



原创:21 翻译:0 转载:15

博客 | 图库 | 写博文 | 帮助

首页 | C++语言 | Java | 数据结构/算法 | Tomcat | SOA | Web Service | Bpel | Xml | Javascript | 网站开发 | 网络相关 | 文章札记 | Linux

panpan3210 的BLOG

分享到: MSN/QQ

论坛

开心

人人

豆瓣

新浪微博

收藏 +

博主的更多文章&gt;&gt;



写留言

邀请进圈子

发消息

加友情链接

进家园 加好友

博客统计信息

51cto推荐博客

用户名: panpan3210

文章数: 36

评论数: 120

访问量: 196942

无忧币: 630

博客积分: 980

博客等级: 4

注册日期: 2008-09-24

热门文章

ofstream和ifstream详细用法

使用Eclipse+Axis2构建We..

C++ XML解析之TinyXML篇

Tomcat源码学习(一) 源码..

关于编译错误 fatal erro..

标准库中的map和set

C++文件操作 判断文件是..

如何将数字(包括double型..

搜索BLOG文章

## 原创 Mule简介

2009-02-20 18:22:18

标签: SOA ESB Mule

原创作品，允许转载，转载时请务必以超链接形式标明文章 [原始出处](http://panpan.blog.51cto.com/489034/132100)、作者信息和本声明。否则将追究法律责任。  
任。 <http://panpan.blog.51cto.com/489034/132100>

笔者一直在研究和开发ESB相关项目，最近一周学习了Mule项目，对Mule开源ESB项目有了一定的认识。本文写一些mule的简介。在下一篇文章中介绍了Mule的安装部署和使用例程。

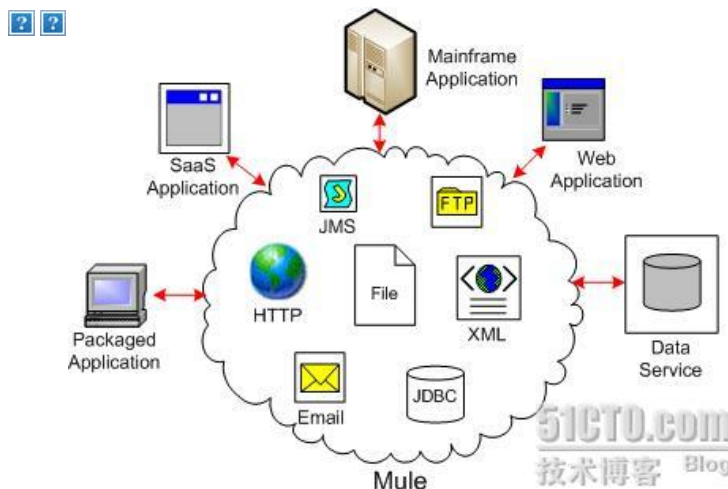
### Mule是什么？

Mule是一个轻量级的基于Java的ESB消息框架，它允许用户快捷地连接多个应用并且在这些应用之间交换数据。Mule使用了SOA的体系结构思想，可以方便的集成已有的应用。它是可升级的、高分布式的对象代理，可以通过异步传输消息技术来无缝的处理服务与应用之间的交互。

Mule框架提供了一个可升级的环境，可以把自己的业务组件部署在里面。Mule管理所有组件之间的交互，不管它们是在同一个虚拟机中还是在internet上，也不管底层使用的传输方式。

Mule围绕着企业服务总线（ESB）架构进行设计，保证了不同的组件或者应用可以通过公共的消息总线进行交互，公共的消息总线一般是由JMS或者其他消息服务器来实现。

在应用中会使用不同的技术，包括JMS, Web Services, JDBC, HTTP等等，Mule可以很好地处理他们之间的交互。



Mule有以下的优点：

我的技术圈(11)

更多>>

网络工程师联盟

linux-北京圈

JAVA EE

Microsoft技术

网络工程及技术

网行天下

最近访客



olivajoy



goodzmq



share..



gxujin



canglin



a5556..



0319z..



jc5566



benbe..



dallasyu



huhan..



cjc110

最新评论

zhaogangz9: 我找了ActiveBpel Designer很久了, ..

[匿名]maxim\_sin: 多谢楼主 这个问题搞了我两天 现在..

squirrel09: 博主, 在ant download的时候报如下..

ld3838029: 有没axis2的中文API。。。

[匿名]a: 非常好! 终于编译通过了。执行ant命..

51CTO推荐博文

更多>>

SQL2005 express升级到2008企业版..

mysql中文乱码,无法插入中文的解..

