



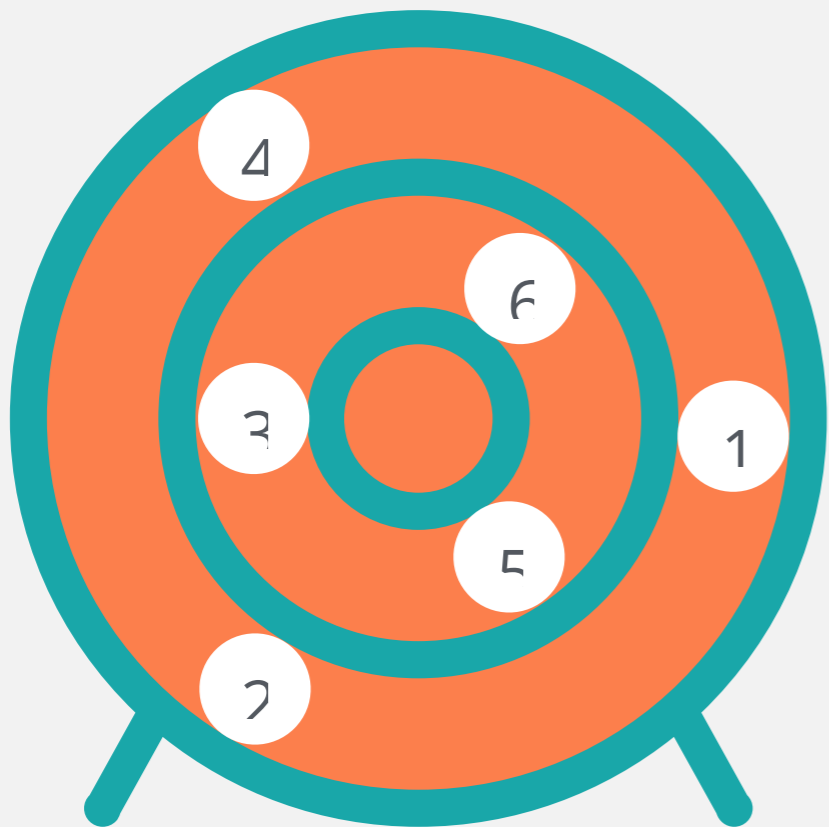
Thrift下的Node.js跨语言异构

来自前端的thrift实践

@刘欣

UC浏览器内核&中间件团队 * 前端工程师

分享内容



- 1 团队/技术背景介绍
- 2 为什么是thrift?
- 3 thrift框架 相关介绍
- 4 成果介绍 —hydra 转码开放平台
- 5 live code
- 6 问答环节

ok... UC中间件是做什么的 ...

