概要设计说明书

目 录Table of Contents

[1 总体设计 3](#_Toc357336810)

[1.1 系统描述 3](#_Toc357336811)

[1.1.1 需求规格概述 3](#_Toc357336812)

[1.1.2 运行环境规定 3](#_Toc357336813)

[1.2 系统结构和处理流程 3](#_Toc357336814)

[1.2.1 系统结构 3](#_Toc357336815)

[1.2.2 基础层 4](#_Toc357336816)

[1.2.3 全局模块层 4](#_Toc357336817)

[1.2.4 应用层 4](#_Toc357336818)

[2 接口设计 4](#_Toc357336819)

[2.1 外部接口（用户接口） 4](#_Toc357336820)

[2.2 内部接口 5](#_Toc357336821)

[3 开发环境 5](#_Toc357336822)

[3.1 硬件环境 5](#_Toc357336823)

[3.2 软件环境 5](#_Toc357336824)

# 总体设计

## 系统描述

### 需求规格概述

中段反导过程的仿真：在侦察到敌方弹道导弹后，我方发射拦截导弹进行拦截。程序需对两方导弹的运动特性进行模拟，包括导弹的运动轨迹，微小的扰动，以及拦截弹的主动制导等要素。并需输出每帧红外图像，供另外的分析程序进行分析。

输入：目标的各种运动参数，传感器配置参数等。

输出：屏幕上显示整个仿真过程，并将从见到目标到相遇的连续红外图像序列保存到磁盘。

性能：每秒50帧。

### 运行环境规定

操作系统：Windows XP / Windows 7

硬件：中高等配置

## 系统结构和处理流程

### 系统结构

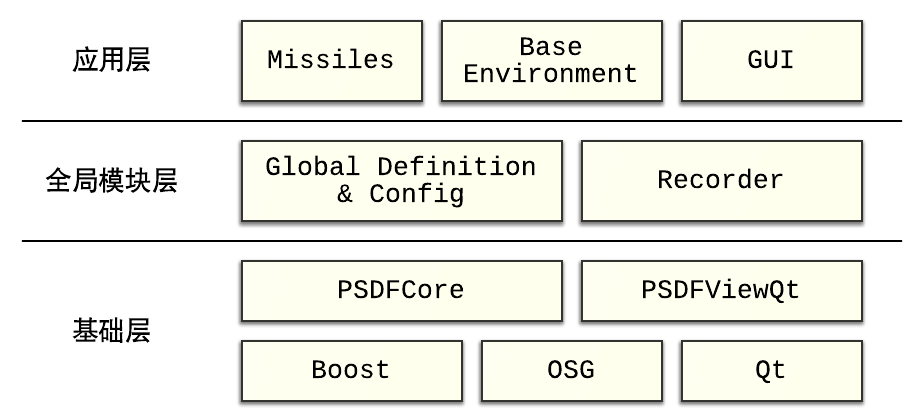


图 1 系统结构

### 基础层

最底层为第三方基础库，其中：

* OSG，全称OpenSceneGraph， 是一个强大的跨平台渲染引擎，基于OpenGL实现。
* Boost，作为C++标准库的扩展，具有非常广泛且方便的功能集合。
* Qt，一个广泛使用的跨平台窗口系统。

上层的PSDFCore和PSDFViewQt是为了仿真系统而开发的通用框架，里面提供了插件机制，事件机制，场景管理，网络数据收发等基础服务，为开发仿真系统提供了便利。

基础层属于已有模块，直接在程序中调用即可，程序主要实现全局模块层和应用层。

### 全局模块层

该层的模块和基础层的模块一样是全局可见的。向所有插件提供了与该应用密切相关的服务，例如事件的定义，全局数据结构的定义，以及全局的配置等。另一个录像机模块，提供了场景的播放/暂停，以及跳转到某一时间点的机制。并负责将每一帧的图像储存起来并通过一定机制发送给其他的分析处理程序。