关于自动化测试的一些思考1

Android超简单短信接收[中级编]

SQL2008空间数据应用[系列]数据表..

Android网络编程之WebKit应用

(1)Mule组件可以是你要的任何类型。你可以很容易地集成一切从一个"plain old Java object" (POJO) 到另一个框架的组件。

(2)Mule和ESB模型允许组件重用。不像其他的框架, mule允许你使用一个已有的组件而不需要改变。组件不需要任何特定的mule代码, 并且没有要求的API。业务逻辑和消息逻辑保持完全分离。

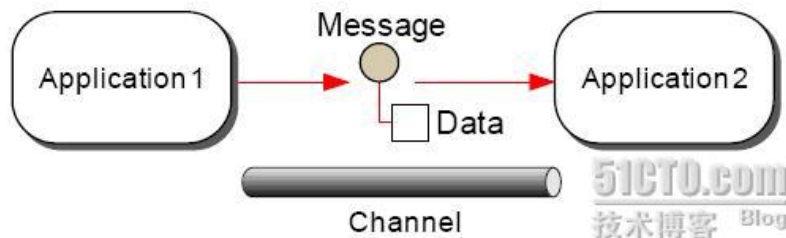
(3)消息可以是任何格式, 从SOAP到二进制图像文件。mule对体系结构不强制任何设计限制, 比如XML消息或者WSDL服务。

(4)你可以以多种拓扑结构开发mule, 不仅仅是ESB。因为它是轻量级的和可嵌入的, mule可以有效地减少到市场的时间, 并且增强项目参评, 比如安全性, 可扩展性。

mulesource提供了管理工具运行管理你的开发(Mule HQ)和基础设施(Mule Galaxy)。

## 理解消息框架

应用网络化的好处是使得一个应用能够发送数据到另一个应用。但是许多应用不能够读取和处理另一个应用的数据。Mule通过在应用之间提供一个消息框架以消息的方式读取、转换和发送数据来解决这个问题。一个消息是一个数据包, 它可以在一个特定的管道(channel, 也成为是一个queue)中处理和发送。如下图所示:

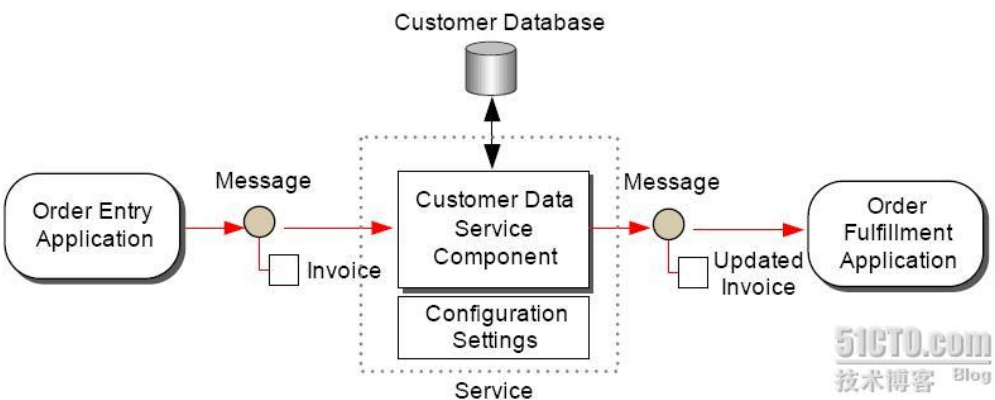


## 理解Mule体系结构

该部分讨论了Mule体系结构的不同部分和他们如何处理消息和数据。这里使用了一个例子: 一个公司需要为客户订单(order)产生发票(invoice), 对这些发票执行一些处理后, 把它们发送到运输部门以满足订单。



处理过程:



