

Mobile Cloud Computing

初步研究汇报

陈矫彦

9月22号

研究内容：

1 Cloud 概述

2 Mobile Cloud Computing(MCC) 综述

3 MCC 实例

4 实验室现有 Mashup 研究成果

目录

- 文献列表
- Cloud 概述
- 移动计算的挑战与优势
- MCC 的定义与研究内容
- MCC 的综合实例
- 实验室 Mashup 现有的研究成果

文献列表

- 1 Mobile Cloud Computing: A Comparison of Application Models(综述)
- 2 Mobile Cloud Computing(硕士毕业论文)
- 3 The NIST Definition of Cloud Computing
- 4 Challenges of Mobile Computing
- 5 A Break in the Clouds: Towards a Cloud Definition
- 6 N-SMART: Networked Suite of Mobile Atmospheric Real-time Sensors(应用例子)
- 7 实验室现有的 Mashup 论文
- 8

Cloud 概述 (1)

- 定义

- 云是一个大的资源池，并且可以比较方便地访问和使用，同时这个资源池是基于硬件虚拟化来实现的；
- 资源可以根据用户的负载来动态配置，弹性比较好；
- 可以根据用户使用量计费

Cloud 概述 (2)

– 模型

- Cloud Software as a Service(SaaS)
 - 用户只能通过一个 thin 的客户端，比如浏览器，来访问软件资源，一个典型的应该就是在
线办公软件；
- Cloud Platform as a Service(PaaS)
 - 用户可以配置应用软件的环境，比如 Google App Engine ；
- Cloud Infrastructure as a Service(IaaS)
 - 用户可以配置操作系统、部署的软件甚至网络，构建一个根据用户需求的特定环境；

Cloud 概述 (3)

– 类型

- Private Cloud(私有云)
 - 组织内部使用，内部部署；
- Community Cloud(社区云)
 - 一些组织共享，内部部署，支持一些特定的社区；
- Public Cloud(公共云)
 - 向公众或者大的工业组织提供服务，由云销售服务商提供；
- Hybrid Cloud(混合云)
 - 同时具备上面 2 种或多种云的特性；

移动计算的挑战与优势 (1)

– 挑战

- 设备限制
 - 处理器速度、内存大小、显示器大小与分辨率、电力
- 网络限制
 - 带宽有限、高延迟、经常断开、带宽不稳定、网络异构性强；
- 移动性 (Mobility) 限制
 - 物理空间移动：导致网络连接中断等问题；
 - 信息空间移动：数据在 Internet 分散，移动导致访问不同的数据，引起 information traffic；
 - 连接空间移动：移动导致连接不同的网络 and 平台

移动计算的挑战与优势 (2)

– 现有移动应用性能分析

- 测试指标

- TCP throughput, Downlink/Uplink RTT, local DNS lookup time, TCP handshake time, Ping latency, page loading time, js execution time, browser concurrency

- 分析和结论

- Uplink throughput 和 RTT 不理想，这对将计算迁移到云端带来很大的挑战，我们首先要做的是**优化协议，优化传输**

移动计算的挑战与优势 (3)

– 优势

- Allow users to access any information using any device over any network at anytime
- 带有较多传感器（比如摄像头），在社会计算方面反而有优势，典型应用如：
 - GreenGPS：汽车移动端采集数据，进行数据挖掘，找出最省汽油的路径，通过手机客户端时行访问；
 - N-SMARTS：通过移动客户端数据，采集大气环境数据，进行实时预报和分析；

移动计算的挑战与优势 (4)

– 解决方案举例

- 数据压缩，权衡数据压缩的计算消耗与传输优化；
- Web service 优化，移动端和云端采用 JSON 和 Restfull WS，因为它们比 SOAP 的 WSDL 协议更加轻量级；
- Coda file system: 缓存文件系统，提供更好的交互，同时应对网络短暂断开等情况；

移动计算的挑战与优势 (5)

– 解决方案举例

- 云端与客户端同时数据缓存，
- 云端返回结果优化；

• MCC 的定义与研究内容 (1)

- MCC 可以通过连接云存储与计算资源，透明地、弹性地增加移动设备的能力
- 云端可以根据客户端的环境变化和计算上下文来动态地调整云端和客户端的计算比例
- 保留移动设备原有的传感和交互能力

