Pros&cons

SOA架构

Pros：SOA架构对系统内部来说降低了系统内部组件的耦合程度，也提高了系统的可修改性。但对系统外部来说SOA架构提高了系统的互操作性。客户端只需要通过Service registry查询服务的接口就可以访问服务，所以系统一方面可以很方便地向Service registry注册自己可以提供的服务，比如购票服务，另一方面又可以向Service registry查询自己需要的外部服务，比如支付服务等。

Cons：在外部访问服务时，SOA架构需要对内部相关子服务的协作进行调节，在设计上提高了复杂性。同时，和服务相关的内部组件可能会对整体服务本身产生影响，即某个子服务的性能也会影响整个系统的性，造成瓶颈。此外，SOA架构的安全性、性能等也要依赖于中间件的设计，给系统架构的过程带来额外的设计成本。

Broker架构

Pros：Broker架构允许将服务端部署到实现不同功能的服务器，可以减轻server端的压力，将server端进行分布化，也使得server端更容易维护。同时也可以做到分发控制，负载平衡。Broker架构也有利于系统的安全性，可以过滤恶意请求。当系统需要变化时，由于客户端和服务端都只依赖broker，使得系统的可修改性和可扩展性都很好。

Cons：由于broker作为客户端和服务端的中间层，所以对系统的通信造成一定的影响。同时，当broker出现问题时，会对整个系统造成严重的影响。

Conclusion:

对于使用本系统的用户而言，性能和安全性是最需要的。Broker架构可以进行分发控制和负载平衡，减轻服务器压力，有效提高了系统的性能。同时broker可以有效截取并过滤客户端请求，提高系统的安全性。而SOA架构只能通过中间件来提高系统性能以及安全性。为了更好地满足ASR，我们决定采用Broker架构对系统进行设计。