# http://camel.apache.org/

# Introduction:

Apache camel is a mediation framework that provides a layer of indirection (middleman) in integrating different systems.

There are two problems that are faced in integration different systems: the message formats that are exchanged between systems, and the protocols involved in moving messages between them.

## History:

* Apache camel is based on ServiceMix which is in turned based on JBI (java business integration specifications)

JBI based on MEP (Message exchange pattern) that comes from WSDL.

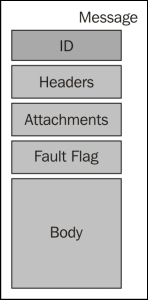
JBI’s disadvantages:

* All message exchange based on NMR (normalized messages router)
* NMR is based on XML. So message exchange has to be confined to XML
* Bad for performance because of marshalling/umarshalling of messages
* Definitions , components scattered in different packages
* Another foundation of Camel is EIP (Enterprise Integration Patterns) Gregor Hohpe and Bobby Woolf.

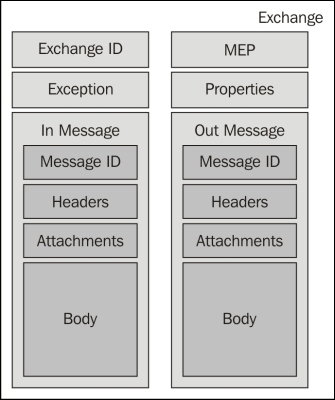
*This is what Camel is: an open source framework that allows you to integrate systems and that comes with a lot of connectors and****Enterprise Integration Patterns****(****EIP****) components out of the box. And if that is not enough, one can extend and implement custom components.*

# Core Concepts:

## Message:

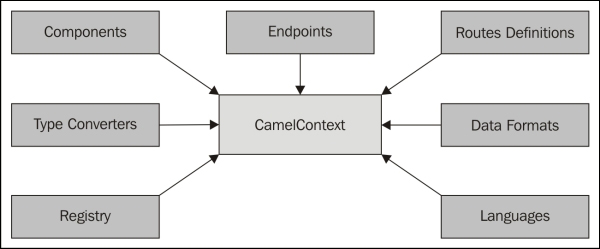


## Exchange:



## Camel Context:

The Camel context is the runtime system and the loading container of all resources required for the execution of the routing.



## Processor:

A processor is a node in the routing which is able to use, create, or modify an incoming exchange. During routing, the exchanges flow from one processor to another. This means all Enterprise Integration Patterns (EIP) are implemented using processors in Camel.

## Routes:

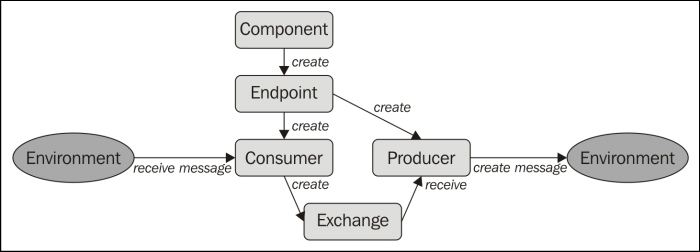
The Camel route is the routing definition. It's a graph of processors. The routes (routing definition) are loaded in the Camel context. The execution and flow of the exchange in a route is performed by the routing engine.

## Channels:

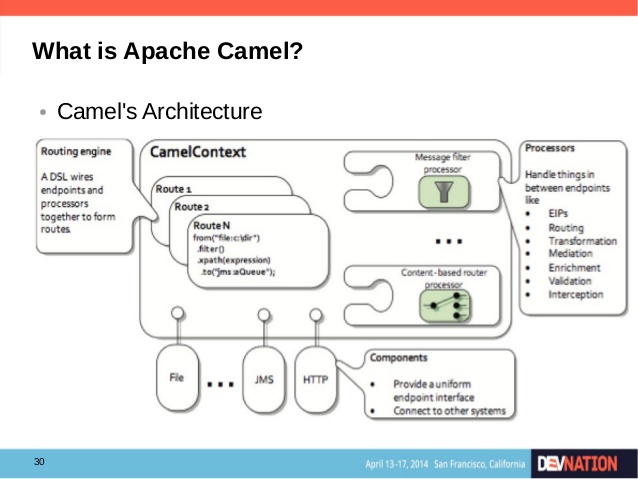
In every Camel route, there is a channel that sits between each processor in the route graph. It's responsible for the routing of an Exchange to the next Processor in the graph.

## Component, endpoint, producer, and consumer

The components are the main extension points in Camel. We don't directly use a component in a route, we define an endpoint from the component. This means a component acts as a factory for endpoints as follows:



# Camel Architecture:



# Routes And Processors

Actually, all Camel Exchange Integration Patterns (EIPs) are implemented using processors.