

设计文档

NBAlabala数据分析系统



2015-3-12

NBAlabala团队

更新历史

|  |  |  |  |
| --- | --- | --- | --- |
| 修改人员 | 日期 | 变更原因 | 版本号 |
| 丁霄汉 | 2015-3-12 | 最初草稿 | V1.0 |
| 丁霄汉 | 2015-3-14 | 完成除了接口定义之外的部分 | V1.1 |
| 梁思宇 | 2015-3-16 | 完成接口定义 | V1.2 |
| 曾婧 | 2015-3-22 | 更新文档，整理和排版 | V1.3 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

目录

1. 引言 1

1.1. 编制目的 1

2. 产品描述 1

3. 逻辑视角 1

4. 组合视角 2

4.1. 开发包图 2

5. 接口视角 4

5.1. 业务逻辑层的分解 4

5.1.1. 业务逻辑层模块的职责 4

5.1.2. 业务逻辑模块的接口模范 4

5.4. 数据层的分解 7

5.4.1. 数据层模块的职责 7

5.4.2. 数据层模块的接口规范 8

# 引言

## 编制目的

本文档详细完成对NBAlabala数据分析系统的概要设计，达到指导开发的目的，同时实现测试人员及用户的沟通。

本文档面向开发人员、测试人员及最终用户编写，是了解系统的导航。

# 产品描述

参考NBAlabala数据分析系统软件规格说明中对产品的概括描述。

# 逻辑视角

NBAlabala数据分析系统中，选择了分层体系结构的风格，将系统分为3层（展示层、业务逻辑层、数据层）能够很好的示意整个高层抽象。展示层包括GUI页面的实现，业务逻辑层包含业务逻辑处理的实现，数据层负责数据的读取。分层体系结构的逻辑视角和逻辑设计方案如图1和图2所示。

