



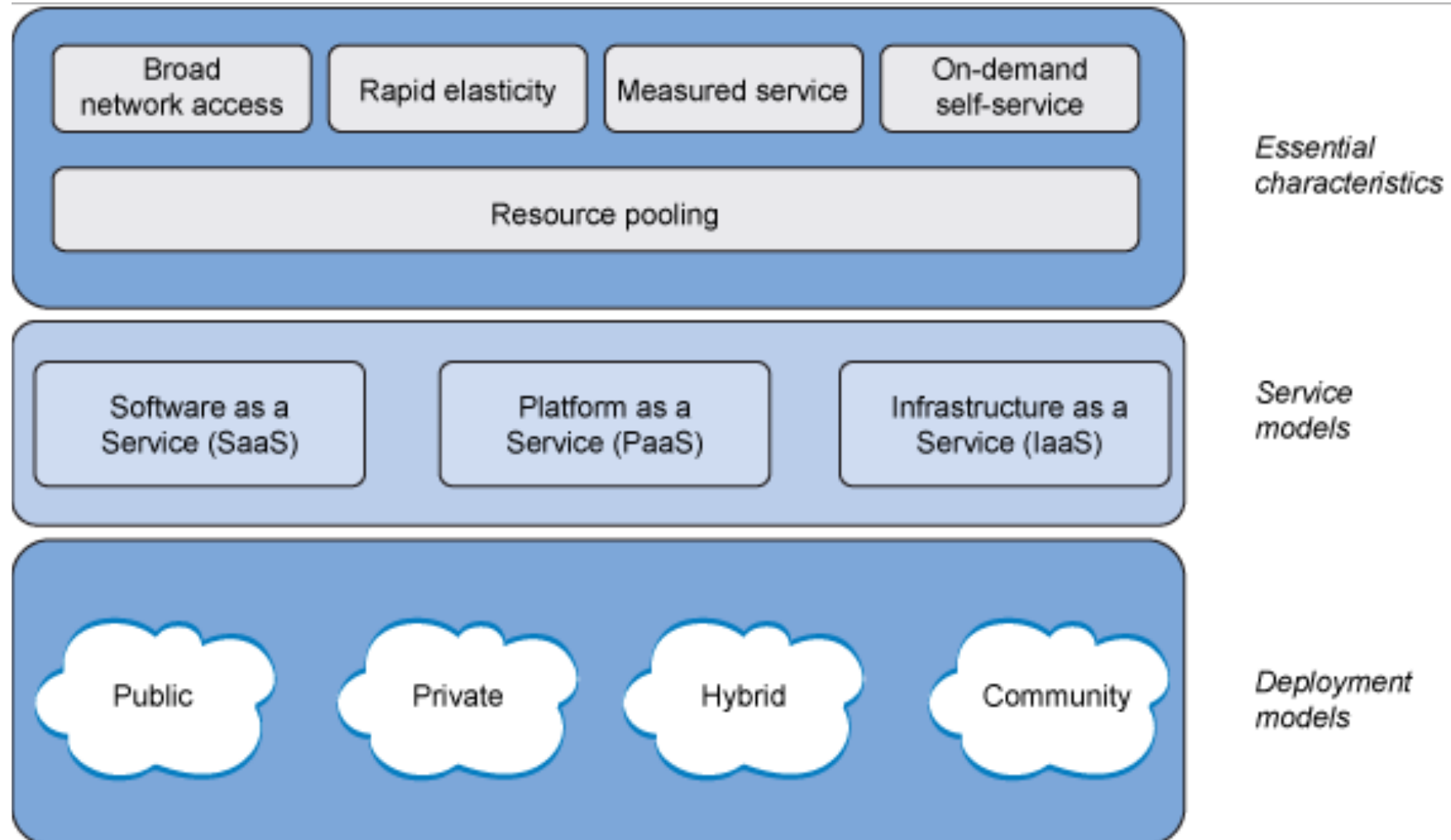
Clearing the Fog: Understanding z Systems Cloud Technology Options

Glenn Anderson
IBM Lab Services and Training



NewEra z Exchange
March, 2016

Cloud computing is a model for service delivery



What do people mean when they say “cloud computing?”



- **Cloud computing is a model for service delivery**
 - **Where is the service coming from?** (public, private or hybrid)
 - **What kind of service is it?** (IaaS, PaaS, SaaS)
 - **Which characteristics of the cloud service delivery model are important to you?** (what is the problem you are trying to solve?)
 - On demand self-service
 - Broad network access
 - Resource pooling
 - Rapid elasticity
 - Measured service



Mainframe and public cloud



- What does it mean?

- You are choosing to move some/all of your current mainframe workload to a public cloud service provider. It could also mean new workloads that could have been run on the mainframe will be run by a public cloud service provider instead.

- Why would you do it?

- Someone thinks it will save money
- Move your workloads off of your own z Systems mainframe to one owned by a public cloud service provider
- Someone thinks the mainframe is dead

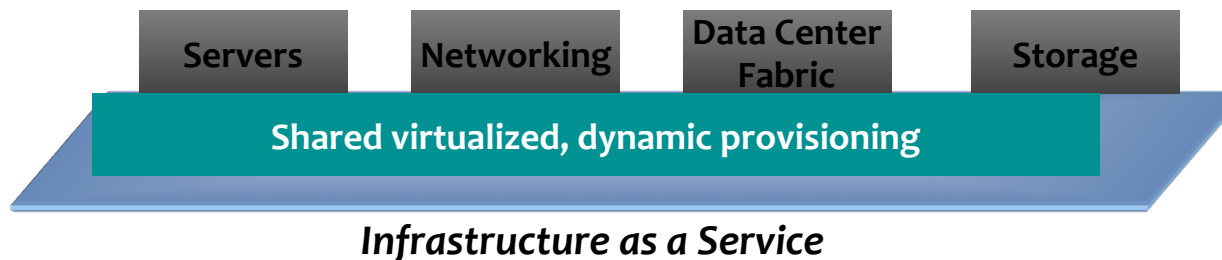
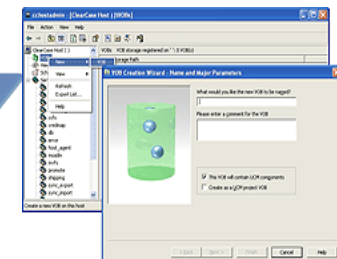
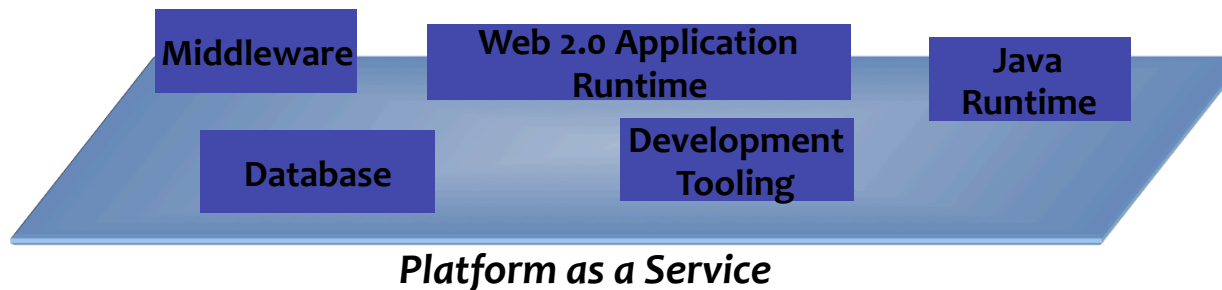
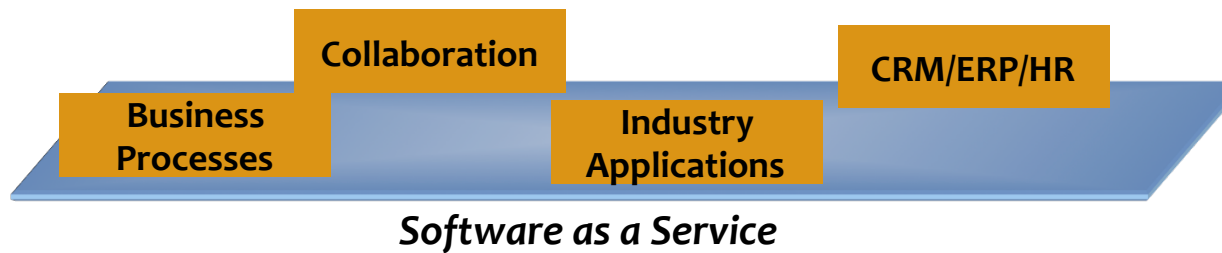


Mainframe and private cloud

- What does it mean?
 - You want to use the mainframe to deliver service to your enterprise using the characteristics of the public cloud service delivery model. These services could be IaaS, PaaS, SaaS, etc.....
- Why would you do it?
 - To enhance your service delivery to be like public cloud, while taking advantage of the strengths of the mainframe
 - To save money through server consolidation
 - Because someone tells you to implement cloud in your current data center



What type of service are you delivering?



