**软件架构设计文档**

团队名称：141250117\_cseiii\_AnyQuant

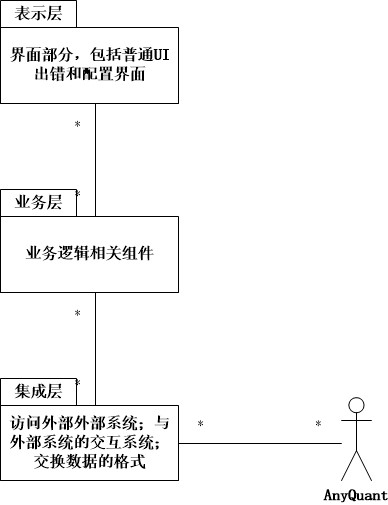
文档更新记录：

|  |  |  |  |
| --- | --- | --- | --- |
| **版本** | **作者** | **版本描述** | **日期** |
| V0.1 | 宋子微 | 草稿 | 2016/3/3 |
|  |  |  |  |

引言：

|  |  |
| --- | --- |
| 内容和说明 |  |
| 1 编写说明 | 本文档提供了量化交易系统软件架构概览，采用若干架构视图描述系统的不同方面，一遍表示构造系统所以需要的重要架构决策 |
| 2 对象与范围 | 本文档的读者是本小组内部的开发和管理人员，参考了RUP的《软件架构文档模板》，用于指导下一循环的代码开发和测试工作 |
| 3 参考文献 | 《软件架构文档模板》，Rational Software Corporation,2002 |
| 4 名词与术语 | 无 |

系统的分层架构：

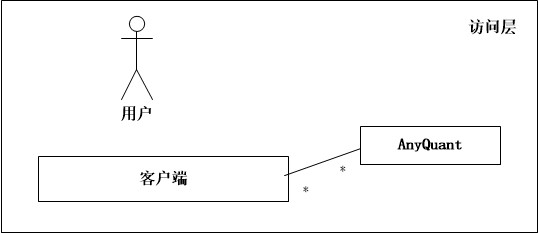


系统划分为以下3个逻辑层次。

1）表示层：用于前台界面展示和配置的层次。

2）业务层：包含业务控制和逻辑的层次。

3）集成层：定义和集成与外部系统交互借口的层次。



系统可以部署在以下1个物理层次.

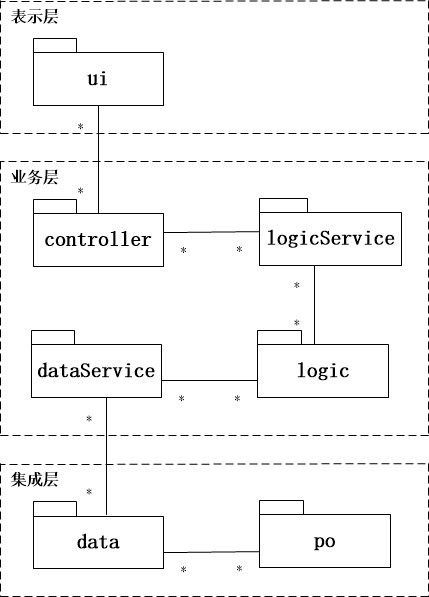
1. 访问层：用于用户访问系统的层次。

系统的架构设计如下。

系统架构中的对象分为7类：

1. ui对象，负责处理系统数据的展现和用户的交互。
2. contorller对象，控制器负责获取用户输入，并调用logicService模块的服务。
3. logicService对象，负责提供服务的抽象借口，获取从数据段组装好的数据。
4. logic对象，负责对于抽象借口的实现模块。
5. dataService对象，负责封装从IDAO获取的批量数据的借口。
6. data对象，负责与API交互，获取数据。
7. po对象，该模块用来将从API中获取的数据封装成数据实体

系统中的组件和组件接口：



|  |  |  |  |
| --- | --- | --- | --- |
| **接口ID** | **连接组件** | **借口信息** | |
| I1 | 连接UI和IController | 语法 | Return(Response)  Interface(Request) |
| 前置条件 | 用户的输入正确 |
| 后置条件 | 处理控制组件处理请求并相应 |
| 不变量 | 用户请求信息 |
| I2 | 连接Controller与logicService | 语法 | Return(result)  Interface() |
| 前置条件 | 无 |
| 后置条件 | 对应的logicService执行对应的业务逻辑 |
| 不变量 | 无 |
| I3 | 连接logicService与dataService | 语法 | Return(dataset)  Interface(command) |
| 前置条件 | 无 |
| 后置条件 | 对应的dataService组件调用特定data类获取API数据，并返回数据集 |
| 不变量 | 无 |
| I4 | 连接data与po | 语法 | Return(data)  Interface(criteria) |
| 前置条件 | API连接正常 |
| 后置条件 | data中的类从API中返回po对象 |
| 不变量 | 无 |