服务器结构



# 值对象框架

在com.gudigital.core包里面包含了值对象的定义，用来表示一种通用的结构化数据。值对象可以用于：

1. 封装请求和响应的内容
2. 作为消息的载体
3. 作为模型数据的载体

值对象可以处理的18种字段类型（包括9种基础类型和对应的9种数组）。

值对象有一个type属性，用来标识其类型（比如请求类型、响应类型、消息类型等）。

# Bean框架

模仿Spring设计的应用框架，主要解决的问题有：

1. 从XML配置文件中获取Bean的参数
2. 统一执行Bean的初始化
3. 在Bean基础上构造服务对象

Bean与值对象一起构成了应用程序的框架

# Socket Server

使用Mina实现的Socket Server。可以将请求转交给Request Handler Factory处理，同时与Session Manager通信，管理用户的会话过程。

前期版本已经完成，但由于GUObject序列化方式的变化，需要重新调试。

# HTTP Server

使用Jetty实现的HTTP Server。可以将请求转交给Request Handler Factory处理，同时与Session Manager通信，管理用户的会话过程。

目前还基本没有做，要求是和Socket Server做到接口统一。

# Session Manager

管理已经登录的用户的会话，通过会话管理向用户发送消息，并且该模块还需要处理短线，重连等情况。

# Request Handler Factory

前端接入层将请求转发给Request Handler Factory处理。对于发送请求的会话来说，如果还没有登录，则调用ILoginHandler来处理。

# Model

模型，处理服务器的数据。

## Data Set

## Data Proxy

## Data Source

实现了基于DBCP的一个实现，基本够用了。

# Event Service

事件服务，提供事件的分发服务。目前实现了同步和异步两种消息分发机制。

# Schedule Service

定时任务服务，基本没有做。