• MCC 的定义与研究内容 (2)

- 基于 MCC 的移动应用：
 - 执行能力增加 (Augmented Execution)：克服计算、内存上的限制
 - 弹性的客户端与服务端执行分配：应对变化的异构环境
 - 应用可移动性：Internet 网上的机器从一端迁移到另一端，可以无缝的恢复执行
 - Ad-hoc 移动云：移动设备虚拟化提供云服务，比如用很多手机来构建 mapreduce 集群
 - 移动应用的模型：Middleware、Cost Model、Programming Abstraction、Solution Generality、Implementation Complexity、Static & Dynamic Adaptation、Network load、Scalability

Augmented Execution(增强执行)

- 移动应用现状

1. Offline Applications

应用程序本地执行，带来的问题是消耗本地资源，对平台可移植性差

2. Online Applications

应用程序在线执行（服务端执行），带来的问题是网络延迟导致实时性差，无法利用移动客户端的传感器设备，无法胜任一些复杂的环境

3. 静态分工

静态在客户端和服务端间划分计算任务，但是没有一种静态分工可以适用所有的异构环境，不同的负载也需要不同的分工

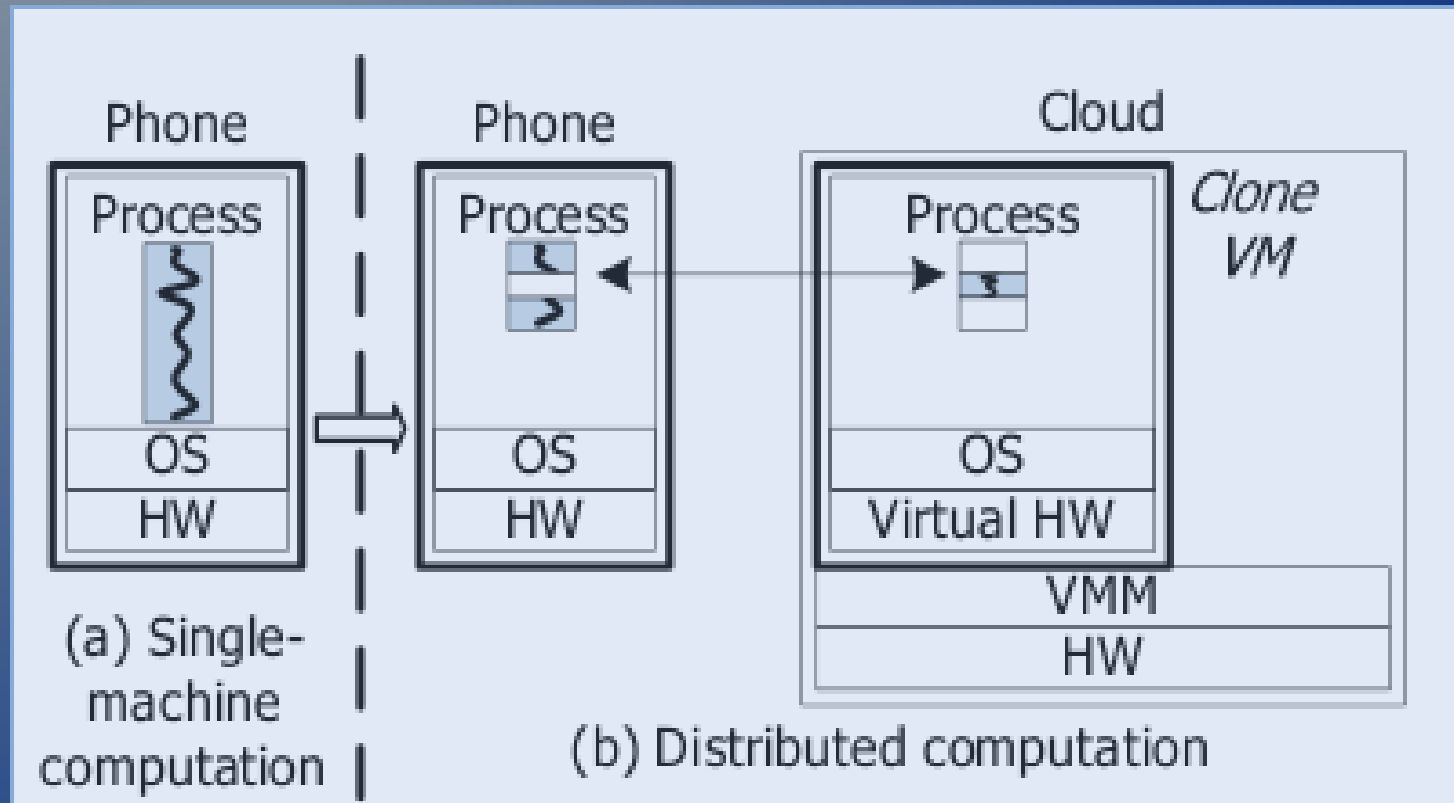
Augmented Execution(增强执行)