图 1逻辑视角

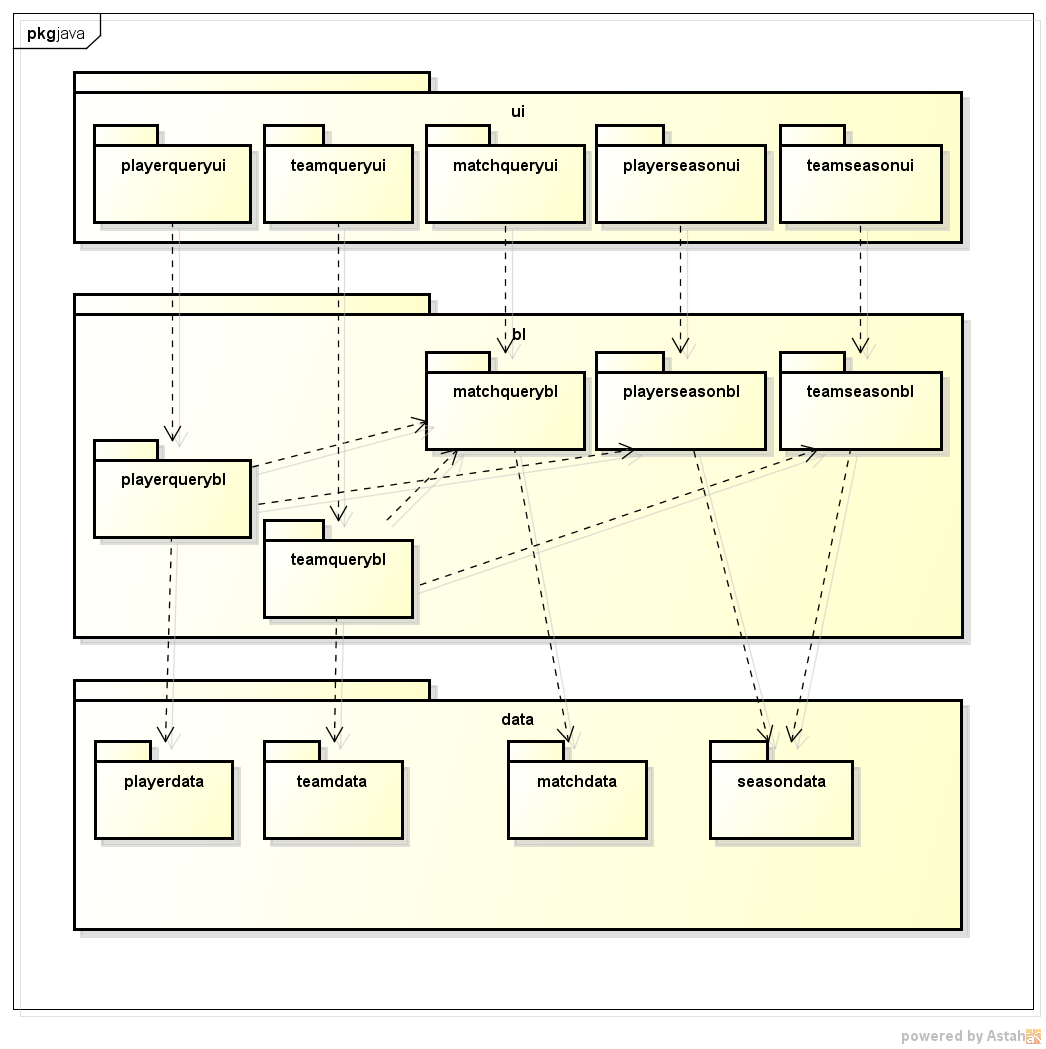


图 2 软件体系结构逻辑设计方案

# 组合视角

## 开发包图

灯具照明行业进销存系统的最终开发包设计如图3和表1所示

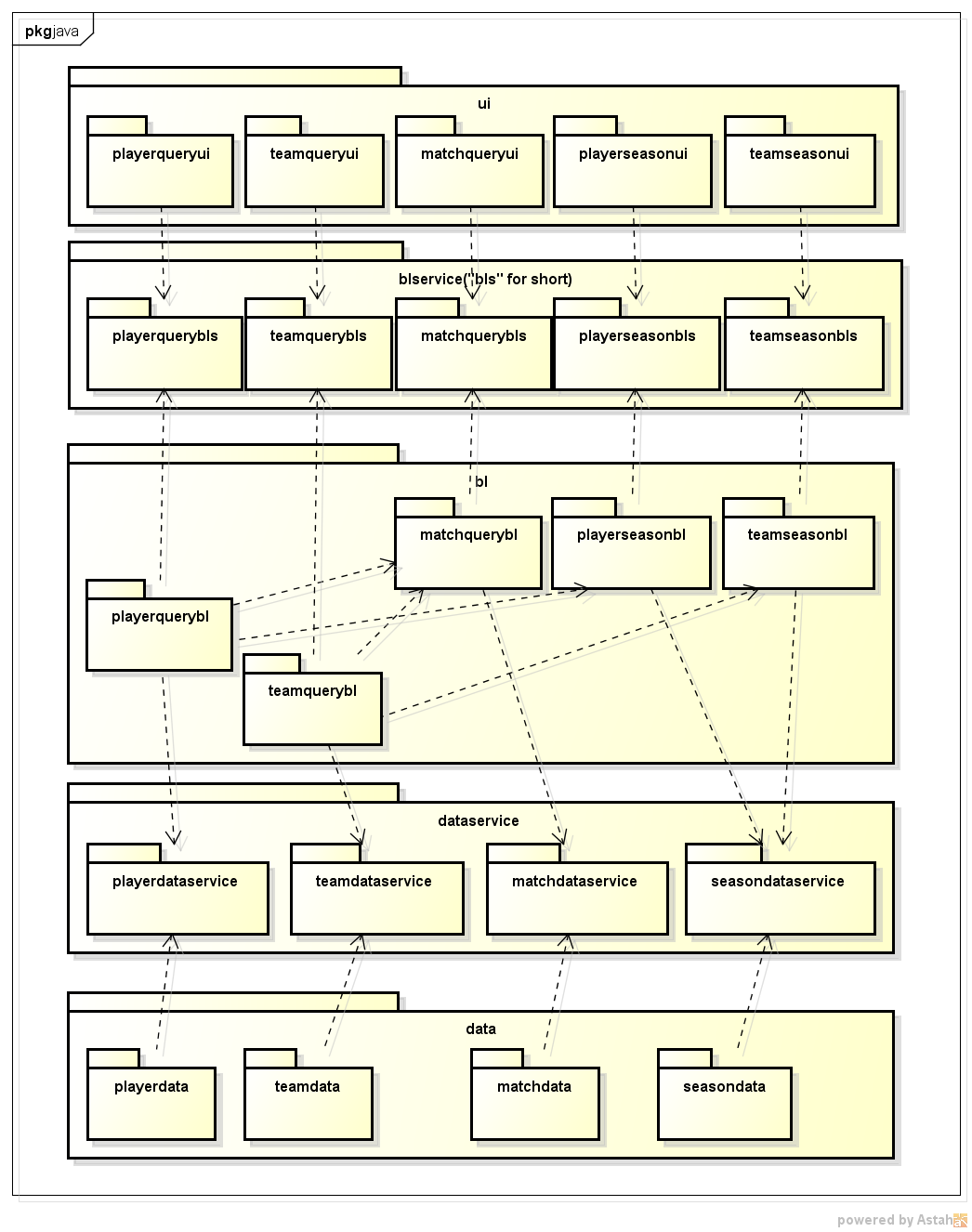


图 3开发包图

**表1 层之间调用的接口**

|  |  |  |
| --- | --- | --- |
| 接口 | 服务调用方 | 服务提供方 |
| playerqueryblservice  teamqueryblservice  matchqueryblservice  playerseasonblservice  teamseasonblservice | 展示层 | 业务逻辑层 |
| playerdataservice  teamdataservice  matchdataservice  seasondataservice | 业务逻辑层 | 数据层 |

# 接口视角

## 业务逻辑层的分解

### 业务逻辑层模块的职责

|  |  |
| --- | --- |
| 模块 | 职责 |
| playerquerybl | 负责实现查询球员信息界面所需要的服务 |
| teamquerybl | 负责实现查询球队信息界面所需要的服务 |
| matchquerybl | 负责实现查询比赛信息界面所需要的服务 |
| playerseasonbl | 负责实现球员赛季数据分析界面所需要的服务 |
| teamseasonbl | 负责实现球队赛季数据分析界面所需要的服务 |

### 业务逻辑模块的接口模范

**playerquerybl接口规范**

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| playerquerybl.getPlayerProfileByInitial | 语法 | public ArrayList<PlayerProfileVO> getPlayerProfileByInitial(char initial) |
| 前置条件 | initial是从A-Z的26个大写英文字母之一 |
| 后置条件 | 根据名字首字母查找球员简况列表，按字典顺序排序并返回 |
| playerquerybl. getPlayerDetailByName | 语法 | public PlayerDetailVO getPlayerDetailByName(String name) |
| 前置条件 | 球员名name确实存在 |
| 后置条件 | 返回该球员的详细信息，包括球员简况、比赛记录、赛季数据 |
| 需要的服务（需接口） | | |
| 服务名 | | 服务 |
| playerdata.getPlayerProfileByInitial(char initial) | | 根据首字母得到球员简况列表 |
| playerdata.getActionImageByName(String playerName) | | 根据球员名字得到其全身像 |
| playerseasonbl.getPlayerSeasonDataByName(String name) | | 根据球员名字得到其赛季数据 |
| matchquerybl.getMatchRecordByPlayerName | | 根据球员名字得到其比赛历程记录 |

**teamquerybl接口规范**

|  |  |  |  |
| --- | --- | --- | --- |
