

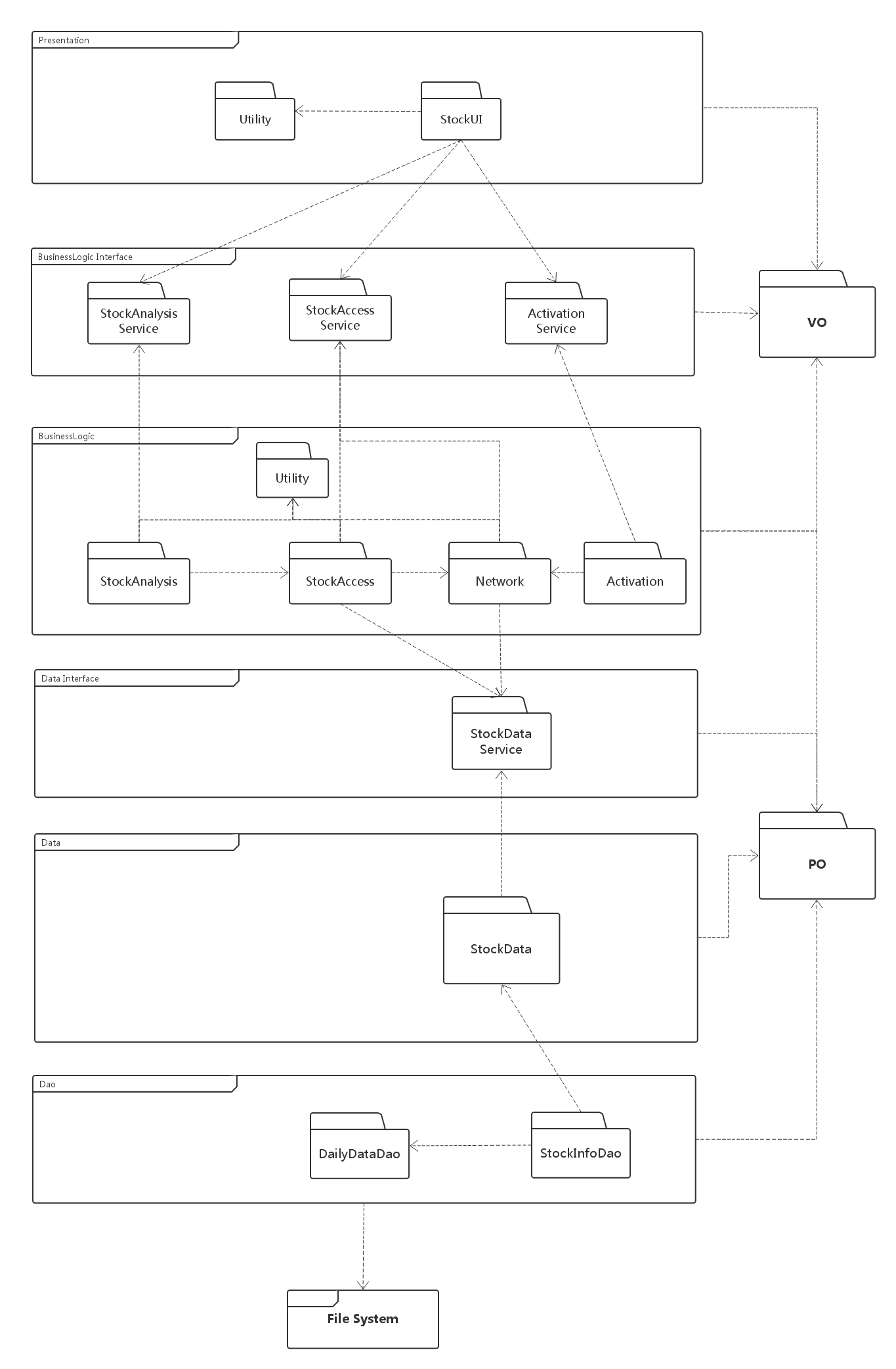
**成员** 倪安松 徐朱峰 余旻晨 吴嘉荣

XEON

项目启动文档

|  |  |  |
| --- | --- | --- |
| 修改人 | 修改时间 | 修改内容 |
| 徐朱峰 | 2016-3-30 | 体系结构文档初版 |
| 徐朱峰 | 2016-3-6 | 添加逻辑层模块接口规范 |
| 倪安松 | 2016-3-20 | 根据迭代二要求适当修改 |
| 倪安松 | 2016-3-21 | 增加数据层接口规范 |
| 倪安松 | 2016-3-30 | 修改体系结构包图 |

1. **组合视角**
2. **开发包图**



1. **接口视角**
2. **模块职责**

|  |  |
| --- | --- |
| 模块名称 | 模块职责 |
| Ui.stockui | 展示主要界面，图表和数据 |
| Ui.utility | 辅助界面显示设置、参数计算的工具模块 |
| Bussinesslogic.activation | 启动模块 |
| Businesslogic.network | 网络模块 |
| Businesslogic.stockaccess | 股票数据获取模块 |
| Businesslogic.stockanalysis | 股票数据的计算与处理模块 |
| Data.impl | 数据实现，负责组合数据传输，数据库连接 |
| Data.Utility | 数据工具，包括数据从网络获取，数据格式转换等 |
| Data.dao | 负责数据的读写 |

1. **业务逻辑层的接口规范**

见代码中规定，下面给出部分复杂接口功能解释和用法

|  |  |
| --- | --- |
| 名称 | 接口功能描述 |
| StockSourceService | 用来提供实际的StockInfo和DailyData的查询操作，数据过滤的实现在这里 |
| StockInfoFilterService | 相当于一个查询的工具，有按照股票代码、年份来过滤的方法，每调一次filterxxx，对象就自行增加一个过滤条件  在调用getResult时，再从StockSourceService那里实际调用数据过滤的实现。  并且支持过滤条件集合的并、交、差操作 |
| StockDailyDataFilterService | 也是一个查询的工具，和StockInfoFilterService用法相似 |

StockInfoFilterSerivce用法：

StockInfoFilterService foo = new ...；

StockInfoFilterService newFoo =

foo.filterYear(2015).filterExchange(ExchangeEnum.SHANGHAI);

Collection<StockInfoPO> stocks = newFoo.getResult();

StockInfoFilterService bar = new ...；

Collection<StockInfoPO> newStocks =

bar.filterYear(2014).union(newFoo).getResult();

1. **数据层的接口规范**

数据层的接口规范描述如下：

|  |  |
| --- | --- |
| 名称 | 接口功能描述 |
| Iterator<DailyDataPO> getStockDaily(String symbol, Calendar startDate, Calendar endDate) throws NetworkConnectionException | symbol 股票代码; param startDate 开始日期 （若为null则默认为最近一个月）; endDate 结束日期（若为null则默认为最近一个月）; return 开始日期到结束日期内的该股票的所有DailyData |
| Iterator<StockInfoPO> getBenchmark() throws NetworkConnectionException | 获取大盘数据，返回所有大盘数据的迭代器 |
| Iterator<StockInfoPO> getAllStock() | 获取所有股票PO，返回所有StockInfoPO的迭代器 |

1. **信息视角**
2. **数据持久化对象**

|  |  |
| --- | --- |
| 名称 | 属性描述 |
| UserPO | 用户类，包含用户的信息 |
| StockInfoPO | 股票信息类，包含股票自身包含的信息 |
| DailyDataPO | 股票日数据类，包含股票当天的实时数据信息 |

1. **持久化格式**

序列化.ser格式存储，但是大部分数据直接从网上API获取