

ThoughtWorks®

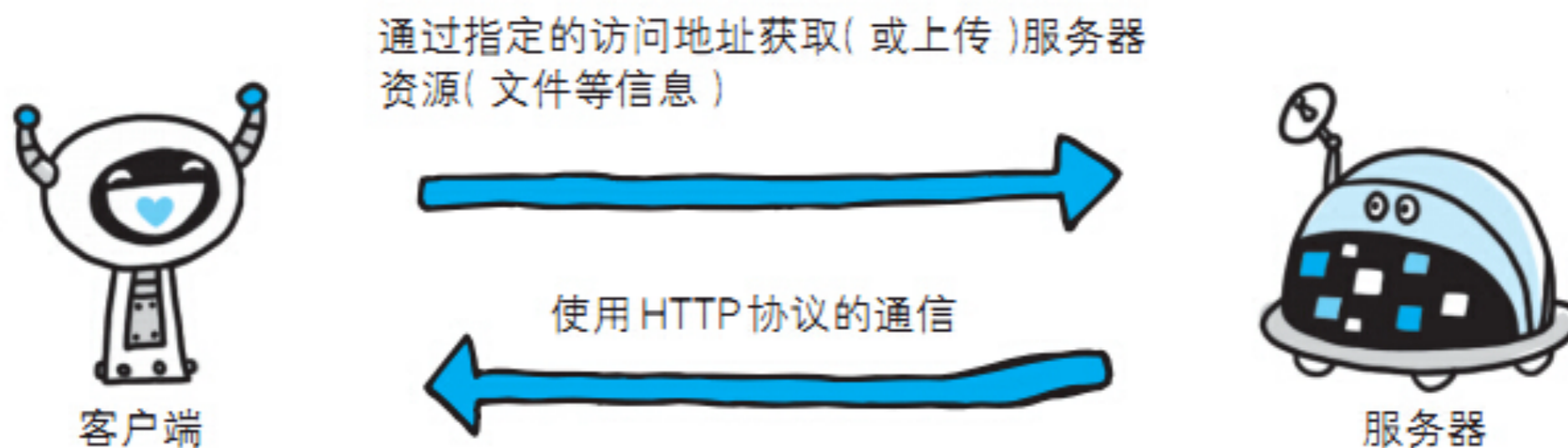
JimmyLv's

REST IN ACTION

*What's real RESTful Architecture &
How to Consume Resource in AngularJS?*

WHAT'S REST?

HTTP: HyperText Transfer Protocol



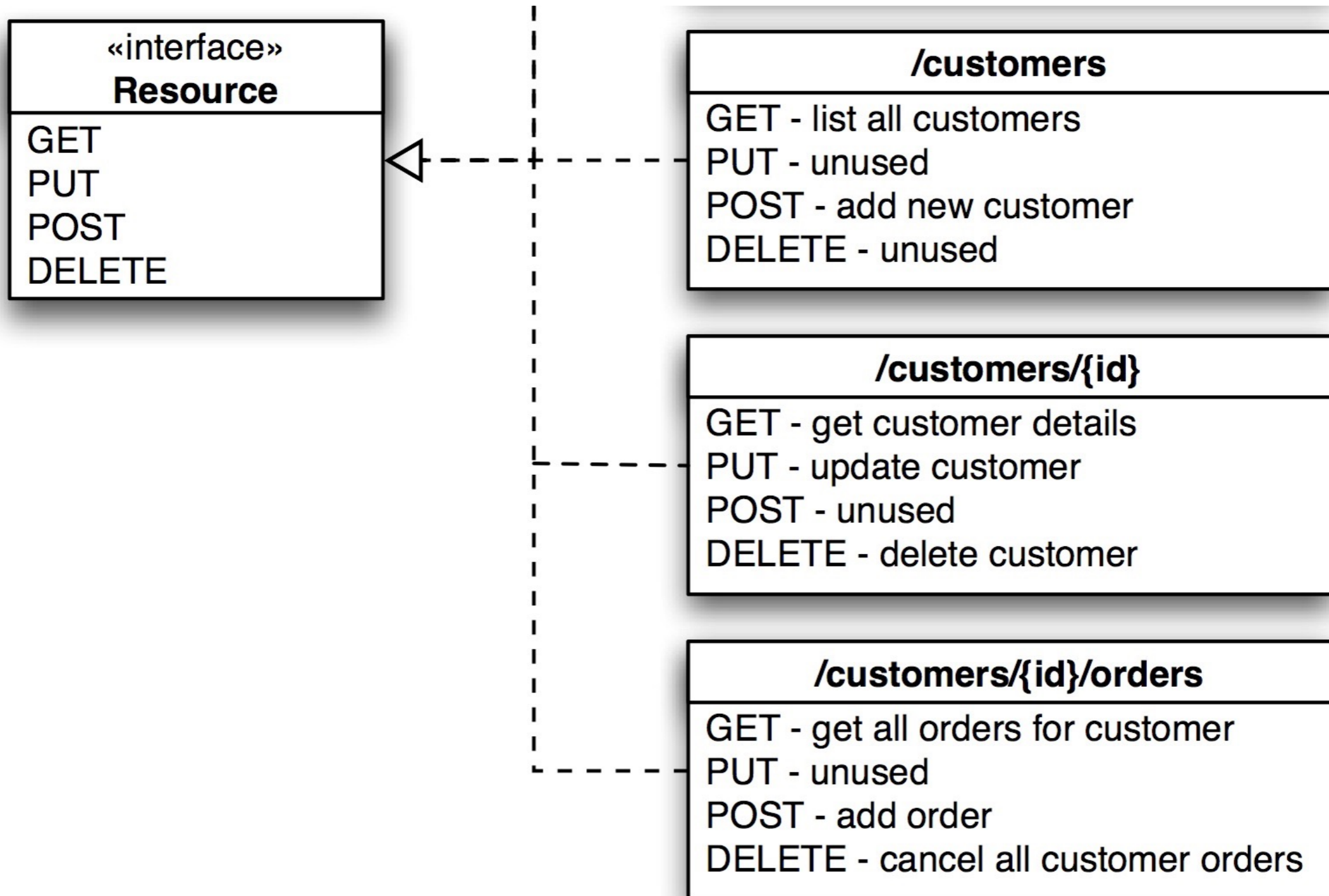
WHAT'S REST?