| 提供的服务（供接口） | | | |
| teamquerybl. getTeamDetailByAbbr | 语法 | | public TeamDetailVO getTeamDetailByAbbr(String abbr) |
| 前置条件 | | 球队缩写abbr确实存在 |
| 后置条件 | | 返回球队详情，包括阵容、球队简况、赛季数据 |
| 需要的服务（需接口） | | | |
| 服务名 | | 服务 | |
| teamdata.getTeamProfileByAbbr(abbr) | | 根据球队缩写得到球队详情 | |
| playerseasonbl.getPlayerNamesByTeamAbbr(abbr) | | 根据球队缩写得到属于这一球队的球员名单 | |
| playerquerybl.getPlayerProfilesByNames(Arraylist<String> playerNames) | | 根据球员姓名列表返回其简况列表 | |
| teamseasonbl.getTeamDataByAbbr(String abbr) | | 根据球队缩写得到其赛季数据 | |

**matchquerybl接口规范**

|  |  |  |  |
| --- | --- | --- | --- |
| 提供的服务（供接口） | | | |
| matchquerybl. screenMatchByDate | 语法 | public ArrayList<MatchProfileVO> screenMatchByDate(Date date) | |
| 前置条件 | Date是非null的java.util.Date对象 | |
| 后置条件 | 返回符合日期筛选条件的比赛简报 | |
| matchquerybl. screenMatchByTeam | 语法 | public ArrayList<MatchProfileVO> screenMatchByTeam(String abbr1, String abbr2) | |
| 前置条件 | abbr1和abbr2为两支球队缩写，或者为””表示不作限制 | |
| 后置条件 | 返回符合两队筛选条件的比赛简报 | |
| matchquerybl. getMatchDetail | 语法 | public MatchDetailVO getMatchDetail(String season, String date, String homeAbbr, String roadAbbr); | |
| 前置条件 | season形如13-14，date形如01-12，homeAbbr和roadAbbr是确实存在的两队缩写 | |
| 后置条件 | 返回当场比赛详情，包括比赛简报、两队logo、每个球员的比赛记录 | |
| 需要的服务（需接口） | | | |
| 服务名 | | | 服务 |
| matchdata.getMatchProfileBySeasonAndDate(String seasonString, String dateString) | | | 通过字符串格式的赛季和日期信息返回符合条件的比赛简报 |
| matchdata.getMatchProfileByTeam(String twoTeams) | | | 根据形如ABC-DEF这样的两队信息得到符合条件的比赛简况 |
| matchdata.getMatchDetailByFileName(String fileName) | | | 根据比赛信息文件名得到比赛详情 |