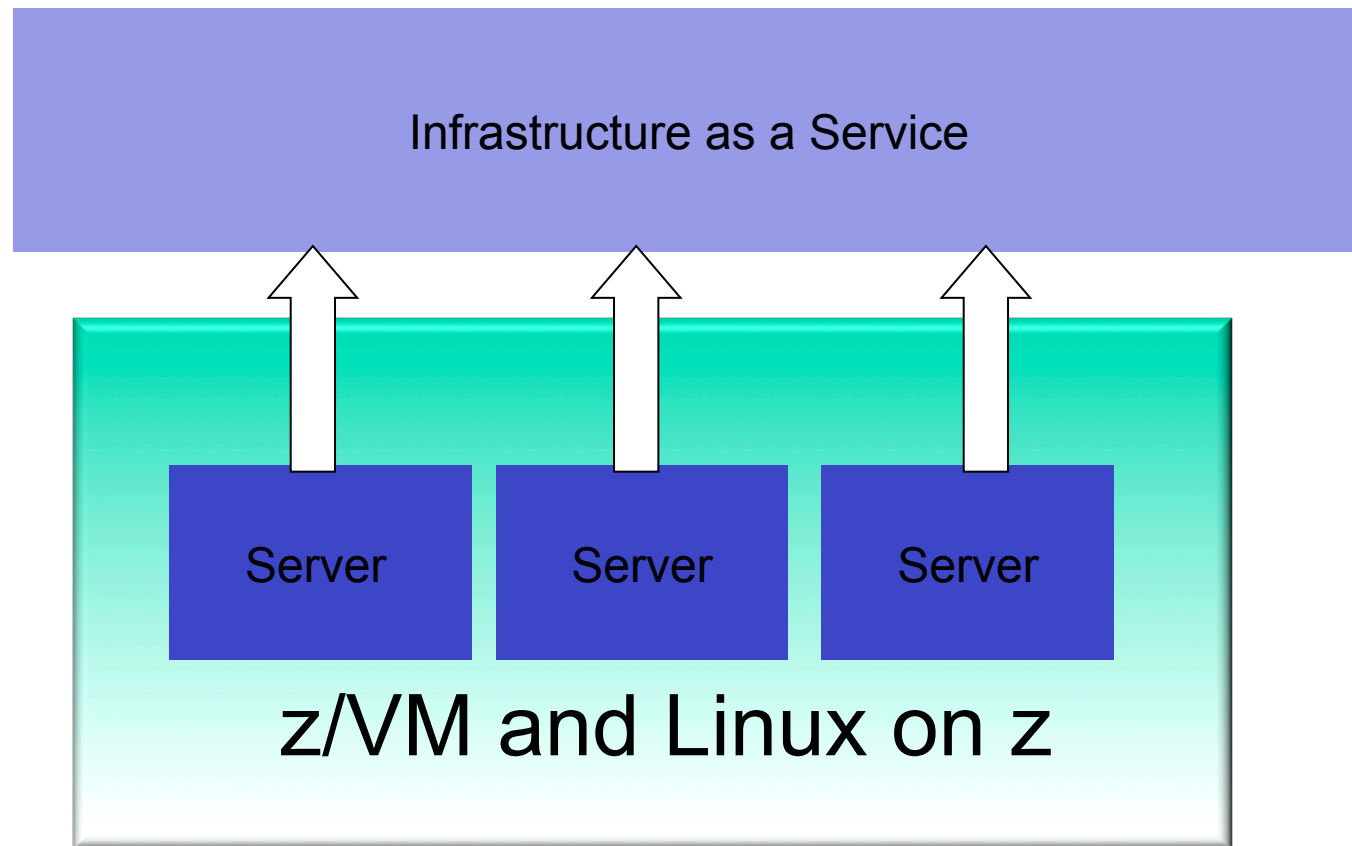
Which of these Characteristics of Public Cloud Computing Interests You as a z Systems IT Organization?



- **On-Demand Self Service**
 - Pick services you need, when you need them
- **Broad Network Access**
 - Available over network through thin or thick clients
- **Resource Pooling**
 - Resources are shared, serving multiple consumers
- **Rapid Elasticity**
 - Capabilities provisioned, in some cases automatically
- **Measured Service**
 - Pay only for what you use



IaaS with z/VM and Linux on z Systems



Cloud Portfolio for Linux on z Systems



Virtualization

Infrastructure &
Virtualization Management

IBM z/VM 6.3

- Support more virtual servers than any other platform in a single footprint
- Integrated OpenStack support



IBM Wave for z/VM

- A graphical interface tool that simplifies the management and administration of z/VM and Linux environments



KVM for z Systems

Differentiation

Entry Level Cloud

Standardization & Automation



IBM Cloud Manager with OpenStack

- A simple, entry level cloud management stack
- Based on OpenStack
- Can be run from System z environment ("managed-from")
- Formerly known as SmartCloud Entry