## 把业务逻辑和消息分离

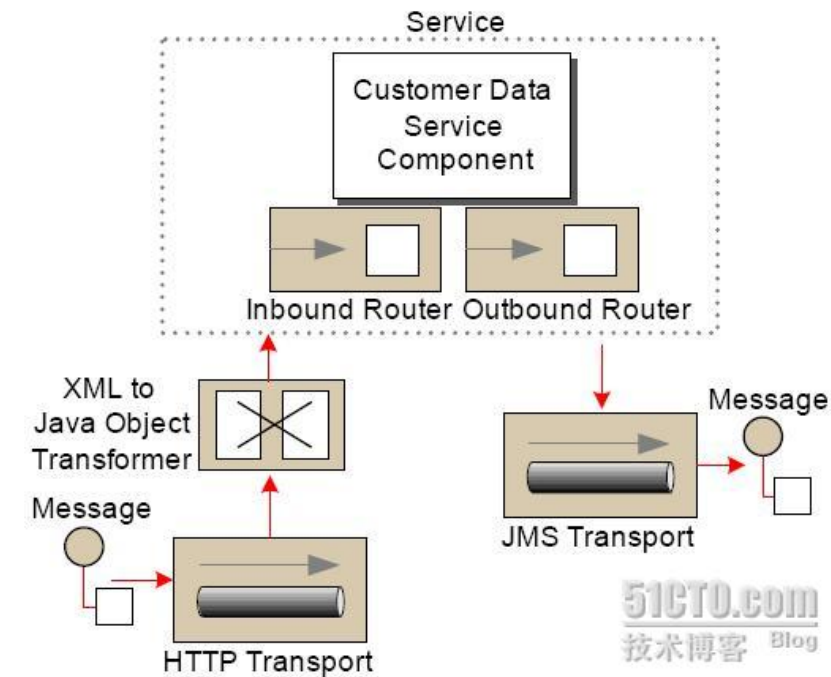
Mule的一个优点是它能处理通过一系列协议发送的消息。例如, 一个发票(invoice)可能是XML格式的, 但是它可能通过HTTP到达, 也可能是作为一个JMS消息, 这取决于哪个应用创建了这个发票(invoice)。如果服务组件仅仅处理业务逻辑和数据, 而不是消息本身, 它是如何读取各种格式的消息的呢?

答案是: 服务组件不知道如何读取消息, 因为默认地, 服务组件和消息格式是屏蔽的。取而代之, 一个transport单独携带消息, transformer在router传递这个消息到服务组件之前改变消息的负载(比如刚才的invoice), 它按照需要格式化使得这个服务组件可以读取。例如, 如果一个XML invoice通过HTTP被发送, HTTP transport携带这个消息, router指导这个消

友情链接

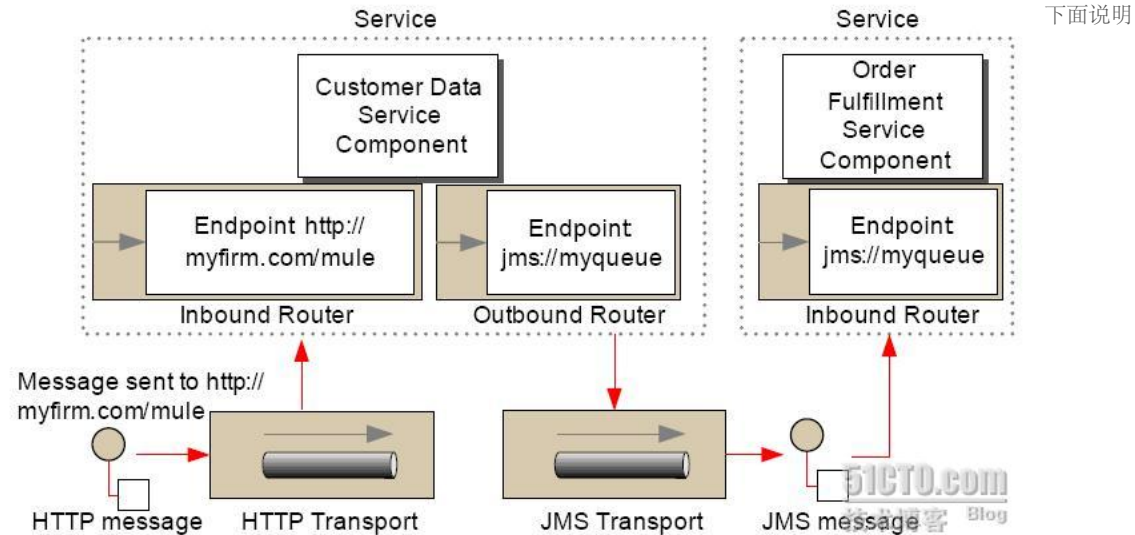
个性塑造  
李云  
默默奋斗的人  
龍々2046→  
Nathan的技术空间  
lgzeng  
polaris  
小松  
子子🐾  
51CTO博客开发

息到达每一个服务组件以处理它，并且transformer按需要沿路改变这个invoice(比如从XML到一个Java对象)。所有的这些transporting,transforming和routing对于消息组件是透明的。



把所有的组织起来

Endpoint是配置的元素，是把所有服务组织起来的关键。你在inbound和outbound routers指定endpoint来告诉Mule使用哪个transport，把消息发送到哪里和服务组件需要接收哪个消息。一个endpoint的主要部分是地址(address)，表达成URI，表示了使用的transport、地址和任意额外的参数。