压缩省流

运行在服务器端的
mozilla内核

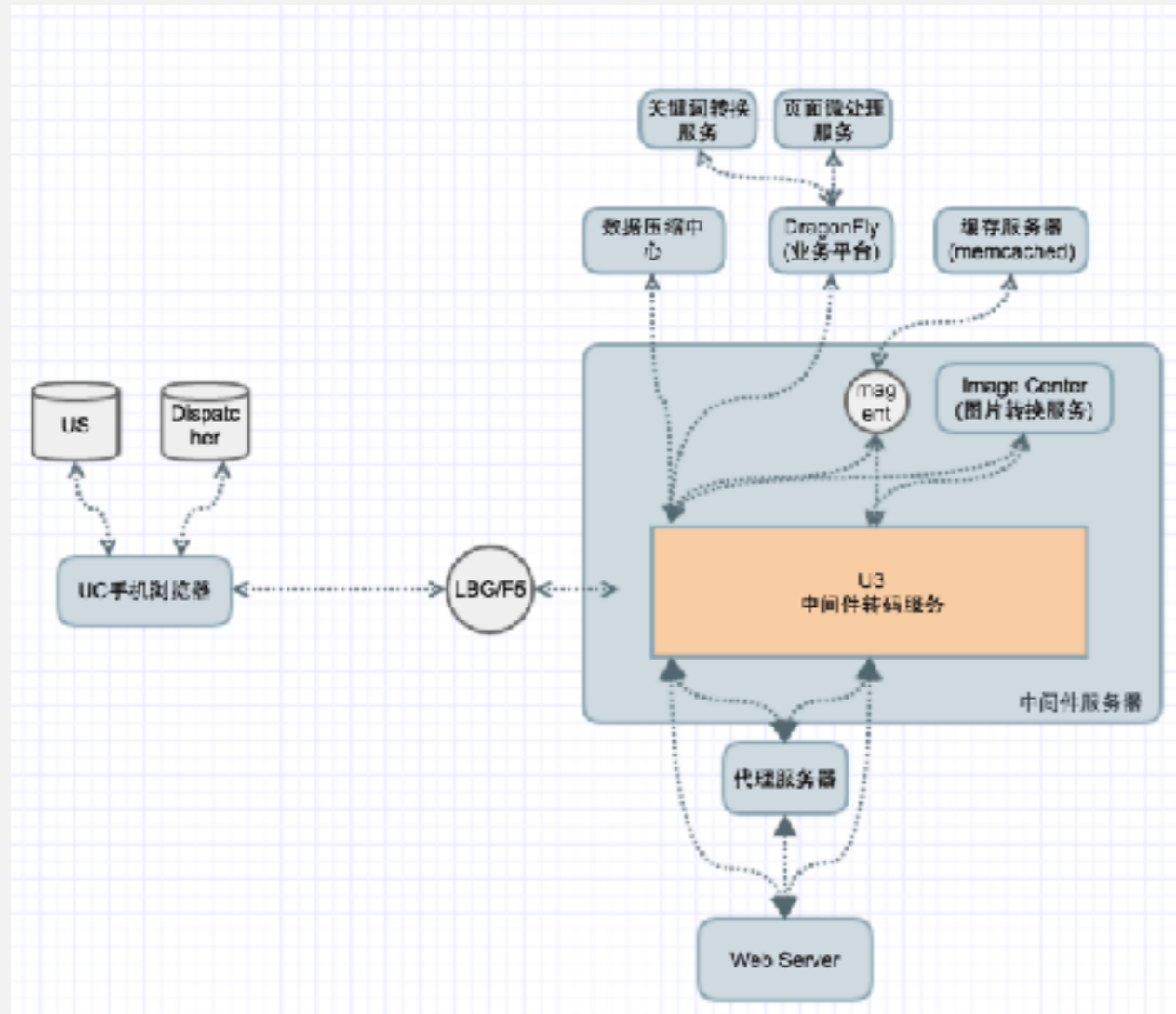
全后端团队

转码定制

历史
悠久

高并发
高可用

中间件架构图



组件繁多

云端浏览器内核

各类缓存交错

网络拓扑复杂

如何突围？

面对现状，我们如何改变？



业界现状

云端加速已经逐渐失去存在的土壤，市场和用户的需求从追求极致速度，变为追求极致的阅读体验



纯后端体系

中间件从基因上来说，是一个纯后端的产品，缺乏良好的对外接入体系，尤其是缺乏标准可开放的接口



前端应用如何融入

历史悠久的项目，如何快速让前端业务接入？进而将中间件服务器端变为一个对外提供数据能力的中台产品？



技术选型

原有系统接口，函数可复用
保持高可用，支持高并发
标准网络协议
尽可能的解耦



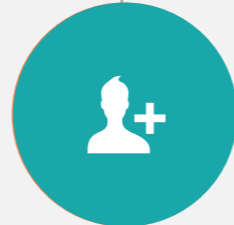
为什么是它？

Thrift 背景



2007 /

Facebook于2007年开发



2008 / Apache

2008年进入Apache开源项目



2014 / nodejs

2014年 官方开始支持nodejs



Now..

广泛运用于不同语言之间系统的异构, C++, python, java使用最多

Thrift的使用场景

什么场合下使用?