Standardization

Advanced Cloud

Orchestration & Optimization

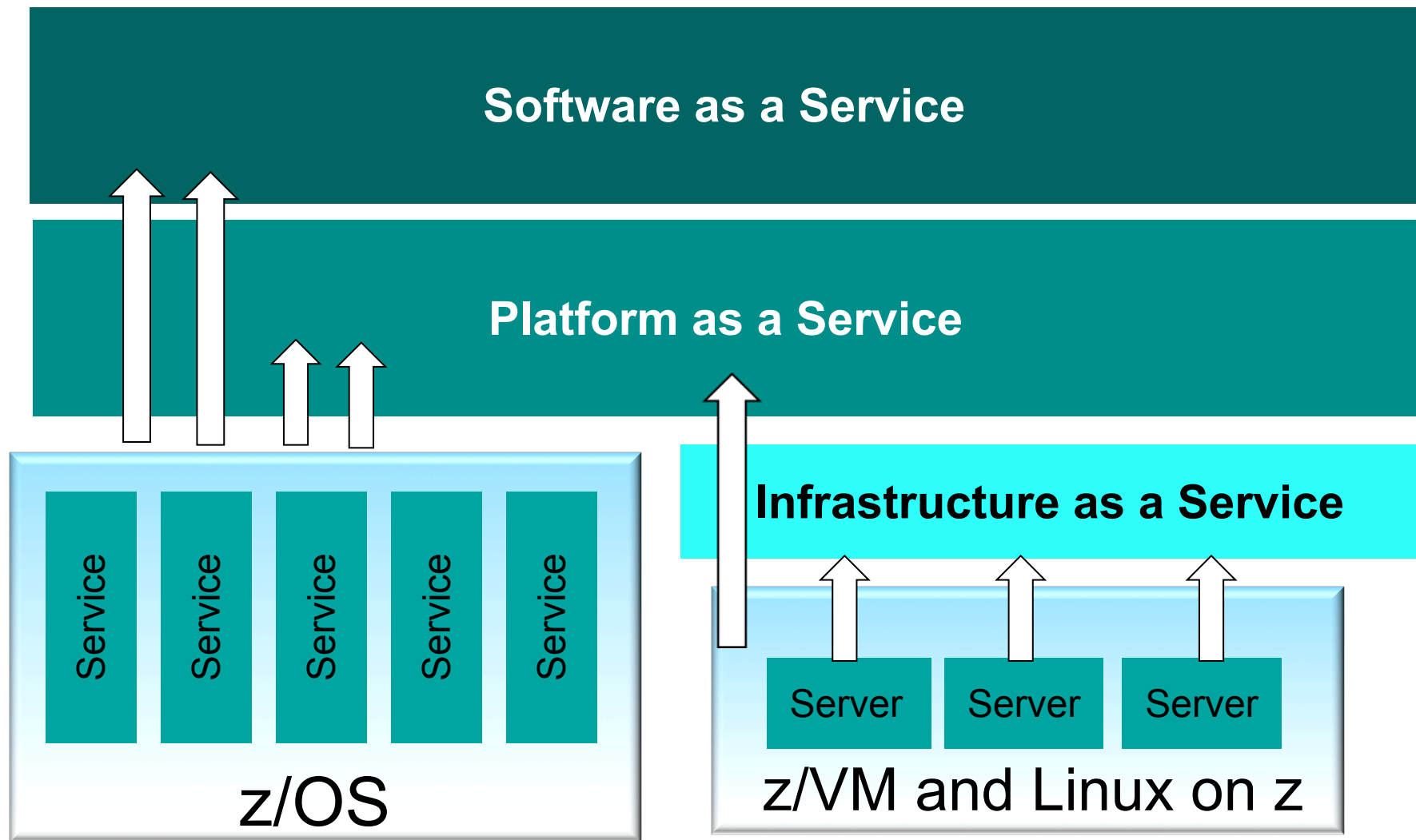


IBM Cloud Orchestrator

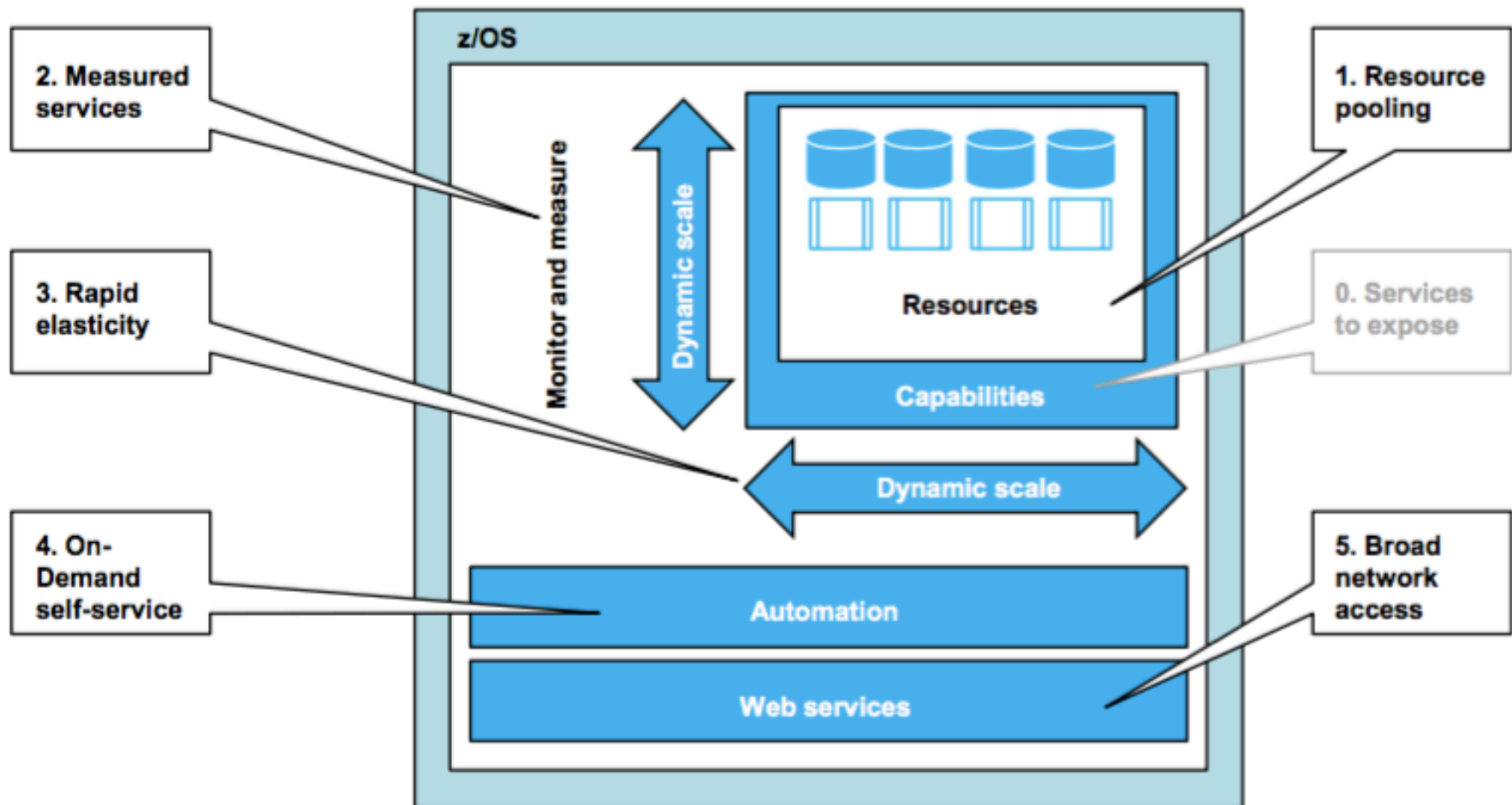
- Builds on functionality of **IBM Cloud Manager with OpenStack** and adds runbook automation and middleware pattern support for workload deployment
- Based on OpenStack
- System z support as "managed-to"
- Formerly known as SmartCloud Orchestrator

*Service Lifecycle
Management*

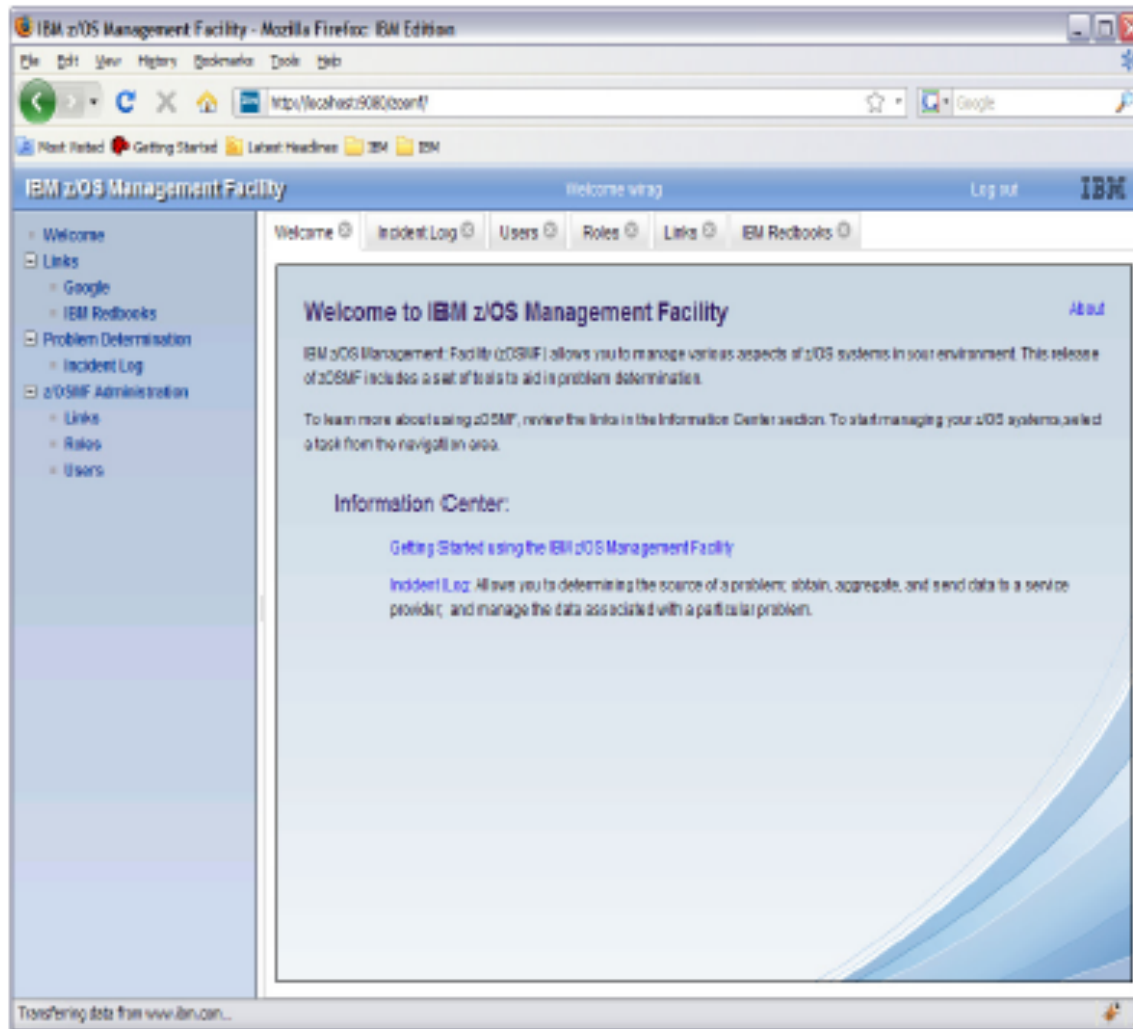
PaaS and SaaS with z/OS



z/OS and cloud characteristics



z/OS Management Facility (z/OSMF)

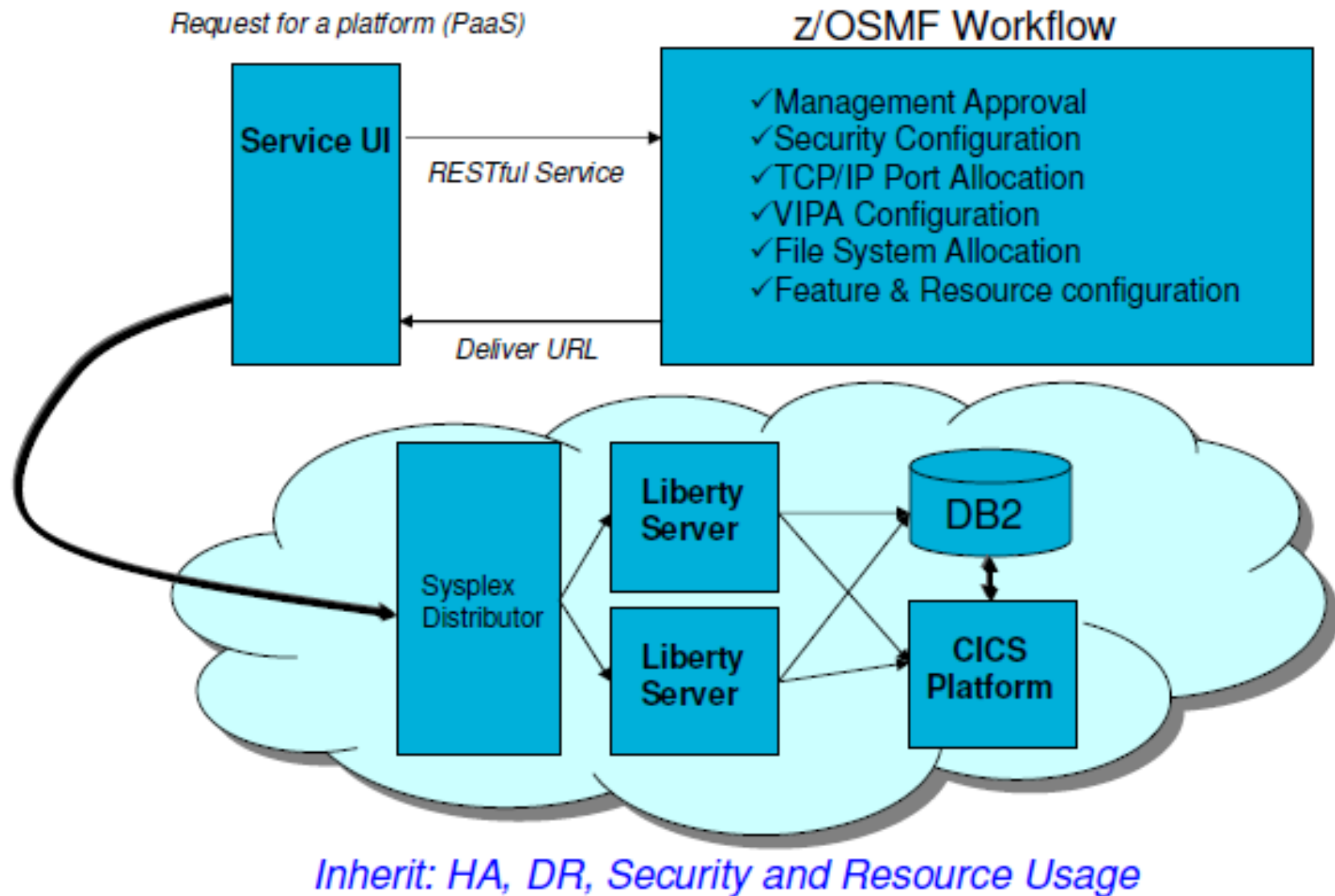


The new face of z/OS

Simplify z/OS

- Software management
- Capacity provisioning
- Workload management
- Network Configuration
- Workflow Automation**

