# 第一次迭代

系统分解为两部分，分别是部署在卫星上的部分和部署在地面上的部分。

卫星部分：

底层支持：提供相机的硬件驱动，并与卫星操作系统交互，依赖通信服务来升级硬件驱动和操作系统。

拍摄模式存储：存储经过定义的拍摄模式，依赖通信服务来增删用户定义的拍摄模式。

图像生成模块：通过相机的传感器读数生成原始图像，依赖底层支持来获取硬件读数。

拍摄模式控制：按照存储的拍摄模式，控制拍摄过程，依赖通信服务来接受拍摄指令。

卫星端通信服务：维护通信协议标准，与地面进行通信，接收指令，发送图像。

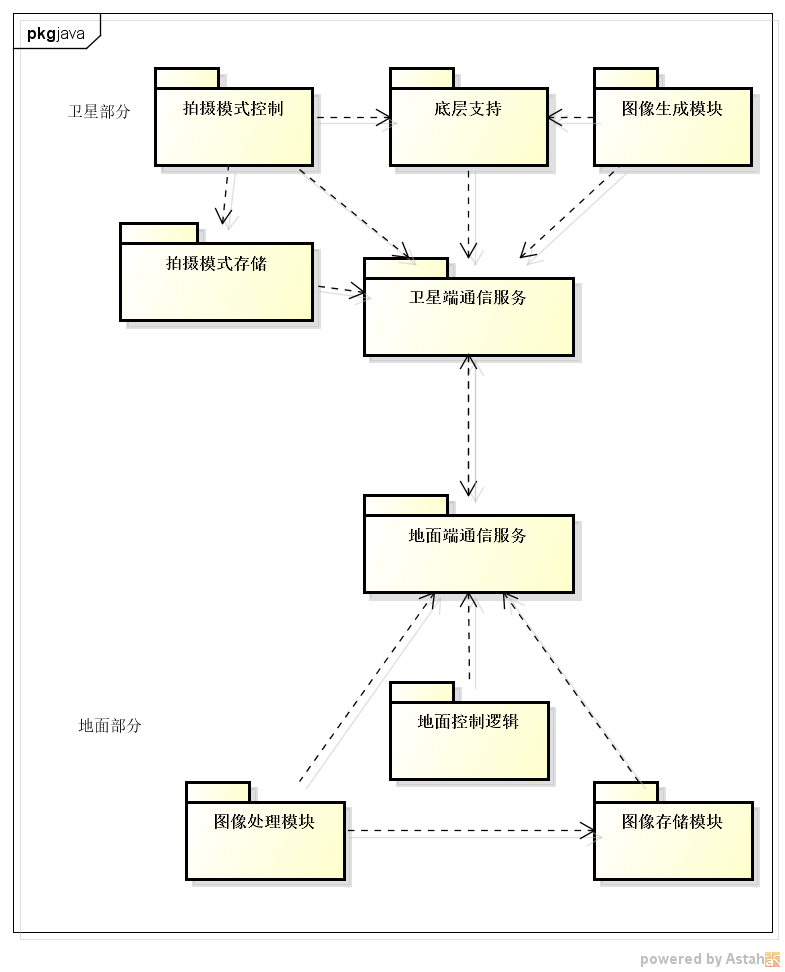
地面部分：

地面端通信服务：维护通信协议标准，与卫星上的相机系统进行通信，接收图像，发送指令。

地面控制逻辑：向用户提供发送控制指令的功能，控制相机行为。

图像存储模块：将接收的图像存储在服务器上，允许有权限的用户访问和修改，依赖通信服务来接收图像。

图像处理模块：处理接收的图像，依赖通信服务来即时处理原始图像，依赖图像存储模块来处理接收后的图像及存储处理后的图像。



# 第二次迭代

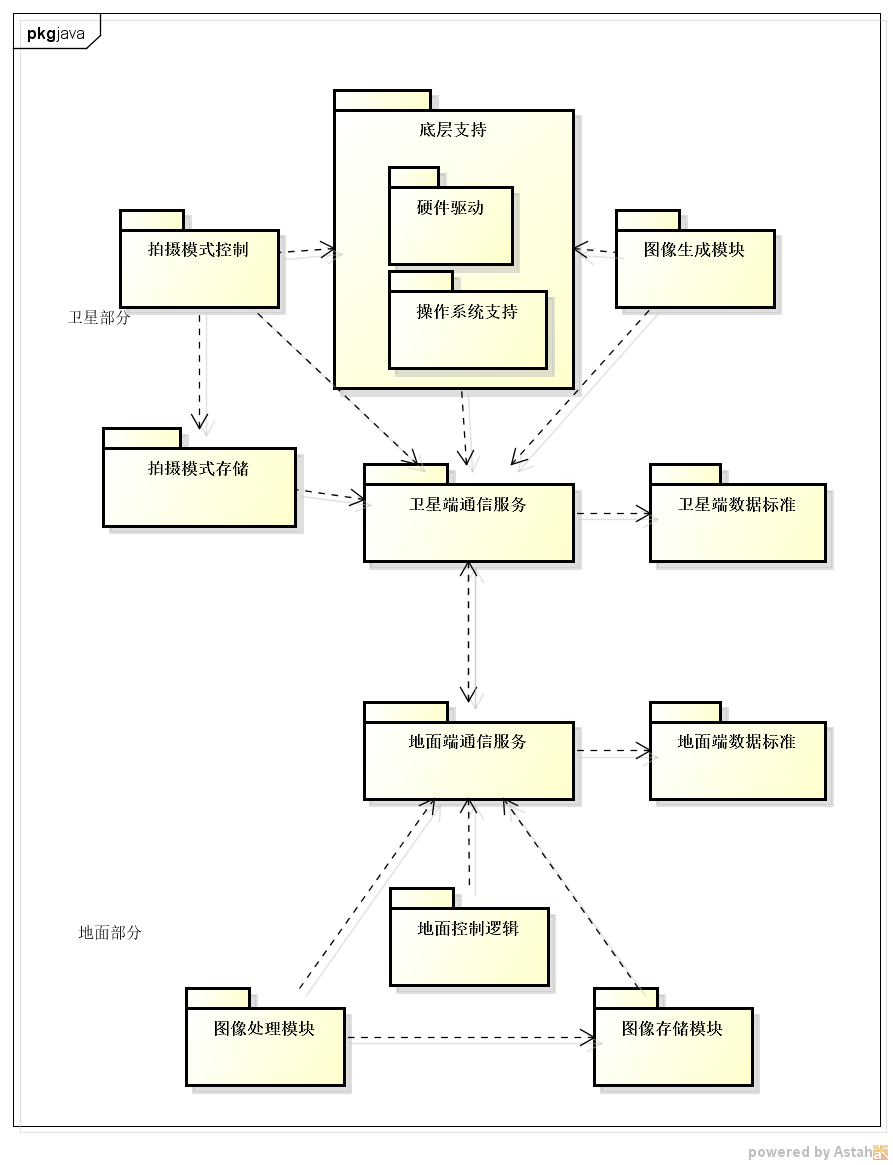
## 选择ASR

第二次迭代选择的是可修改性，因为对于该系统，硬件和操作系统的升级是可预见的，图像格式和数据标准也是存在潜在变化的。而且卫星发射后对软件系统的修改显然只能远程进行，易修改性显得尤为重要。

## 候选策略表

|  |  |
| --- | --- |
| 模块分离 | 采用。分离出数据标准模块，维护通信协议的标准不再由通信标准承担，将可能发生变更的部分独立出来，以应对数据标准的潜在变更。 |
