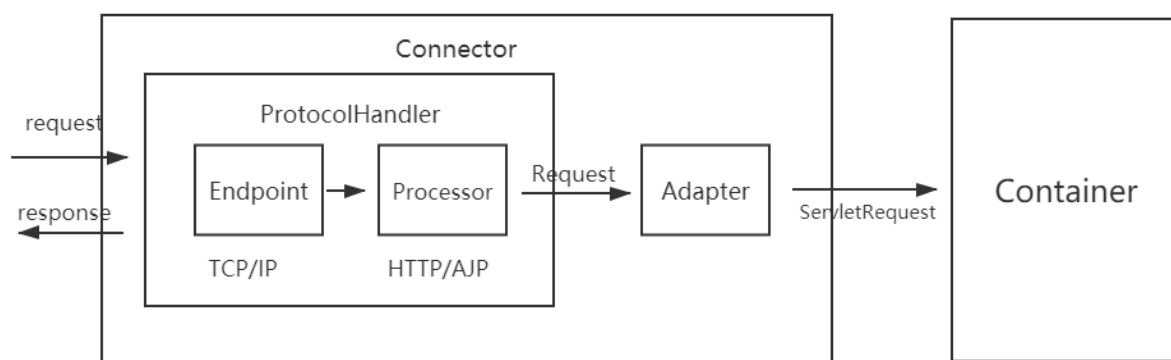


Tomcat体系结构

Tomcat 设计了两个核心组件连接器（Connector）和容器（Container）来完成 Tomcat 的两大核心功能。

连接器，负责对外交流：处理Socket连接，负责网络字节流与Request和Response对象的转化；

容器，负责内部处理：加载和管理Servlet，以及具体处理Request请求；



连接器组件Coyote

Coyote 是Tomcat 中连接器的组件名称，是对外的接口。客户端通过Coyote与服务器建立连接、发送请求并接受响应

EndPoint

EndPoint 是 Coyote 通信端点，即通信监听的接口，是具体Socket接收和发送处理器，是对传输层的抽象，因此EndPoint用来实现TCP/IP协议的。

Processor

Processor 是Coyote 协议处理接口，如果说EndPoint是用来实现TCP/IP协议的，那么Processor用来实现HTTP协议，Processor接收来自EndPoint的Socket，读取字节流解析成Tomcat Request和Response对象，并通过Adapter将其提交到容器处理，Processor是对应用层协议的抽象。

ProtocolHandler

Coyote 协议接口，通过Endpoint 和 Processor，实现针对具体协议的处理能力。

Tomcat 按照协议和I/O 提供6个实现类：AjpNioProtocol，AjpAprProtocol，

AjpNio2Protocol , Http11NioProtocol, Http11Nio2Protocol ,
Http11AprProtocol。

Adapter

由于协议不同，客户端发过来的请求信息也不尽相同，Tomcat定义了自己的Request类来封装这些请求信息。ProtocolHandler接口负责解析请求并生成Tomcat Request类。但是这个Request对象不是标准ServletRequest，不能用Tomcat Request作为参数来调用容器。Tomcat设计者的解决方案是引入CoyoteAdapter，这是适配器模式的经典运用，连接器调用CoyoteAdapter的Service方法，传入的是Tomcat Request对象，CoyoteAdapter负责将Tomcat Request转成ServletRequest，再调用容器。

Tomcat Servlet 容器 Catalina

Tomcat就是一个Catalina的实例，因为Catalina是Tomcat的核心

Tomcat/Catalina实例