### 应用层

本程序主要有两个插件：Missiles插件和Base Environment插件，和一个包含GUI的主程序。

其中Missiles插件负责进行攻击导弹和拦截导弹的仿真。主要包括运动学仿真，红外成像仿真，以及红外探头的误差仿真等。

Base Environment插件负责为态势搭建基础的环境。如地球，星空，可能还会有太阳和月亮。

主程序主要提供用户接口以及调用所有插件，完成所需求的功能。

# 接口设计

## 外部接口（用户接口）

对于每次仿真，用户需要输入：

* 侦测到敌方导弹时，敌方导弹速度，仰角，偏角。
* 拦截导弹关机时速度，和其它跟动力学相关的参数。
* 红外相机参数，如成像速度，成像分辨率，盲元率等。

屏幕上主要显示两个窗口，一个为总体态势，用户可在其中自有漫游，随意观察导弹，地形等。另一个为红外图像窗口，显示红外传感器每帧捕获到的图像，用户不能在该窗口进行漫游操作。

另外程序还提供仿真进度条，从我方检测到敌方导弹开始，到摧毁敌方导弹结束。并配以“播放/暂停”按钮，在仿真过程中，用户可以点击“暂停”按钮暂停仿真，点击“播放”按钮继续仿真，也可以直接点击仿真进度条跳转到某个时间点。

## 内部接口

对Recorder主要定义如下接口：

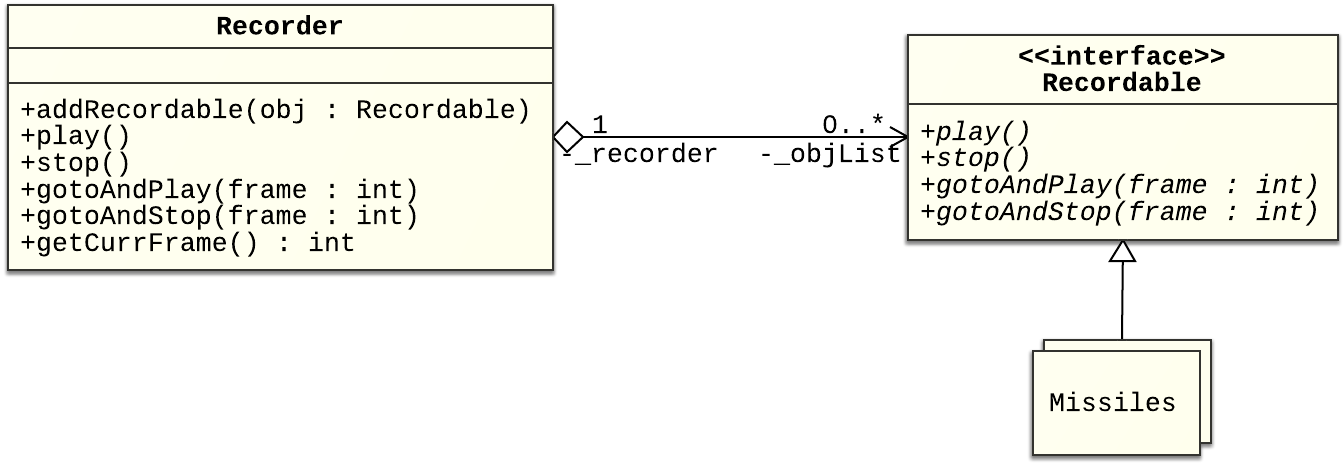


图 2 内部接口

其中Recorder类维护了Recordable列表，Recordable接口定义了所有跟回放相关的方法，有具体类根据情况实现，在本程序中，Recordable的实现类主要为导弹类，包括敌方的弹道导弹和我方的拦截弹。Recorder类中的方法是用来响应UI上按钮的点击事件。程序开启后，Recorder遍历Recordable列表，调用每个元素的getReady()方法，让每个元素进行相应的准备工作。每当用户点击播放按钮时，Recorder中的play()方法被调用，然后Recorder会依次调用每个元素的play()方法，在每个元素实现play()方法时，先调用Recorder的getCurrFrame()方法取得当前帧号，在根据帧号进行相应的行为。

# 开发环境

## 硬件环境

教研室PC机，单机环境。

## 软件环境

Windows 7；Visual Studio 2008 sp1；Qt；Boost