下面说明

上图所示的逻辑数据流：

- (1)用户放一个order到公司网站，并且一个invoice被创建成一个XML格式并且提交到http://myfirm.com/orders。
- (2)HTTP transport接收这个XML invoice并且把它包装成一个Mule message。这个用户数据服务的inbound endpoint设置成http://myfirm.com/orders，并且它的inbound router指定这个消息必须包含一个Java对象，所以HTTP transport准备转换这个XMLInvoice并且分发这个消息到这个服务。
- (3)XML到对象 transformer把XML invoice转换成一个Java对象。
- (4)这个transport把这个消息传给用户数据服务。
- (5)用户数据服务组件查询master customer database（数据库）以获取关于永和的额外数据，并且更新invoice的数据。

- (6)HTTP transport使用outbound router配置来决定它必须分发消息到http://myfirm.com/verify。
- (7)HTTP transport使用Inventory Verification服务的inbound router配置来获取消息并且把它传输到服务组件。
- (8)服务组件更新invoice，有那个warehoused的一个ID code，这个仓库有所有的现在库存的invoice。
- (9)outbound endpoint 指定一个JMS地址，所有这个JMS transport分发这个消息到order fulfillment application，这个应用挑选订单。

Mule中的几个名次解释：

1. Connectors：就是支持不同协议的连接器。例如：Http,FTP,Mail,Soap,JMS和MW等等.通常连接器有三种类型：只用于接收，只用于发送和两者皆可。

2. EndPoints Address：终端地址，类似于jms://topic:myTopic这样的东西。前面的 j m s 就是连接器的类型。后面是各个连接器能识别的地址。这个地址可以是接收器识别的地址，例如：jms://topic:myTopic表示对myTopic这个队列进行监听。也可以是发送器识别的地址。例如 pop3://user:password@mail.mycompany.com将某一个消息发送到远程邮箱里面。

注意这里的地址是你自己命名的，通常不同的应用有不同的地址，例如某个应用是jms://topic:myTopic。另一个是jms://topic:myTopic2

3. UMO Components：可以想象成这是ESB总线上的一个个芯片，控制器。在Mule里面，他们是一些POJO，负责接收消息，然后进行处理，在发送出去。例如，一个UMO它的作用是监听jms://topic:myTopic里面的消息，如果有，加以处理后发送到pop3: //user:password@mail.mycompany.com这个邮箱里面。

显而易见，UMO通过简单的调用EndPoints Address来接收消息和转发消息。它不需要知道自己按照什么协议接收和发送消息。这就是ESB的作用所在，即提供统一的总线接口。

在下一篇文章中介绍了Mule的安装部署和使用例程。

本文出自 "C++技术" 博客，请务必保留此出处<http://panpan.blog.51cto.com/489034/132100>

 0人 了这篇文章

类别： [SOA](#) | [技术圈\(5\)](#) | [阅读\(1965\)](#) | [评论\(1\)](#) | [推送到技术圈](#) | [返回首页](#)

[上一篇 用VC进行屏幕截取编程](#) [下一篇 Mule安装与开发部署一个简单例子](#)

相关文章

- |                                     |   |
|-------------------------------------|---|
| <a href="#">评估企业是否部署SOA 做到勿临渴掘井</a> | <a href="#">SOA的竞争前沿—ESB技术发展与应用趋势</a>                 |
| <a href="#">SOA、BPEL、ESB的前生后世</a>   | <a href="#">浅析微软Service Layer Guidelines和OSOA架..</a>  |
| <a href="#">SOA的含义及ESB</a>          | <a href="#">ESB能够让SOA落地么？且看ESB的前世今生</a>               |
| <a href="#">SOA不是“万能钥匙” [SOA征文]</a> | <a href="#">面对SOA，企业CIO你准备好了吗？</a>                    |
| <a href="#">soa在学校信息管理的应用 SOA征文</a> | <a href="#">构建SOA 的IT捷径:BEA AquaLogic Service Bus</a> |

文章评论

[1楼]  [匿名]51CTO游客

2009-03-02 16:28:54

希望博主把mule的配置文件解释下，自己配配置文件都不知道该写那些？？

博主回复: 2009-03-02 16:42:44  
恩，这个需要到Mule的官方网站上看，而且你下载的Mule安装文件里自带的有例子，查看一下就行了，比较容易理解。

发表评论

昵 称:

[登录](#) [快速注册](#)

验证码:

请点击后输入验证码 [博客过2级，无需填写验证码](#)

内 容:

Copyright By 51CTO.COM 版权所有

**51CTO** 技术博客