高并发

数据采用二进制格式传输，相对XML和JSON体积更小，对于高并发、大数据量环境更有优势



跨语言服务

大型项目下，极有可能调用跨语言服务，跨团队，甚至是外部企业合作，尤其明显

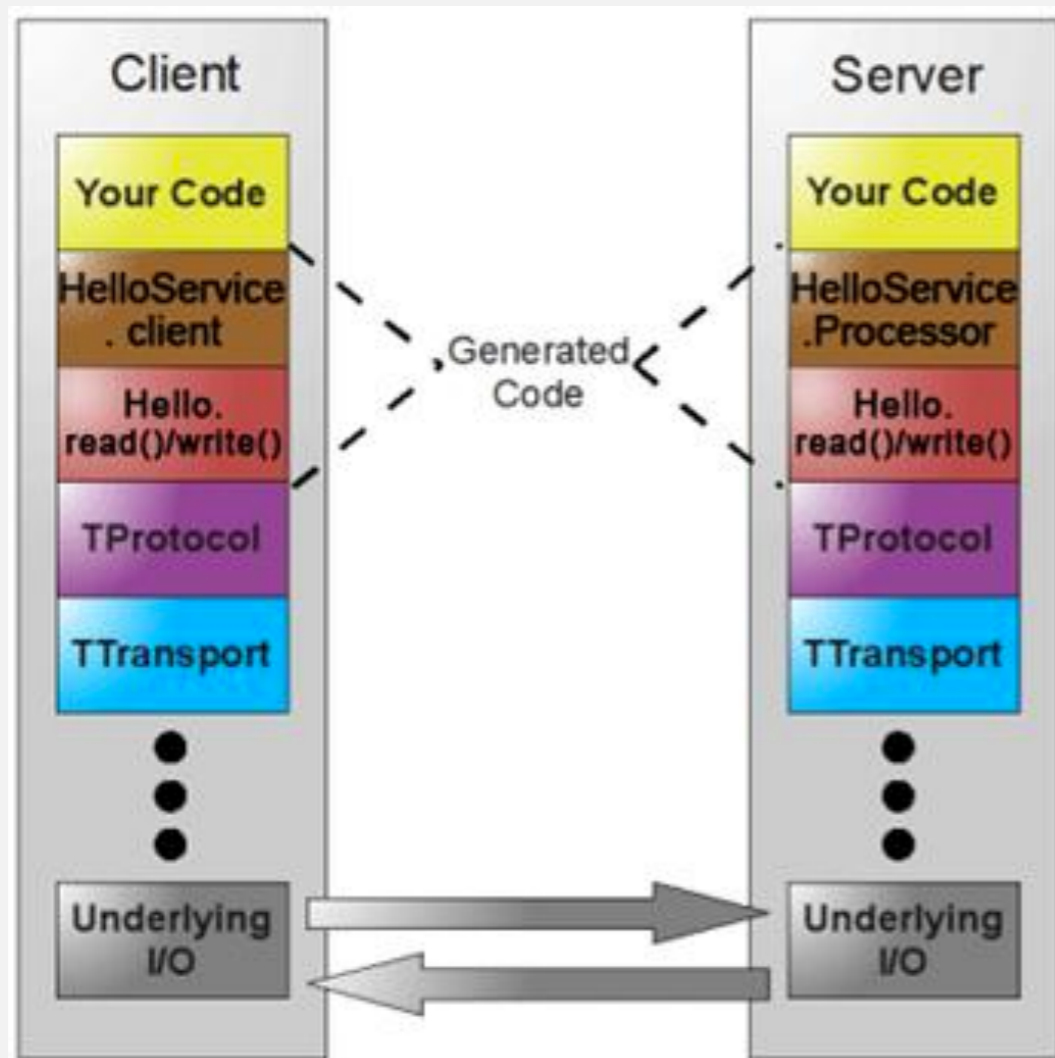


快速开发

代码自动生成，在熟悉Thrift后，开发速度有保障

Thrift的实现原理

它的架构如何？又是如何工作的？



图中黄色部分是用户实现的业务逻辑

褐色部分是根据 Thrift 定义的服务接口描述文件生成的客户端和服务端代码框架

红色部分是根据 Thrift 文件生成代码实现数据的读写操作。

红色部分以下是 Thrift 的传输体系、协议以及底层 I/O 通信，使用 Thrift 可以很方便的定义一个服务并且选择不同的传输协议和传输层而不用重新生成代码。

Rpc的优势

http restful也可以，为什么要选择Rpc?