Provisioning a Liberty Server (PaaS) on z/OS with z/OSMF



APIs: the building blocks for apps

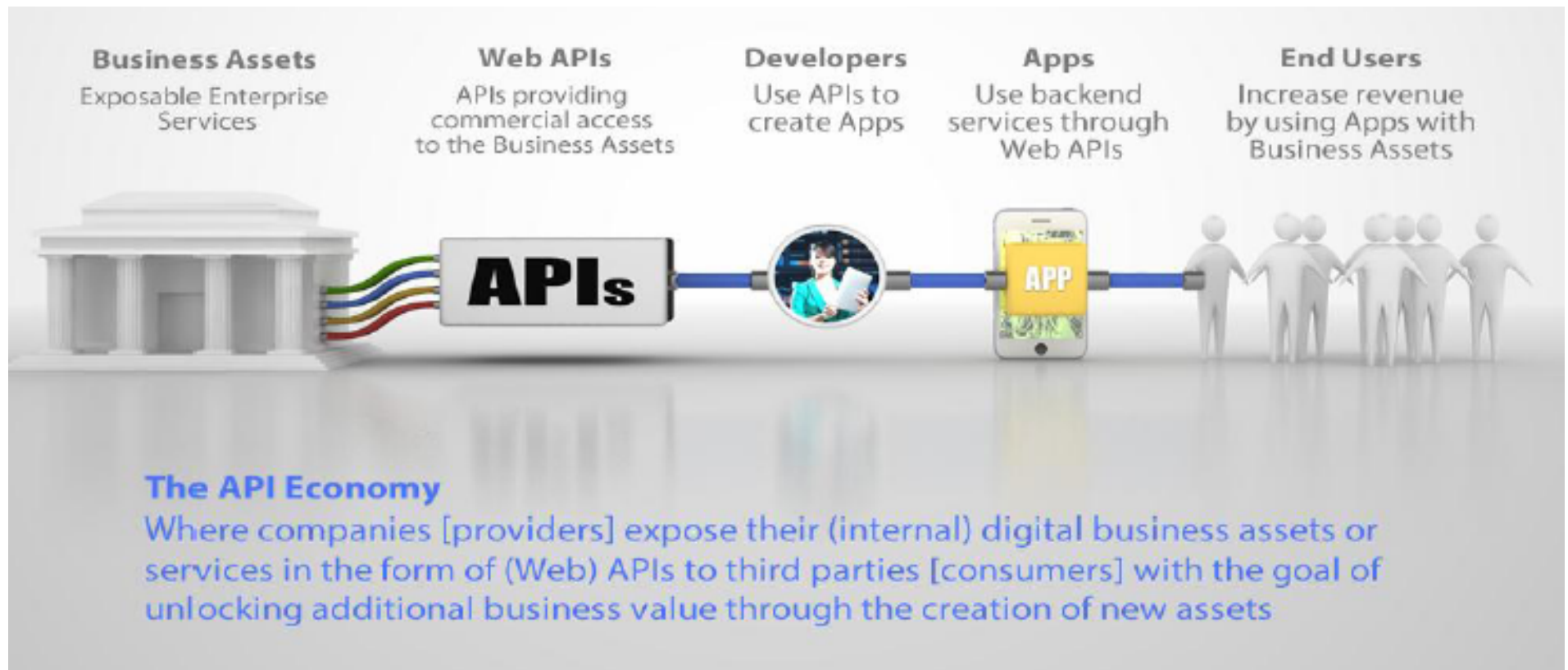


The “**API economy**” has changed how developers think about building apps, and how organizations deploy software in the cloud