**playerseasonbl接口规范**

|  |  |  |  |
| --- | --- | --- | --- |
| 提供的服务（供接口） | | | |
| playerseasonquery.getAllPlayersSortedByName | 语法 | public ArrayList<PlayerSeasonRecord> getAllPlayersSortedByName() | |
| 前置条件 | 无 | |
| 后置条件 | 返回按名字字典顺序排序的球员赛季数据 | |
| playerseasonbl. getResortedPlayers | 语法 | public ArrayList<PlayerSeasonRecord> getResortedPlayers(PlayerSortBasis basis, SortOrder order) | |
| 前置条件 | 界面已经取得了按某种顺序排列的球员赛季数据 | |
| 后置条件 | 返回对当前表重新排序后的表 | |
| playerseasonbl. getScreenedPlayers | 语法 | public ArrayList<PlayerSeasonRecord> getScreenedPlayers(Position position, ScreenDivision division,ScreenBasis basis) | |
| 前置条件 | 界面已经取得了按某种顺序排列的球员赛季数据 | |
| 后置条件 | 根据球员位置、赛区、筛选依据筛选并返回前50个球员 | |
| 需要的服务（需接口） | | | |
| 服务名 | | | 服务 |
| seasondata.getAllPlayerSeasonData() | | | 得到全部球员赛季数据 |
| seasondata.getScreenedPlayerSeasonData(Position position, ScreenDvision division) | | | 根据球员位置和赛区得到符合条件的全部球员赛季数据 |

**teamseasonbl接口规范**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 提供的服务（供接口） | | | | |
| teamseasonbl.getTeamDataSortedByName() | 语法 | public ArrayList<TeamSeasonRecord> getTeamDataSortedByName() | | |
| 前置条件 | 无 | | |
| 后置条件 | 得到按字典顺序排序的所有球队赛季数据 | | |
| teamseasonbl. getScreenedTeamData | 语法 | public ArrayList<TeamSeasonRecord> getScreenedTeamData(ScreenDivision division) | | |
| 前置条件 | 无 | | |
| 后置条件 | 返回符合赛区条件的全部球队赛季数据 | | |
| teamseasonbl. getResortedTeamData | 语法 | public ArrayList<TeamSeasonRecord> getResortedTeamData(TeamSortBasis basis, SortOrder order) | | |
| 前置条件 | 界面已经得到了按某种顺序排列的球队赛季数据 | | |
| 后置条件 | 按排序依据和顺序重新排列并返回球队赛季数据 | | |
| 需要的服务（需接口） | | | | |
| 服务名 | 服务 | | | |
| seasondata.getScreenedTeamSeasonData(ScreenDivision division) | | | 得到某一赛区的全部球队赛季数据 |

## 数据层的分解

数据层主要给业务逻辑层提供数据访问服务，主要是读取txt格式持久化的赛季数据。

### 数据层模块的职责

数据层模块职责如下表所示

**数据层模块职责**

|  |  |
| --- | --- |
| 模块 | 职责 |
| playerdata | 负责读取球员信息，包括头像、全身像、身高等个人信息 |
| teamdata | 负责读取球队信息，包括logo、所在地等基本信息 |
| matchdata | 负责从比赛数据文件中检索所需的比赛信息 |
| seasondata | 负责遍历全部比赛数据生成赛季数据 |