- 动态分工的必要性

1. 不同的设备平台有不同的资源，比如有的平台带有 GPU 处理器，那么可以胜任简单的图片处理工作；
 2. 不同的网络环境带来不同的 throughput，网络较差的时候，虽然云端较快的完成了计算，但是结果传输的延迟却带来更大的影响；
 3. 云端的环境，比如云端在逻辑上离得较远；
 4. 应用的负载，对不同的工作量要进行权衡；
- 。 。 。

Augmented Execution(增强执行)

- 系统模型比较



Augmented Execution(增强执行)

- 分类

1. Primary functionality outsourcing : 更像是现有的客户端服务端的应用模型 ;
2. Background augmentation : 一些独立的进程 , 可以完全卸载到云端执行 , 比如病毒扫描 ;
3. Mainline : 介于上面两者之间 ;
4. Hardware : 在云端虚拟一个同样的 more powerfull VM ;
5. Multiplicity : 将任务卸载到云端 , 并在云端并行执行 ;

Augmented Execution(增强执行)

- 分工优化模型

前提：

A = the entire set of modules

P_d = the partition for mobile device d

P_s = the partition for the server's in cloud

$P_d \cup P_s = A$ and $P_d \cap P_s = \{\}$

$$\begin{aligned} & \underset{P_d, P_s}{\text{minimize}} && \sum_{m \in P_d} C_p(m, d, i) + \sum_{m \in P_s} C_p(m, s, i) + \sum_{m1 \in P_d, m2 \in P_s} C_c(m1, m2, t, i) \\ & \text{subject to} && m \in P_d, \forall m \in L_1(d) \\ & && m \in P_s, \forall m \in L_1(s) \\ & && m1, m2 \in P_d \text{ or } m1, m2 \in P_s, \forall (m1, m2) \in L_2 \end{aligned}$$