Bank Externalized Services



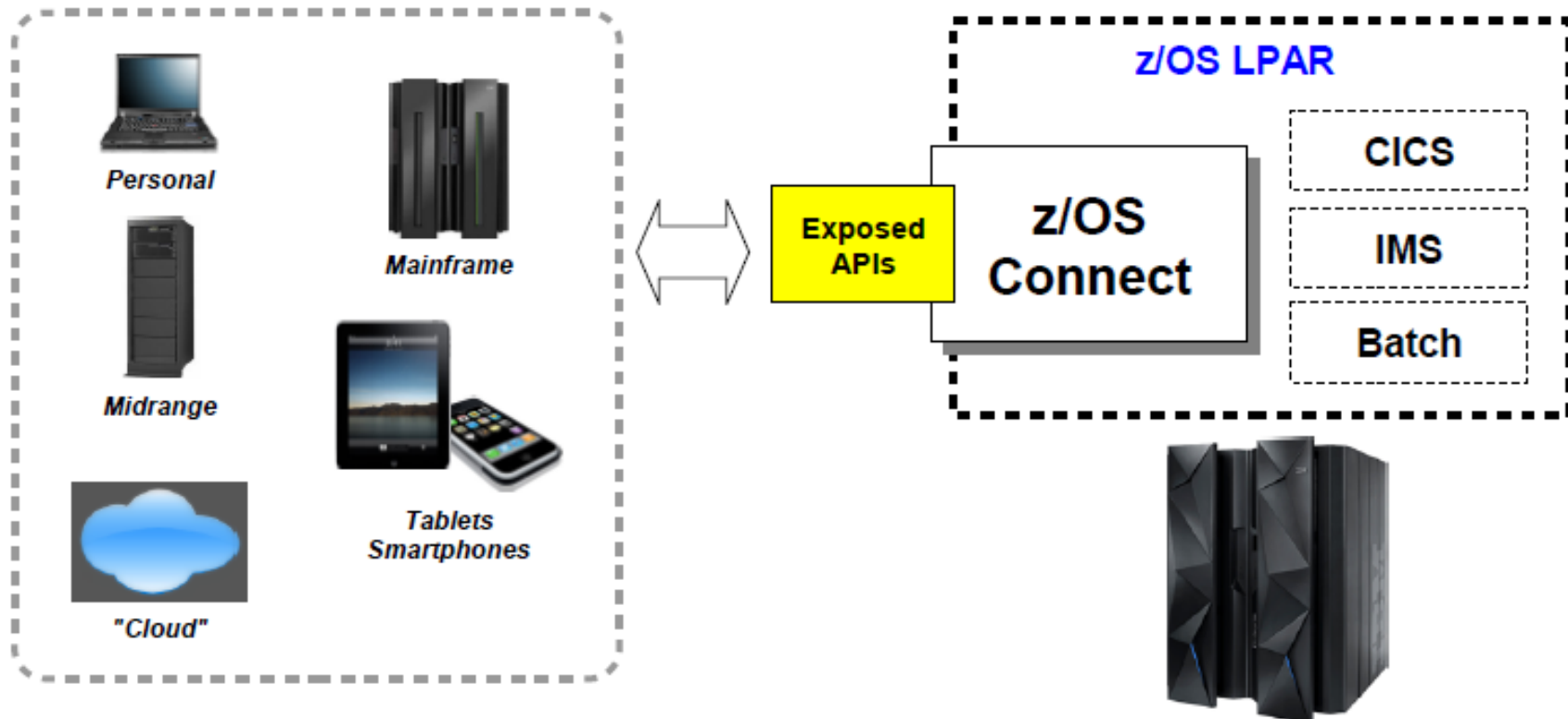
API economy lifecycle



API economy: mainframe as a service

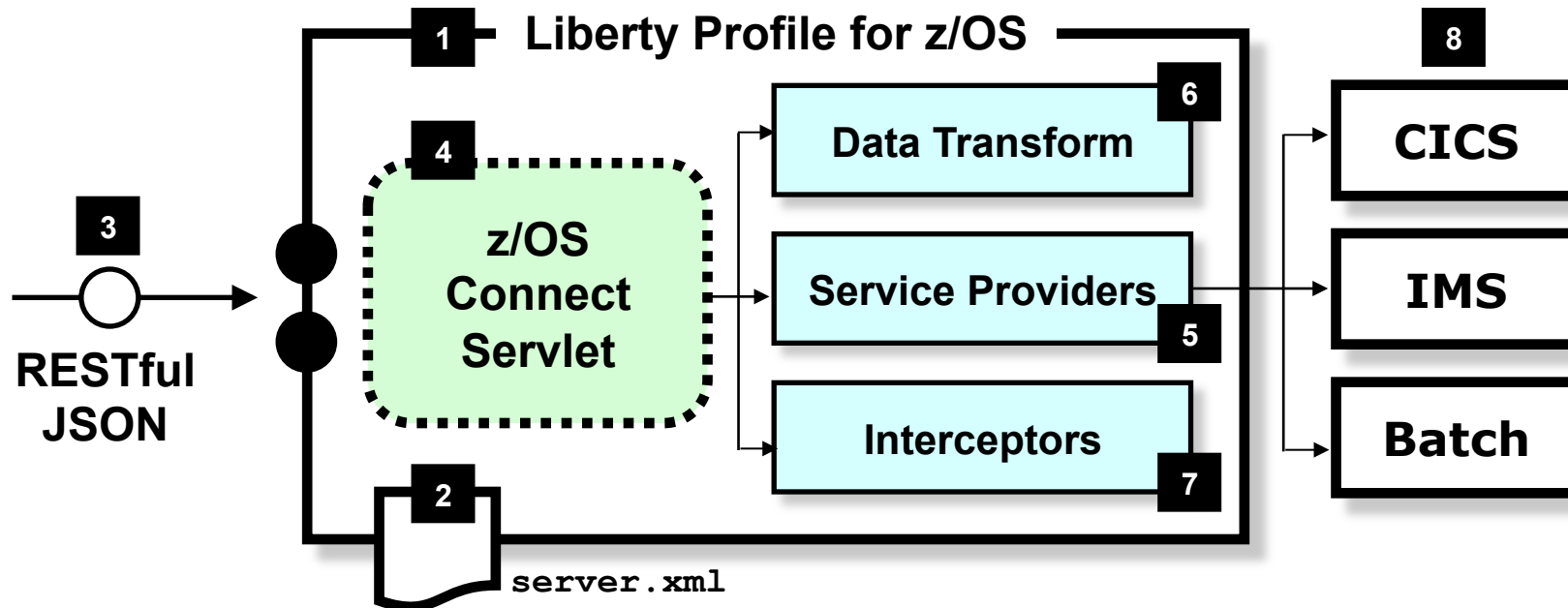


Another use-case for z/OS Connect is as a standard gateway into the z/OS LPAR to expose programs as a service:



**z/OS Connect provides a way to do this
with a single entry point (HA is possible)
and common protocol (REST/JSON)**

What is z/OS Connect?



1 z/OS Connect is software function that runs in Liberty Profile for z/OS.

2 z/OS Connect is described and configured in the Liberty `server.xml` file

3 z/OS Connect is designed to accept RESTful URIs with JSON data payloads

4 One part of z/OS Connect is a servlet that runs in Liberty Profile z/OS.

5 A 'Service Provider' is software that provides the connectivity to the backend system

6 z/OS Connect provides the ability to transform JSON to the layout required by backend

7 'Interceptors' are callout points where software can be invoked to do things such as SAF authorization and SMF activity recording

8 Initially the backend systems supported will be CICS, IMS and Batch

RESTful Services

Stands for Representational State Transfer ... this is a protocol built on HTTP, using HTTP verbs (GET, PUT, POST, DELETE), where the URI indicates the service requested:



URI = Uniform resource identifier

`https://mysite.com/CustomerApp/getCustomer?cn=1234`

JSON – JavaScript Object Notation

It is a way of passing data back and forth as a series of name/value pairs.



URI = Uniform resource identifier

`https://mysite.com/CustomerApp/update?cn=1234`

```
{
  "firstName": "John",
  "lastName": "Smith",
  "age": 25,
  "address": {
    "streetAddress": "1234 Main Street",
    "city": "Anytown",
    "state": "NY",
    "postalCode": "10021-1234"
  },
}
```

The data being passed in is appended to the URI and passed in to the server

JSON can be passed back to the client as well.

Overview of z/OS Connect EE V2.0

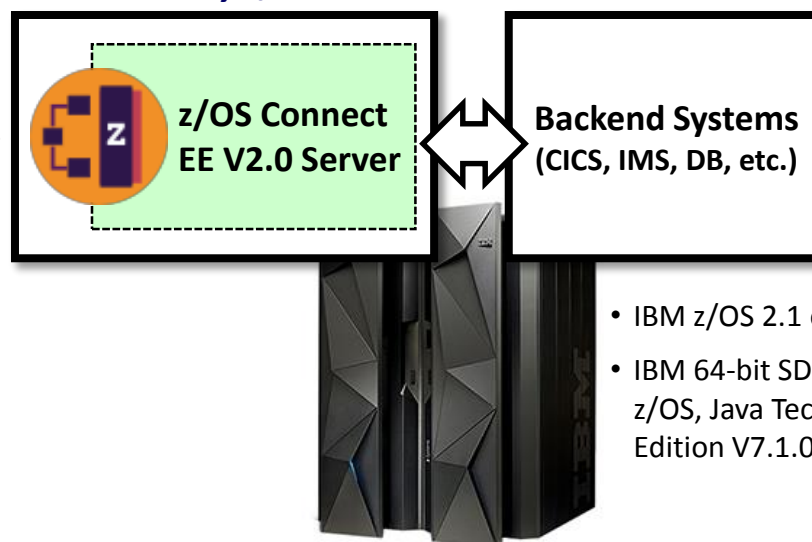


Runtime Server

1

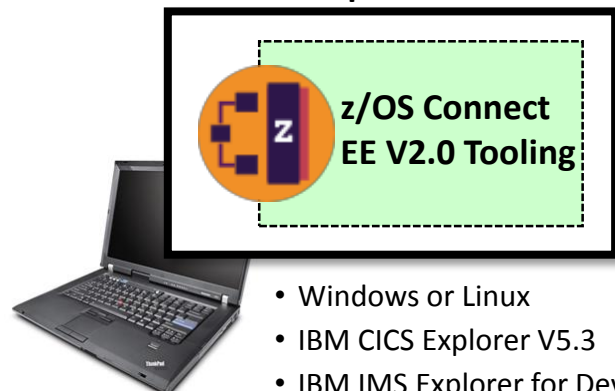
- Runs on Liberty z/OS
- Hosts APIs you define to run in it
- Connects with backend system
- Liberty + z/OS Connect = “instance”
- You may have multiple instances

Liberty z/OS



- IBM z/OS 2.1 or higher
- IBM 64-bit SDK for z/OS, Java Technology Edition V7.1.0 or V8.0.0

Eclipse



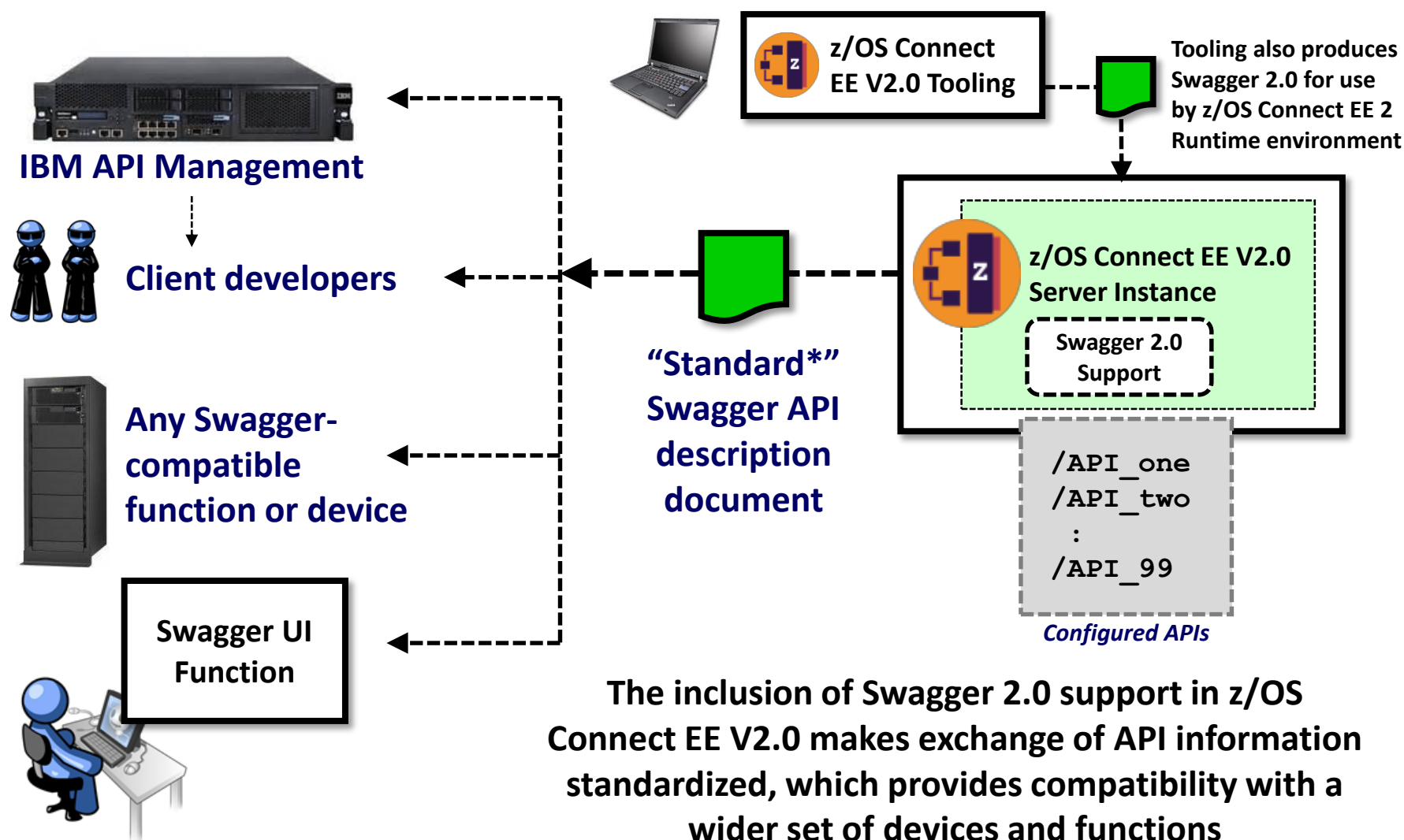
- Windows or Linux
- IBM CICS Explorer V5.3
- IBM IMS Explorer for Development V3.2
- IBM Explorer for z/OS Aqua V3.0