REST: REpresentational State Transfer

- Resource: 资源，即数据。比如policies, claims, customers, orders等；
- Representation: 某种表现形式，比如用HTML, JSON, XML, JPEG等；
- Representational State: 在客户端可被呈现的状态，服务器不保存通信状态；
- State Transfer: 通过转移和操作资源的表述，来间接实现操作资源的目的。

WHAT'S REST?

REST: (资源) 表述性状态转移



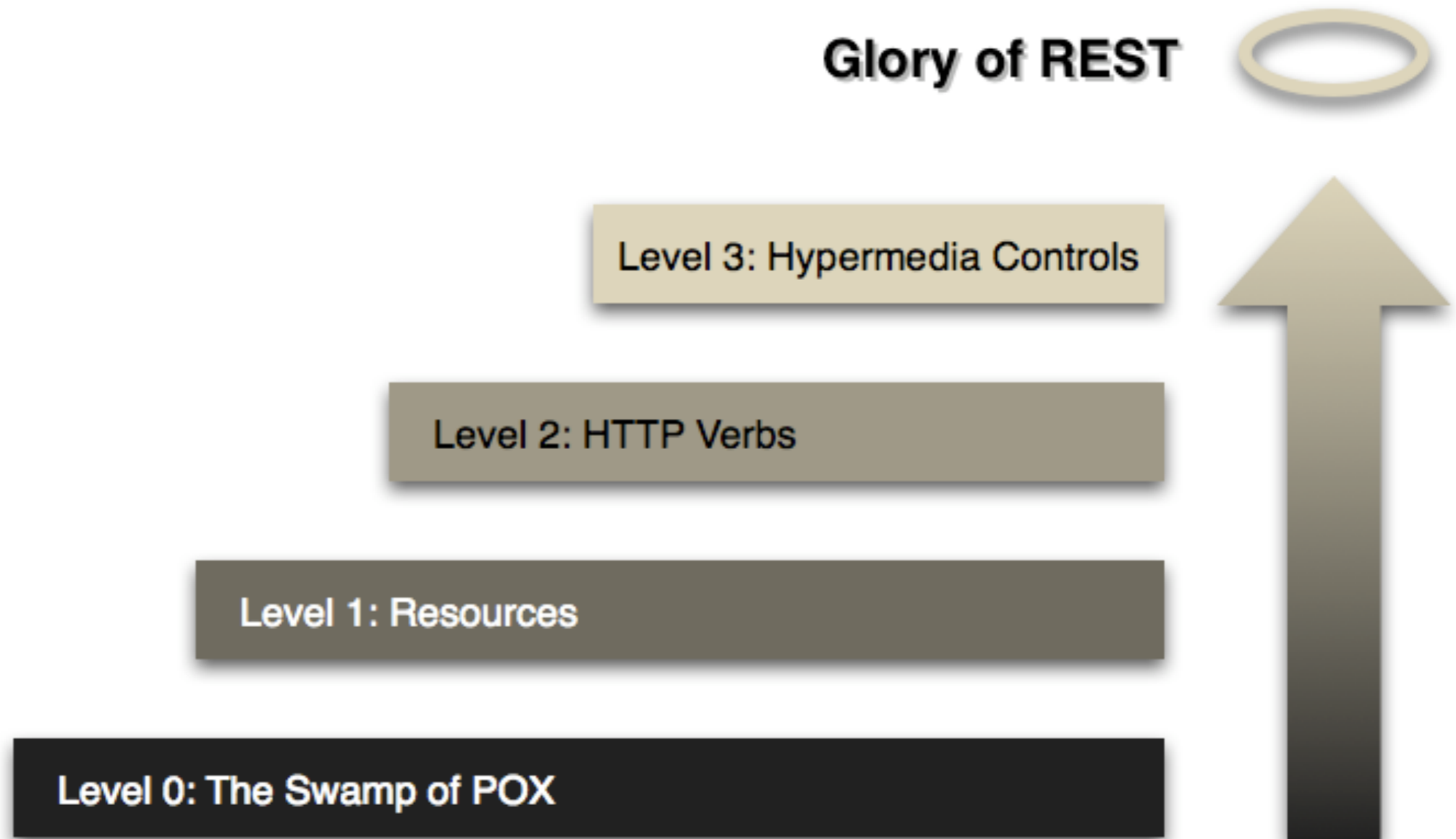


Hypermedia As The Engine Of Application State!