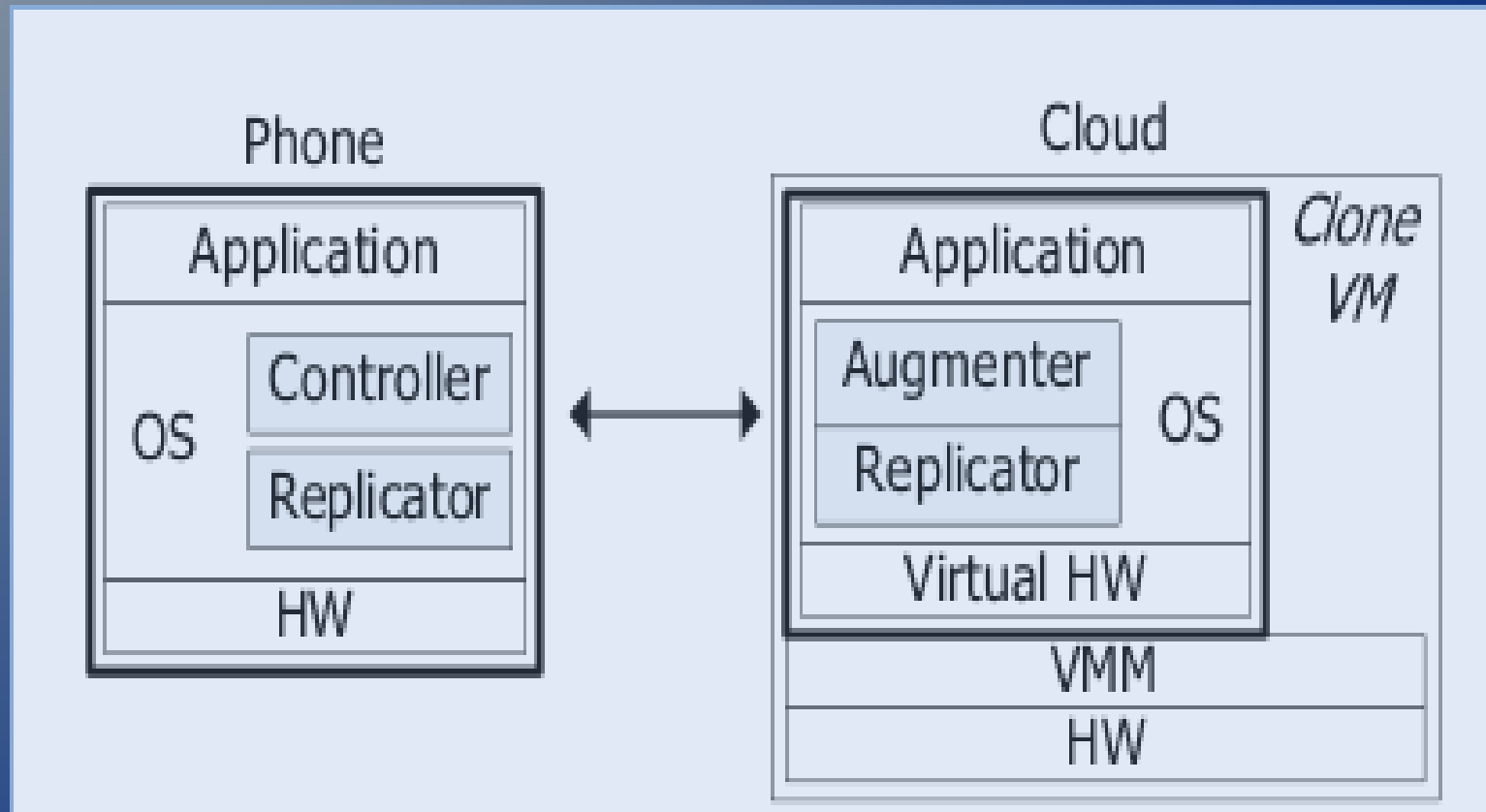
Augmented Execution(增强执行)

- 分工的系统支持

1. 客户端和服务端的应用程序和状态同步（应用迁移问题）；
2. 客户端和服务端的进行实例化，并且实现不同 modules 之间的进程通信（进程间通信或者 RPC）；
3. 分工方案选择：找出所有的分工方案，计算出分工方案的 cost，同时找到最好的分工方案。这个过程必须 light weight；
4. 安全和隐私：数据保密性和完整性，如何根据隐私保密的制定分工策略；

Augmented Execution(增强执行)

- 基于虚拟化云端的实现架构



VM-based Cloudlets

1. 基本思想：建设很多像加油站一个 cloudlet，移动客户端可以用 WLAN 连接上去，再通过 Augmented Execution 类似的形式增加移动客户端的能力；
2. 优势：使用离移动设备最近的云，解决了 3G 无线 (WAN) 带来的高延迟、低带宽问题，作者认为改进 WAN 基础设施在现在看来是不可行的；
3. 关键技术：根据连接的用户需要，cloudlet 瞬间的定制化，这要求实现快速的虚拟化迁移，动态虚拟化合成；
4. 部署实现上也有很大挑战，比如隐私和安全问题，对不同的手机的支持等。

弹性地划分应用

1. 内含：根据动态变化的异构环境，来动态的将应用划分成多个模块，然后在远程执行这些模块；

2. 现有工作：

- 分布式模块管理：自动、动态地什么时候、哪些模块卸载到云端；
- 应用程序划分的弹性化探索
- R-OSGi：分布式 Java 应用程序执行模型
- MAUI：有效地将移动端到代码迁移到云端，对减少耗电有很大作用
- 通过执行过程动态配置，实现应用程序的划分