Tooling Platform

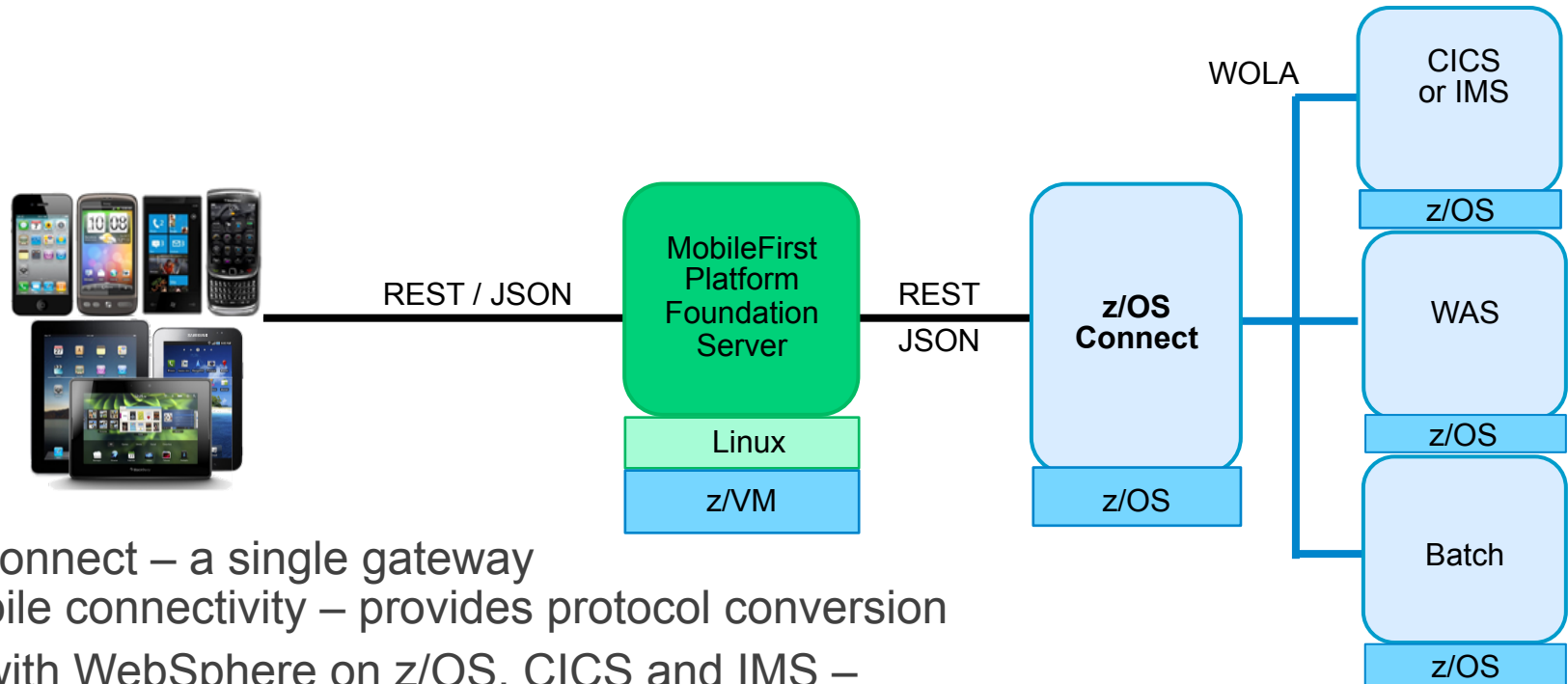
2

- Integrates with an Eclipse environment
- Define APIs
- Define data mapping
- Deploy APIs to runtime server
- Export API archive for other tools to deploy

Discoverable APIs with Swagger 2.0

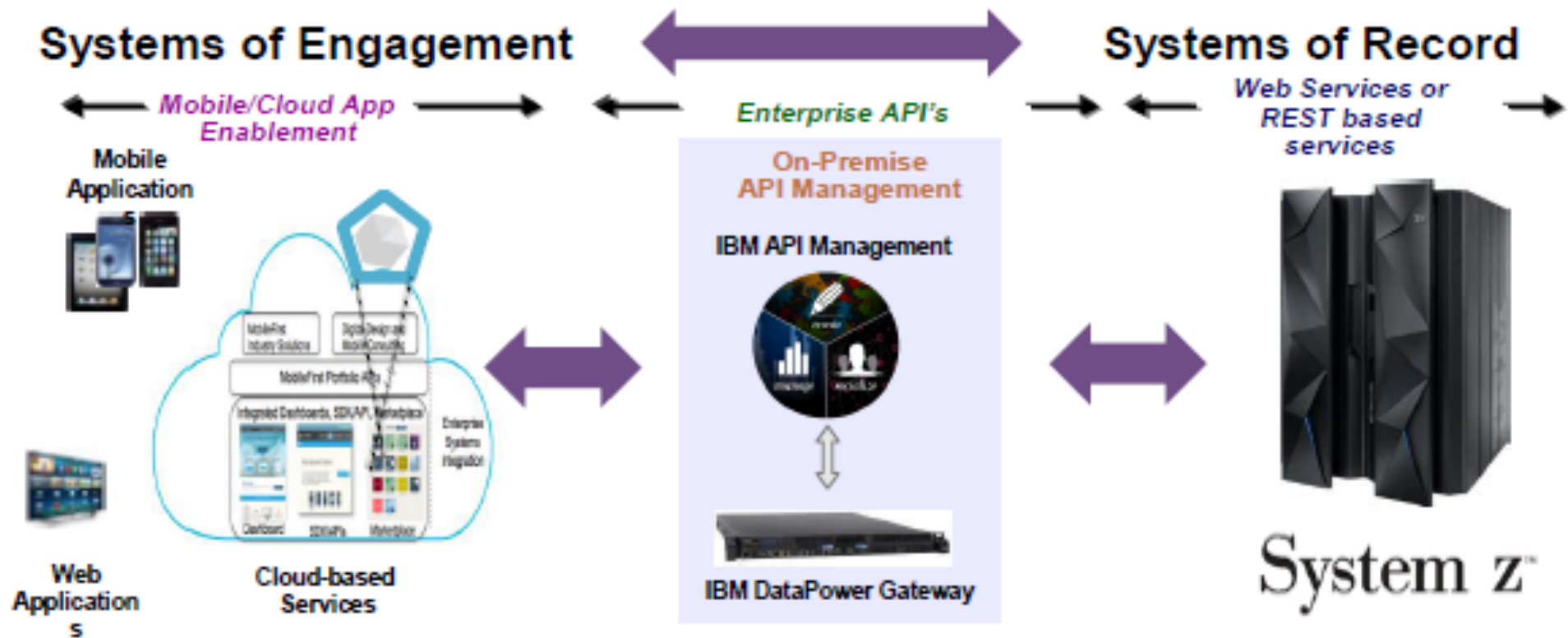


MobileFirst and z/OS Connect



- z/OS Connect – a single gateway for mobile connectivity – provides protocol conversion
- Ships with WebSphere on z/OS, CICS and IMS – at no additional charge
- Integrated into z/OS services (e.g., WLM, SMF, etc.)

IBM API Management



Get started with API Management



You can design, publish, and manage APIs through the API Manager console.



[Import APIs, or compose a new one.](#)



[Create a new plan, add resources and rate limits, and deploy.](#)



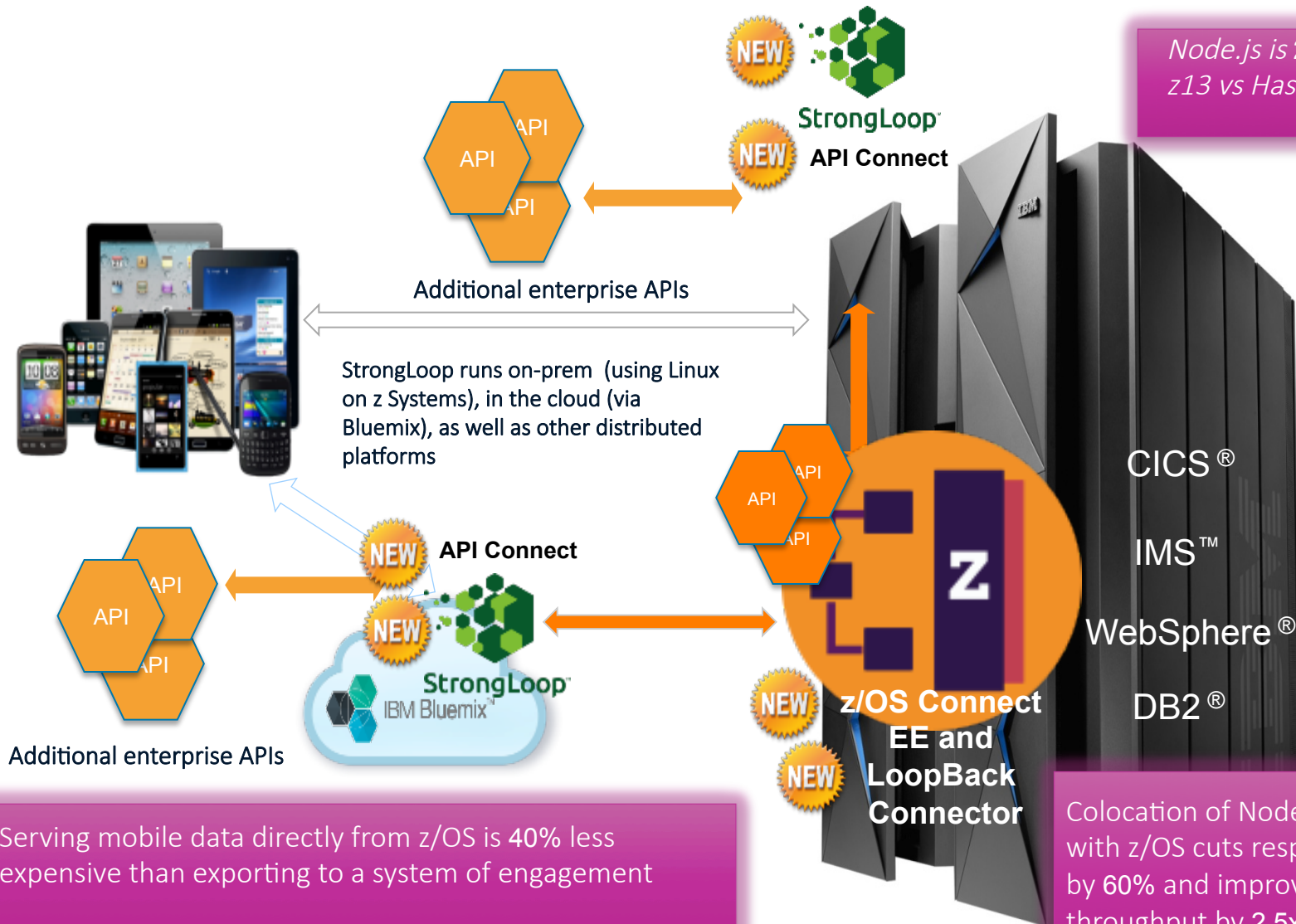
[Invite other Bluemix organizations to use your APIs.](#)