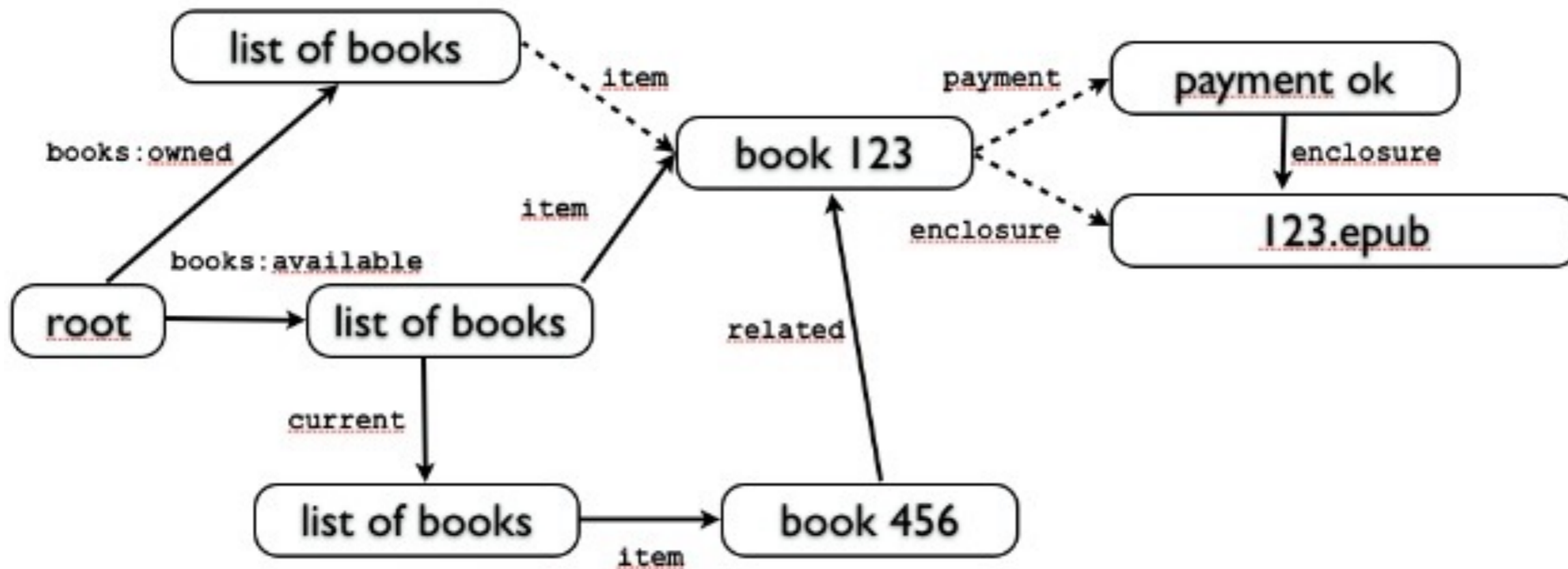
WHAT'S REST?

HATEOAS: 使用超媒体作为应用状态引擎



WHAT'S REST?

Hypermedia: 相互链接的资源





GitHub API漫游



WHAT'S REST?

REST约束/原则：正确使用HTTP

- 客户-服务器 (Client-Server)：通信只能由客户端单方面发起，表现为请求-响应的形式。
- 无状态 (Stateless)：通信的会话状态 (Session State) 应该全部由客户端负责维护。
- 可缓存 (Cacheable)：响应内容可以在通信链的某处被缓存，以改善网络效率。
- 统一接口 (Uniform Interface)：客户和服务端之间通过统一的接口相互通信，以提高交互的可见性。
- 分层系统 (Layered System)：通过限制组件的行为（即，每个组件只能“看到”与其交互的紧邻层），将架构分解为若干等级的层。
- 按需代码 (Code-On-Demand, 可选)：支持通过下载并执行一些代码（例如 Java Applet、Flash或JavaScript），对客户端的功能进行扩展。

HOW TO CONSUME RESOURCE?

ngResource

- { 'get': {method:'GET'},
- 'save': {method:'POST'},
- 'query': {method:'GET', isArray:true},
- 'update': {method:'PUT'},
- 'delete': {method:'DELETE'} };

```
var User = $resource('/users/:userId', {userId: '@id'});
User.get({userId:123}).$promise
  .then(function(user) {
    $scope.user = user;
  });
```

THANK YOU

Any Question or Feedback?

Jimmy Lv | 吕靖

jinglv@thoughtworks.com

<http://blog.jimmylv.info/2015/11/11/what-is-really-rest.html>

ThoughtWorks®