应用程序的移动性研究

1. 内含：在程序执行过程中，在主机之间迁移应用，类似于进程迁移，但比进程迁移做得更多，它涉及更多个进程迁移；
2. 现有工作：
 - ISR：虚拟机网络状态迁移，相当于虚拟的动态迁移（在执行过程中迁移），缺点是迁移整个虚拟机消耗更多的时间和带宽；
 - 基于 java 平台的自适应移动迁移，整个解决方案支持单个应用程序的迁移，并且支持自适应性；

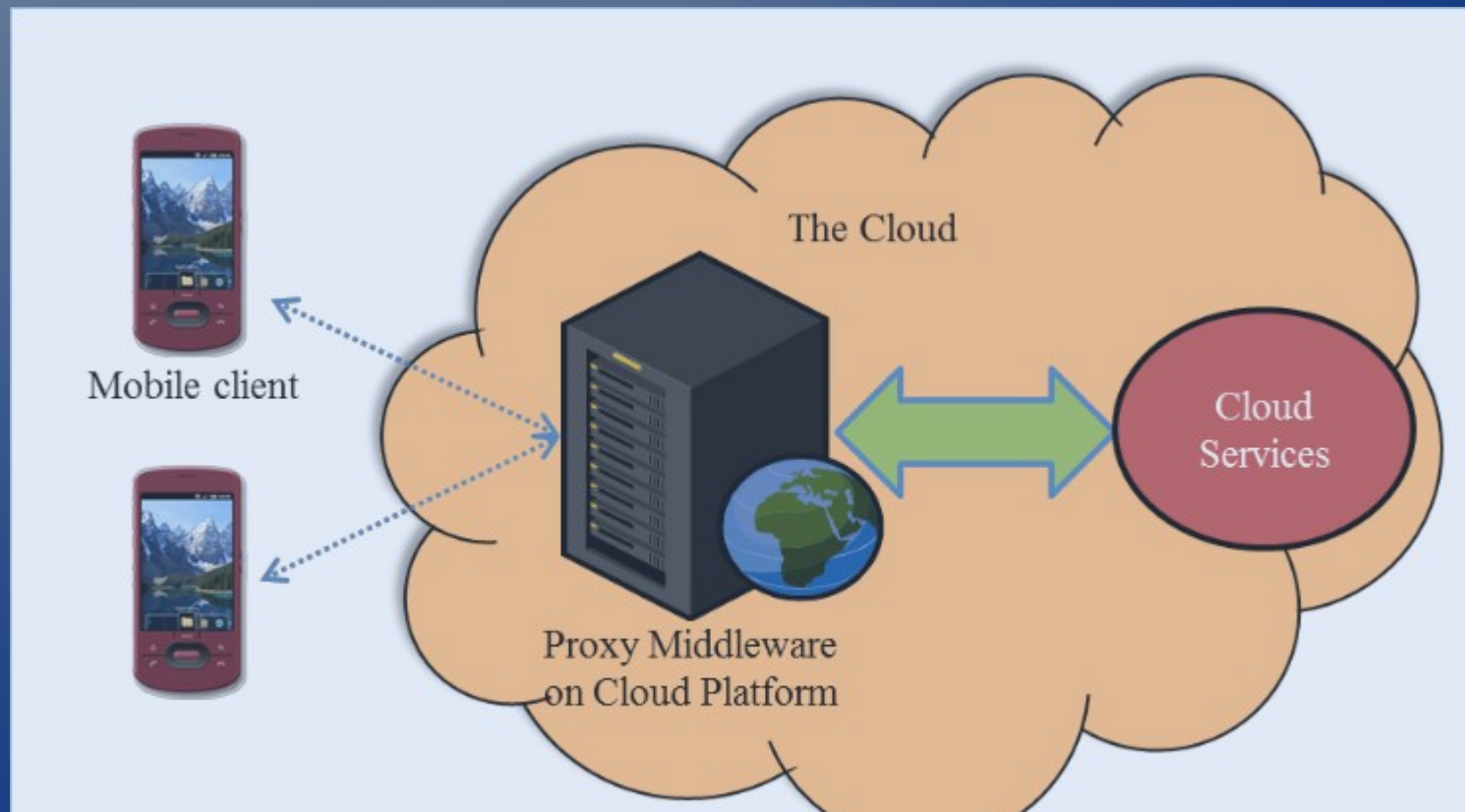
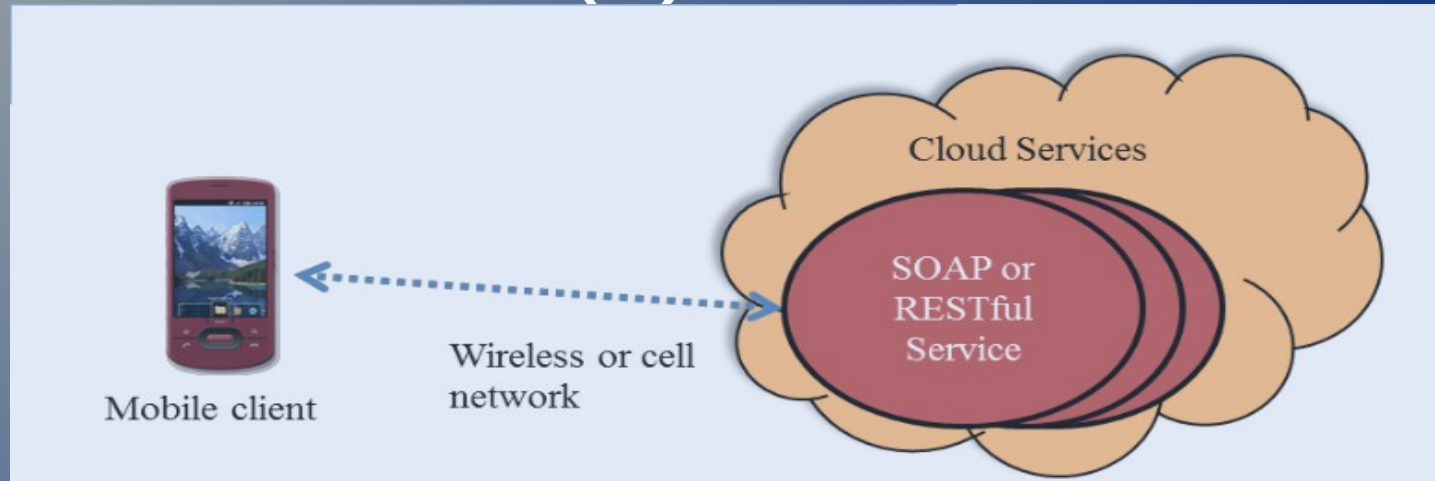
移动设备提供的云