[Publish your plan.](#)

Application developers will be able to discover and consume your APIs from the Bluemix catalog.

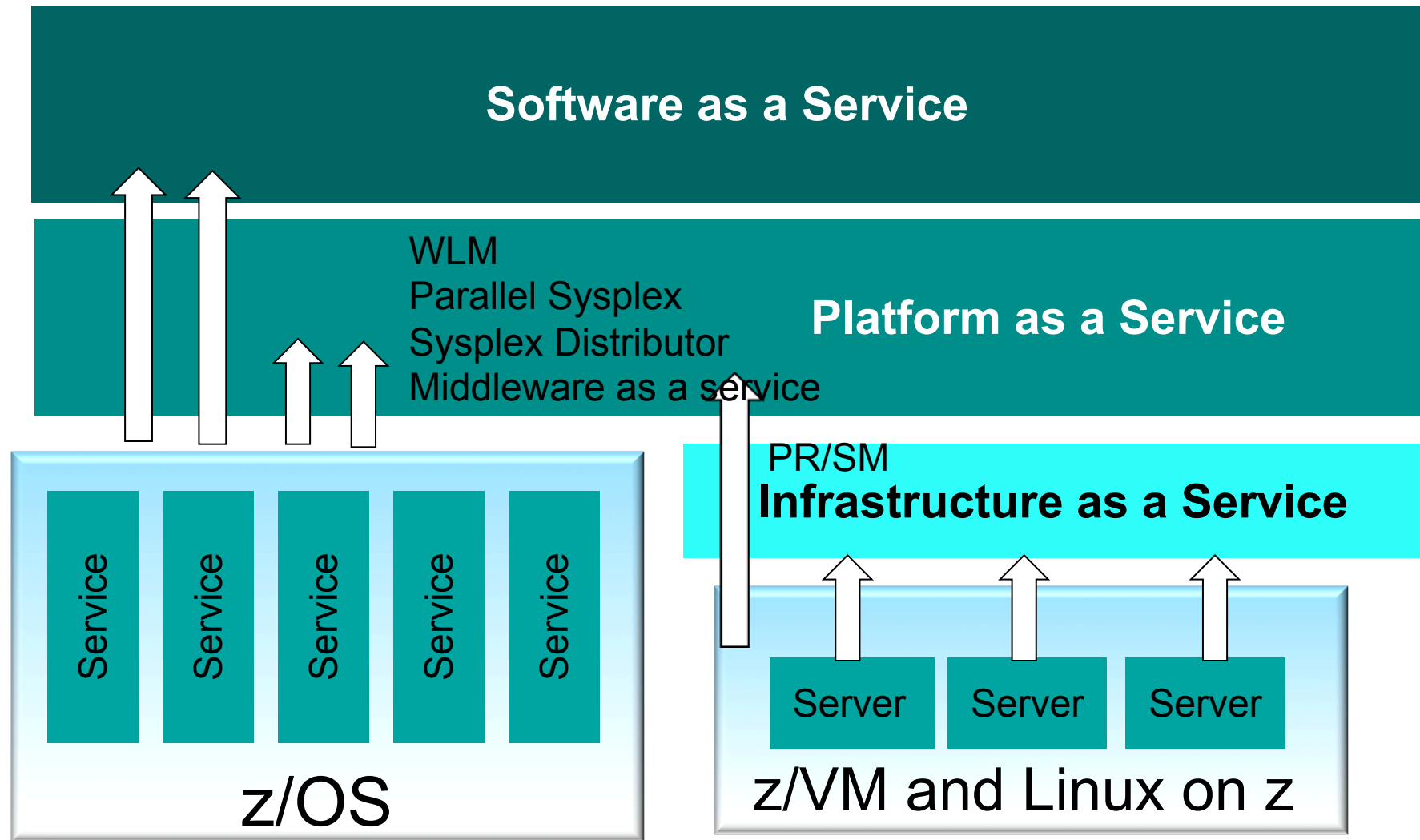
z Systems and the API Economy



The z/OS client web enablement toolkit provides a set of application programming interfaces (APIs) to enable traditional, native z/OS programs to participate in modern web services applications.

- Pieces of the toolkit:
 - A z/OS HTTP/HTTPS protocol enabler to externalize HTTP and HTTPS client functions in an easy-to-use generic fashion for user's in almost any z/OS environment
 - A z/OS JSON parser which parses JSON coming from any source, builds new JSON text, or adds to existing JSON text.
- The toolkit allows its two parts to be used independently or combined together.
 - Payload processing is separate from communication processing.
- The interfaces are intuitive for people familiar with other HTTP enabling APIs or other parsers
- Easy for newbies

Cloud Services provided by z Systems



Mainframe and hybrid cloud



- What does it mean?

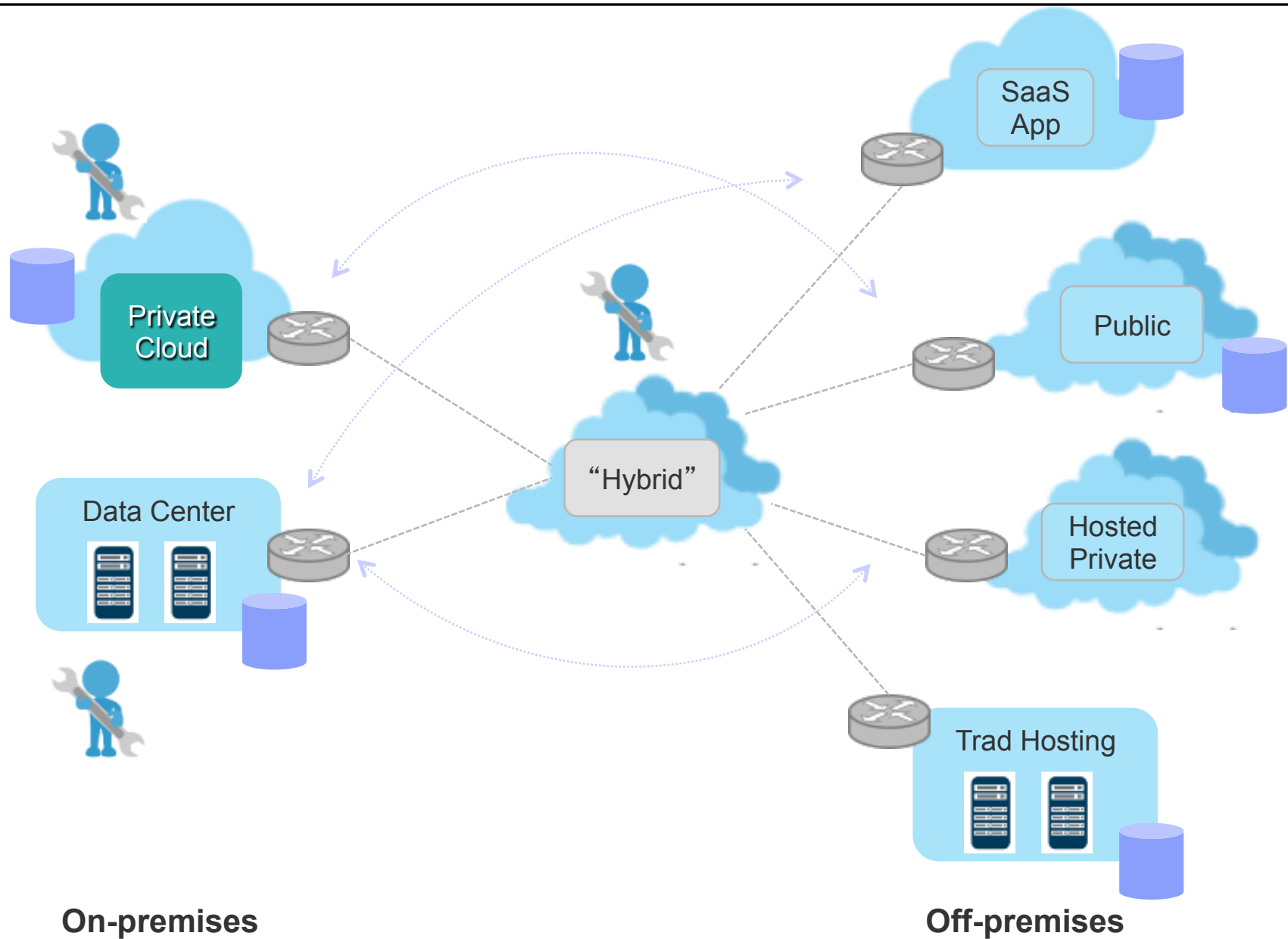
- You want to combine the strengths of the mainframe with the benefits of the public cloud service delivery model

- Why would you do it?

