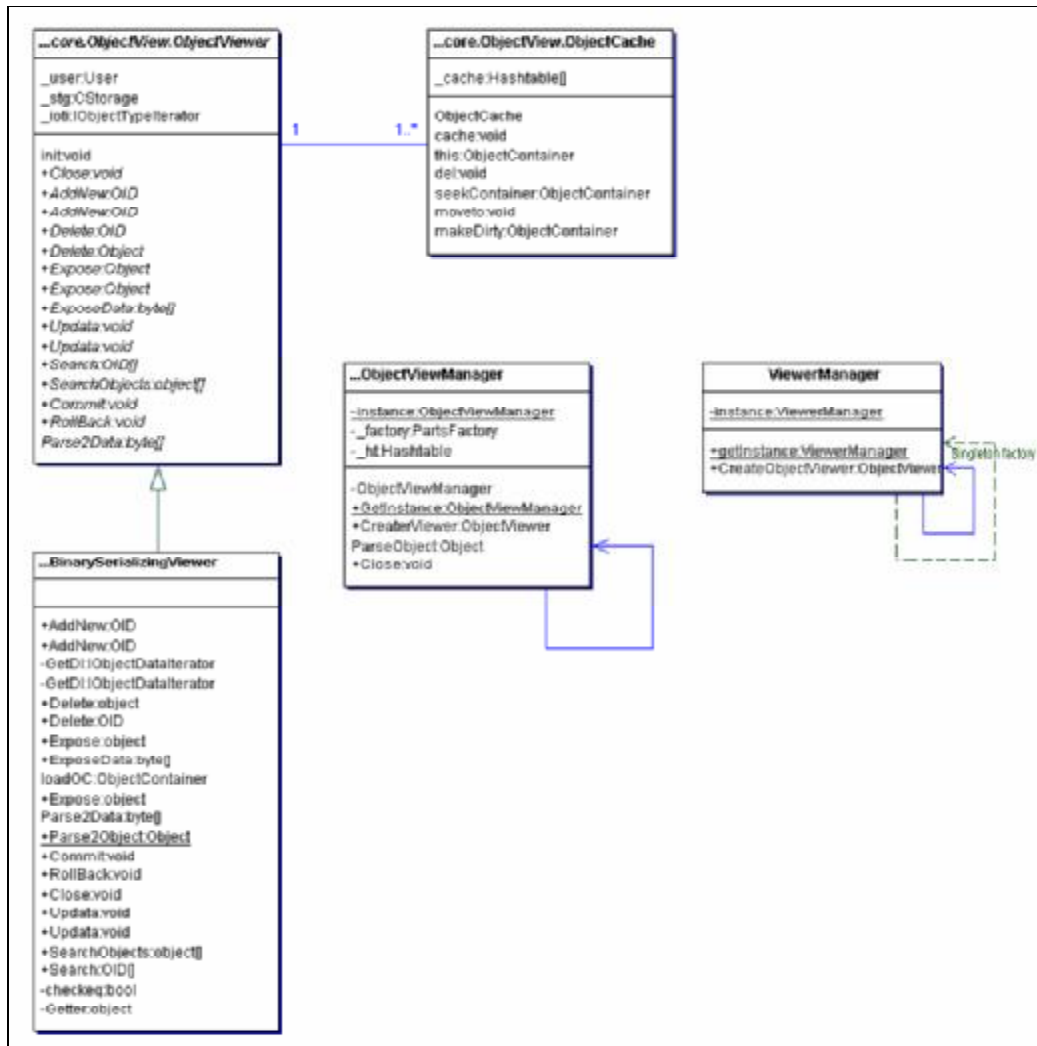
程序逻辑

如图二

算法

对象搜索算法 BinarySerializingViewer.SerachObject()

对给定的对象标示符,从缓存和存储中搜索该标示对应的对象.(流程图如下所示)



(图二)

