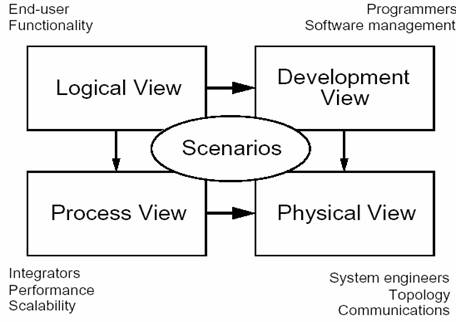
## 4.视图之间关系

### 4.1 视图之间关系说明

上述的5种视图, 每一个视图只关心系统的一个侧面, 5个视图结合在一起才能反映系统的软件体系结构的全部内容。各个视图之间的关系如下图所示, 其中Physical View就是物理视图(部署视图),



### 4.2 视图-视图关系

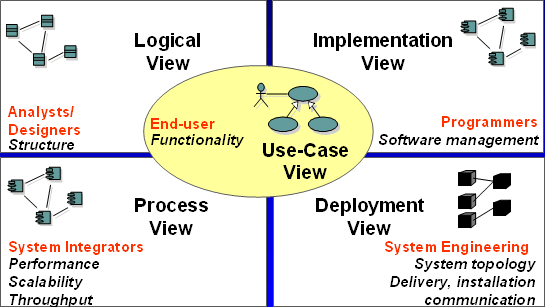
#### 用例视图和其他四个视图的关系

用例视图是五个视图的核心，是因为它确定了以下信息，而其它4个视图都是需要围绕着这些信息进行设计：

系统边界：有了边界，才能够确定系统的设计范围；同时，通过边界能够识别出系统需要与用户或其它系统进行交互；

系统用户：明确的用户定义是系统需求分析的先决条件；

功能和场景：通过识别出系统与用户或其它系统的交互，可以分析出系统需要提供哪些功能，以及这些功能存在哪些应用场景；



#### 逻辑视图与开发视图的关系

|  |  |
| --- | --- |
| *逻辑视图* | *开发视图* |
| BBS论坛系统 | B/S架构 |
| BBS论坛界面 | CSS |
| 用户界面交互 | JavaScript + Ajax + JSP |
| 管理员界面交互 |
| 用户数据 | MySQL表 |
| 帖子数据 |
| BBS系统服务 | SSH后端服务框架 |
| 管理员(用户管理, 帖子管理, 板块管理, 公告管理, 帮助管理) |
| 用户(浏览, 注册, 登录, 搜索, 发帖, 回帖, 个人中心) |

#### 逻辑视图与运行视图的关系

|  |  |
| --- | --- |
| *逻辑视图* | *运行视图* |
| 信息访问 | 使用Hibernate, 通过持久化数据对象，进行对象关系的映射，并以对象的角度来访问数据库 |
| 界面与交互 | 属于客户端技术, 不在讨论范围 |
| BBS系统服务 | 使用Struts,  1、HTIP请求。  2、HITP请求到达服务器，经过控制器，使用框架结构填充FormBean。  3、经过填充的数据请求通过框架中控制器的Action组件进行数据业务处理。  4、经过控制器处理过的业务数据调用struts的基础部件实现业务逻辑。  5、将业务逻辑经过转换形成JSP业务进行处理，井将处理结果返回 |
| 管理员(用户管理, 帖子管理, 板块管理, 公告管理, 帮助管理) |
| 用户(浏览, 注册, 登录, 搜索, 发帖, 回帖, 个人中心) |

#### 逻辑和开发视图与部署视图的关系

|  |  |
| --- | --- |
| *开发视图, 运行视图* | *部署视图* |
| MySQL, Hibernate | MySQL5.7 on Database Server |
| SSH服务框架, Struts和Spring | Apache Tomcat7.0 on Web Server |
| CSS | 客户端浏览器, 要求支持运行JavaScript脚本 |
| JS, AJAX, JSP |