支持多接口

有较多接口需要对接
系统的选择

01

部署灵活

RPC:远程过程调用。RPC的核心并不在于使用什么协议。RPC的目的是让你在本地调用远程的方法，而对你来说这个调用是透明的，你并不知道这个调用的方法是部署哪里。通过RPC能解耦服务，这才是使用RPC的真正目的。。

02

03

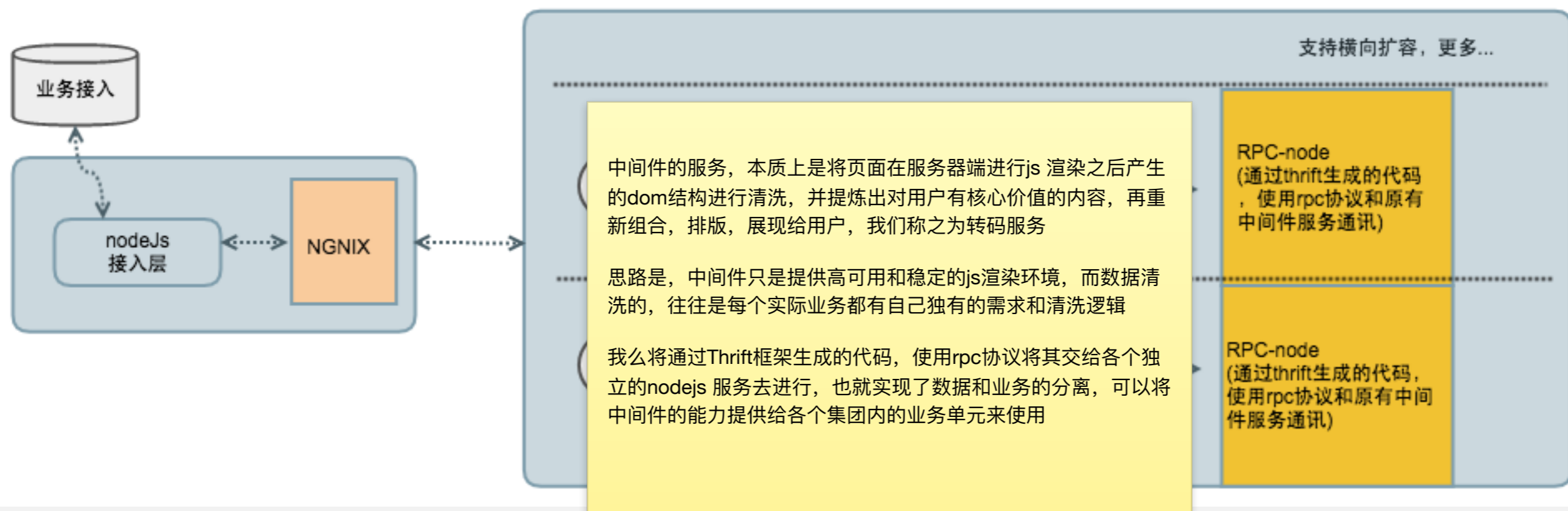
长连接



win

UC云端服务的Thrift运用

我们是如何用Thrift来解开前后端耦合的？



实际成果输出

Hydra转码开放平台——铁马爬虫系统

我们快速的中间件的js渲染能力输出给了业务方

业务方不用在本地使用phantomjs之类耗费大量本地资源，且难以被前端掌控的各类高并发，高可用问题

The screenshot displays the Hydra transcode platform interface. On the left, a configuration panel for a crawler is visible, titled '伽利略懂球'. It includes a 'Open URL' field with the value 'http://news.sohu.com/'. Below this, there are two 'Find' rules: one for 'Find Each' with the XPath expression '//BODY/DIV[@id='c']' and another for 'Find' with the XPath expression '//A/text()'. The 'Save' field for the first rule is set to 'title', and for the second rule, it is set to 'link'. There are also several '+' buttons for adding more rules. On the right, a '预览' (Preview) tab is active, showing the rendered output: '我是一个小铁匠，打马本领强 <[▽]>'. The interface also features a '源码' (Source Code) tab and a '运行' (Run) button.

性能展示

Hydra转码开放平台——铁马爬虫系统



性能

1. 平均页面抽取时间4.5s（在服务器端经过js渲染之后），同比phantomjs, electron都要快接近1倍

成功率

浏览器数据获取

99 %

业务数据获取

91 %

PI
YOU
G
RE



live code



微信 366348