| 增强内聚 | 采用。将底层支持模块分成硬件驱动和操作系统支持模块，前者提供相机的硬件驱动支持，以应对硬件升级需要。后者为系统提供卫星操作系统的接口，实现和卫星其他部分的交互，以应对操作系统变更。 |
| 重构 | 未采用。重构更侧重于维护可变更性的方法，而非体系结构设计决策。 |
| 延迟绑定 | 采用。具体措施同“模块分离”。分离出数据标准模块，专门维护数据标准，而非将其写死在系统中，本身也是延迟绑定的措施。 |
| 封装 | 未采用。在体系结构的层面上体现不明显。 |
| 使用中间件 | 未采用。容易造成性能损失。 |

## 第二次迭代结果



# 第三次迭代

## 选择ASR

第三次迭代选择的ASR是安全性。涉及航天工业，硬件昂贵，数据价值高，安全性不容忽视。若假定本卫星相机系统是用于国防侦查所用（所谓拍摄模式，即是为了长时间周期性对某一地点进行拍照侦查所需），安全性则尤为重要。

## 候选策略表

|  |  |
| --- | --- |
| 拒绝可疑访问 | 未采用。由实际系统特性决定，本系统用户是少数专业用户，而非面向大众的网络服务。 |
| 用户认证 | 采用。由于系统可能涉及机密信息，对用户的身份认证是必须的。在地面端增加用户认证模块，以加密方式存储用户资料，用户向卫星发布控制指令必须经过身份认证。 |
| 数据加密 | 采用。由于地面和卫星的通信可能被拦截和监听，加密是必须的。 |
| 攻击发生时收回数据访问权限 | 采用。增加攻击侦测模块，当侦测到攻击发生时，图像存储模块将会拒绝一切外界访问，直到确认安全。 |
| 通过检验和或哈希值验证数据完整性 | 未采用。假定在地面和卫星的通信过程中已经采取了完整性验证手段，是非常合理的。 |

## 第三次迭代结果