1. 含义：一组移动设备虚拟成云，向其它移动设备提供资源
2. 现有工作：
 - XMPP：扩展消息协议，用于实现移动设备间的通信，以完成移动设备云虚拟化
 - Hyrax：实现了手机上的 mapreduce

移动云计算探索点

1. 编程抽象化：移动云计算的开发应该更加简单和直观，这要求我们为移动云计算提供平台，而不是一个或者几个应用！
2. Cost 模型：对一些执行参数，比如执行时间、资源消耗、电池用量、安全、网络带宽有一个测量和计算的模型，也就是要制定标准！
3. 自适应性：它是实现可移动性的关键，当前两个极端，一个是完全由程序来实现自适应性，另一个是程序对自适应性透明，由系统来实现，事实上应该是做一个权衡。
4. 云端集成：包括数据传输优化、数据一致性、持久性、缓存机制、云端监控等一系列云端问题
5. 信任、安全、隐私是永远的话题！

- Mobile Cloud Computing 的综合实例
(1) – 架构



- Mobile Cloud Computing 的综合实例
(2) – 特点

- 问题 1 : 连接丢失
 - 客户端和云端双缓存、云端推送
- 问题 2 : 带宽和延迟限制
 - 协议转化 : SOAP Ws 转为 RESTful Ws 或者 JSON、结果优化
- 问题 3 : 有限的资源
 - 云端计算、个人 Mashup 平台

- 实验室 Mashup 现有的研究成果
- 实现了一个 Mashup 平台：
 - 包括基于语义的推荐机制和执行引擎
- 发表了若干论文：
 - MMAI: Mashup-based Mobile App Inventor
 - Towards our Real Life-SMMS: Semantic-based Mobile Mashup System
 - Voronoi Diagram-based Navigation for Web Mashups (TBD)
 - Mashup by Surfing a Web of Data APIs