**Object Component Runtime**

功能

对象组件运行环境，包括识别组件、管理组件状态、执行组件计算。

性能

TBD

输入项目

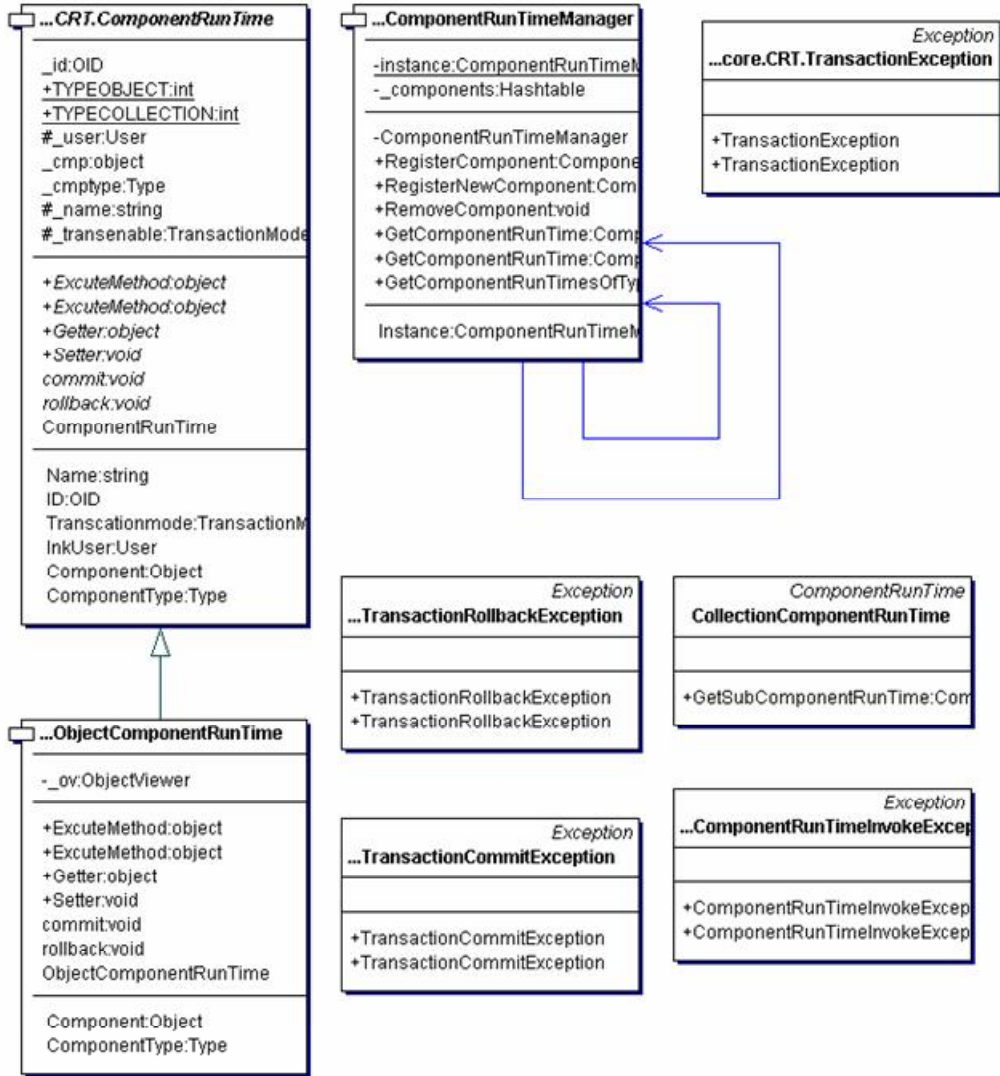
程序域中的对象和请求消息。

输出项目

组建方法用的计算结果。

程序逻辑

如图三



(图三)

**Network Interface**

功能

提供一个远程接口调用系统功能，实现 web service 调用接口。

设计

该模块包括两个部分。

Web Service Engine.: 解析识别 web service 请求，并将其转化为对组件方法的调用。

Web Service Proxy : 系统功能的 web service 代理。

性能

TBD

输入项目

Web Service 请求消息。







撰写人: tim

e\_mail: [fe\\_tim@hotmail.com](mailto:fe_tim@hotmail.com)

2003.6.15