### 数据层模块的接口规范

数据层模块的接口规范如下表

**数据层模块的接口规范**

playerdata**接口规范**

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| playerdata. getActionImageByName | 语法 | public Image getActionImageByName(String name) |
| 前置条件 | name不为null |
| 后置条件 | 根据球员姓名返回其全身像。如果没有找到该球员的全身像，返回一个缺省全身像表示之 |
| playerdata. getPlayerProfileByInitial | 语法 | public ArrayList<PlayerProfileVO> getPlayerProfileByInitial(  char initial) |
| 前置条件 | initial是A-Z的大写之母之一 |
| 后置条件 | 返回符合该首字母的球员的简况列表 |
| playerdata. getPlayerProfileByName | 语法 | public PlayerProfileVO getPlayerProfileByName(String name) |
| 前置条件 | 球员姓名name不为null |
| 后置条件 | 返回该球员简况。如果没有找到该球员的信息，返回一个缺省简况表示之 |
| playerdata. getPlayerProfileByNames | 语法 | public ArrayList<PlayerProfileVO> getPlayerProfileByNames(  ArrayList<String> names) |
| 前置条件 | names不为null |
| 后置条件 | 根据球员姓名列表返回简况列表 |

**teamdata接口规范**

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| teamdata. getTeamProfileByAbbr | 语法 | public TeamProfileVO getTeamProfileByAbbr(String abbr) |
| 前置条件 | 球队缩写abbr确实存在 |
| 后置条件 | 返回该球队信息 |
| teamdata. getAreaByAbbr | 语法 | public ScreenDivision getAreaByAbbr(String abbr) |
| 前置条件 | 球队缩写abbr确实存在 |
| 后置条件 | 返回该球队所属联盟（东/西） |
| teamdata. getDivisionByAbbr | 语法 | public ScreenDivision getDivisionByAbbr(String abbr) |
| 前置条件 | 球队缩写abbr确实存在 |
| 后置条件 | 返回该球队所属赛区（太平洋/西南……） |

**matchdata接口规范**

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| matchdata. getMatchProfileBySeasonAndDate | 语法 | public ArrayList<MatchProfileVO> getMatchProfileBySeasonAndDate(String season, String date) |
| 前置条件 | 赛季season形如13-14，date形如01-12 |
| 后置条件 | 返回符合赛季和日期条件的比赛简报 |
| matchdata. getMatchProfileByTeam | 语法 | public ArrayList<MatchProfileVO> getMatchProfileByTeam(String team) |
| 前置条件 | 球队缩写team确实存在 |
| 后置条件 | 查找返回有该球队的比赛简况 |
| matchdata. getMatchDetailByFileName | 语法 | public MatchDetailVO getMatchDetailByFileName(String fileName) |
| 前置条件 | fileName对应的比赛数据文件存在 |
| 后置条件 | 读取该文件并返回该场比赛详情 |
| matchdata. getMatchRecordByPlayerName | 语法 | public ArrayList<PlayerMatchPerformanceVO> getMatchRecordByPlayerName(  String playerName) |
| 前置条件 | 无 |
| 后置条件 | 返回有该球员参加的比赛记录，如果该球员一场比赛都没参加则返回一个空的表 |

**seasondata接口规范**

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| seasondata. getScreenedPlayerSeasonData | 语法 | public ArrayList<PlayerSeasonRecord> getScreenedPlayerSeasonData(Position position, ScreenDivision division) |
| 前置条件 | 无 |
| 后置条件 | 根据位置、赛区来返回符合条件的球员 |
| seasondata. getAllPlayerSeasonData | 语法 | public ArrayList<PlayerSeasonRecord> getAllPlayerSeasonData() |
| 前置条件 | 无 |
| 后置条件 | 返回全部球员赛季数据 |
| seasondata. getScreenedTeamSeasonData | 语法 | public ArrayList<TeamSeasonRecord> getScreenedTeamSeasonData(ScreenDivision division) |
| 前置条件 | 无 |
| 后置条件 | 根据赛区返回符合条件的记录 |
| seasondata. getTeamAbbrByPlayer | 语法 | public String getTeamAbbrByPlayer(String playerName) |
| 前置条件 | 无 |
| 后置条件 | 根据球员名字返回其最后一次比赛的球队。如果找不到该球员的比赛记录，返回“未知” |
| seasondata. getPlayerSeasonDataByName | 语法 | public PlayerSeasonRecord getPlayerSeasonDataByName(  String playerName) |
| 前置条件 | 无 |
| 后置条件 | 根据球员名字返回其赛季数据。如果没有该球员比赛记录，返回一缺省值。 |
| seasondata. getTeamDataByAbbr | 语法 | public TeamSeasonRecord getTeamDataByAbbr(String abbr) |
| 前置条件 | 球队缩写确实存在 |
| 后置条件 | 返回该球队的赛季数据 |
| seasondata. getPlayerNamesByTeamAbbr | 语法 | public ArrayList<String> getPlayerNamesByTeamAbbr(String abbr) |
| 前置条件 | 球队缩写存在 |
| 后置条件 | 返回该球队的所有球员名单 |