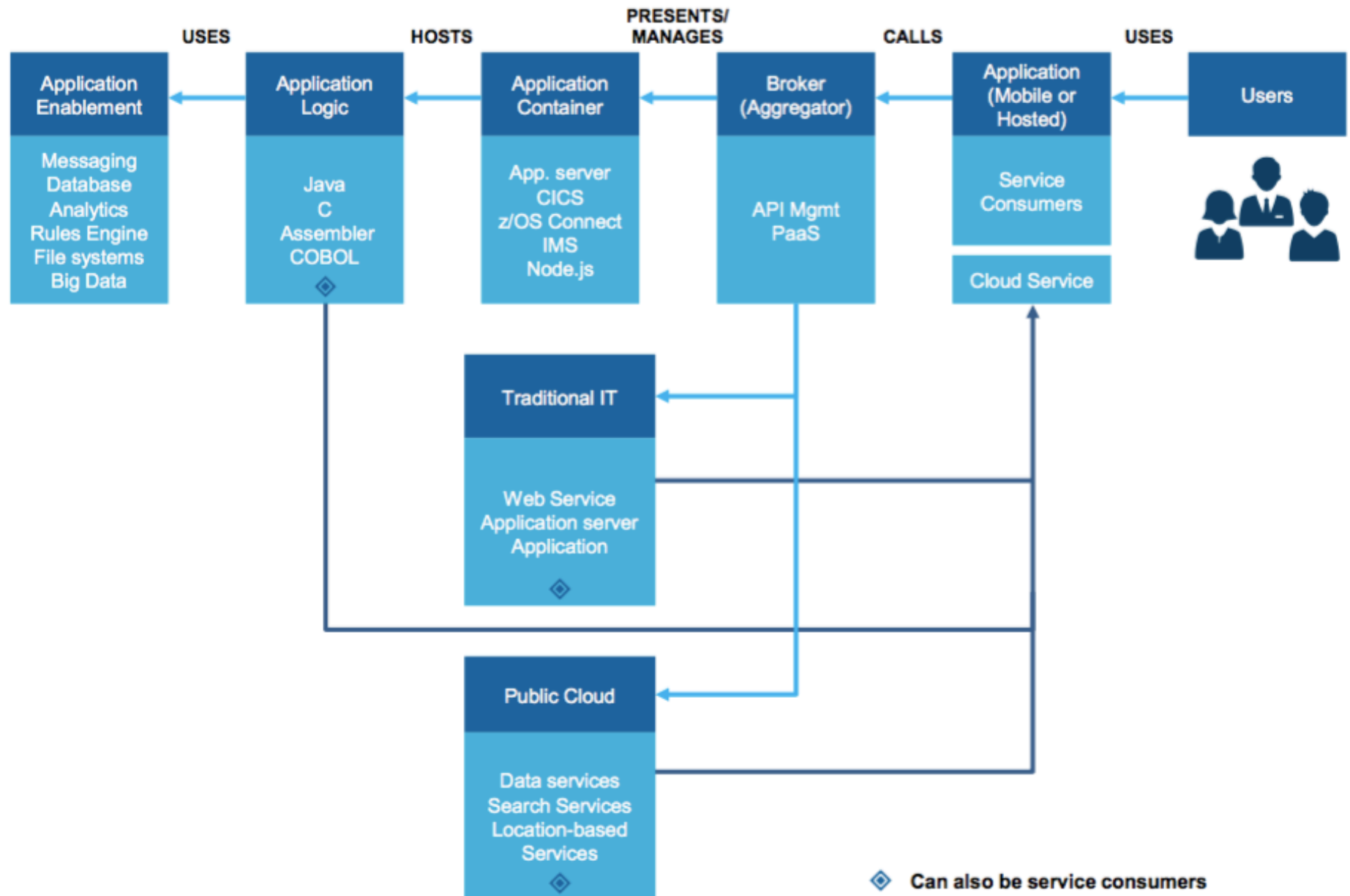
- Offers a mix of on and off premises – the best of both worlds
 - To use public cloud services in a “Fit for Purpose” model that follows the “Systems of Engagement” and “Systems of Record” architecture
 - Because it sounds like a cool thing to do
 - As a compromise between an all public cloud strategy and your current mainframe-based environment



The hybrid cloud reality



Hybrid cloud service delivery model



Hybrid cloud example: Adding a loyalty program



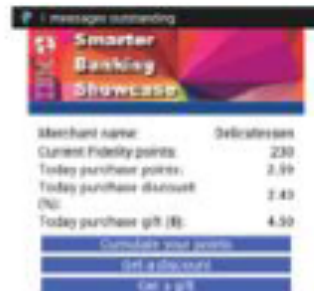
M. Majorana buys some Luxury food in a Shop



The Bank, located in Montpellier, receives the payment request and realizes a relationship between the Lady and the Merchant : A loyalty program !

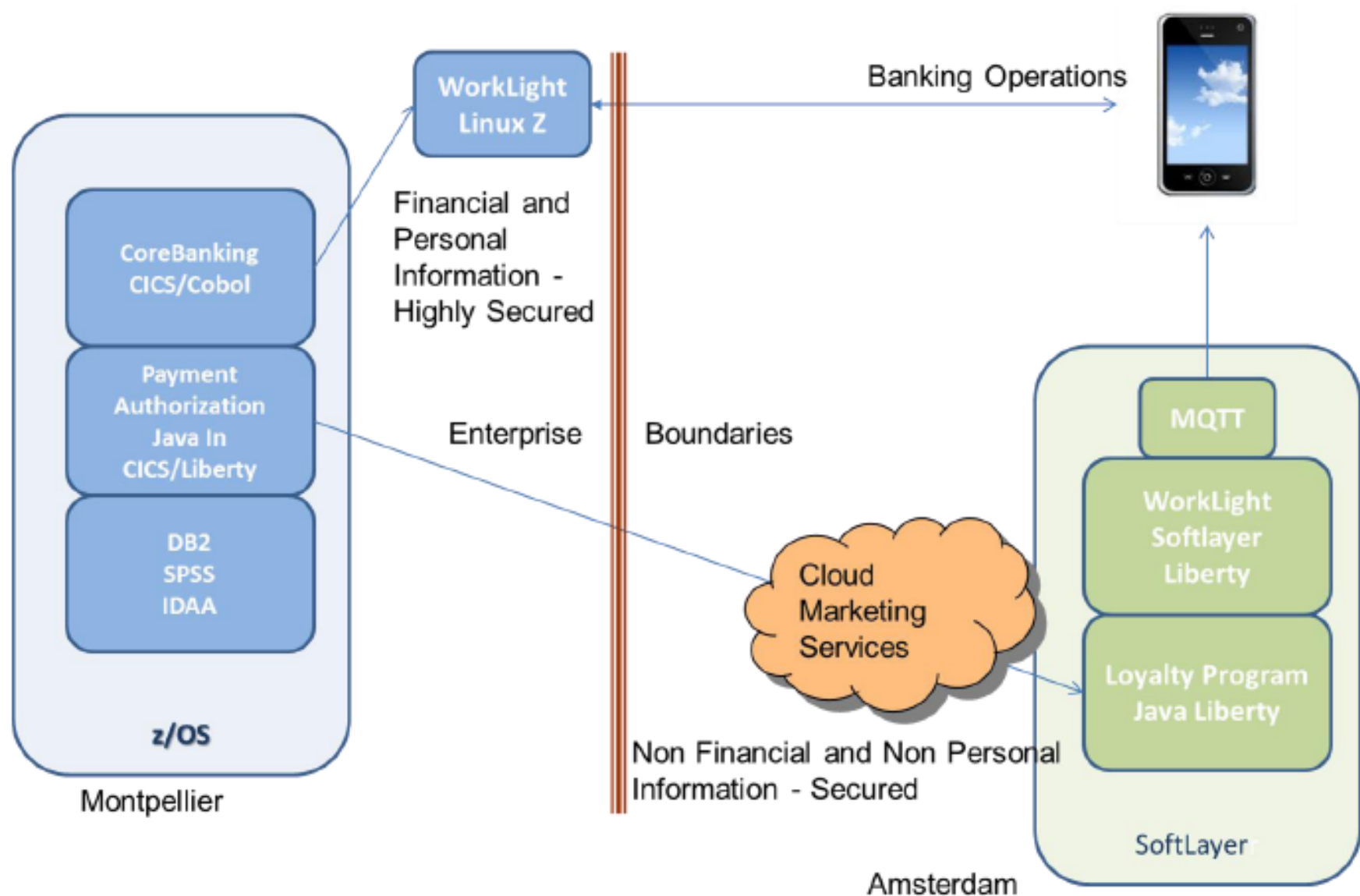


While continuing processing the payment the Bank ask a service deployed on a SoftLayer server

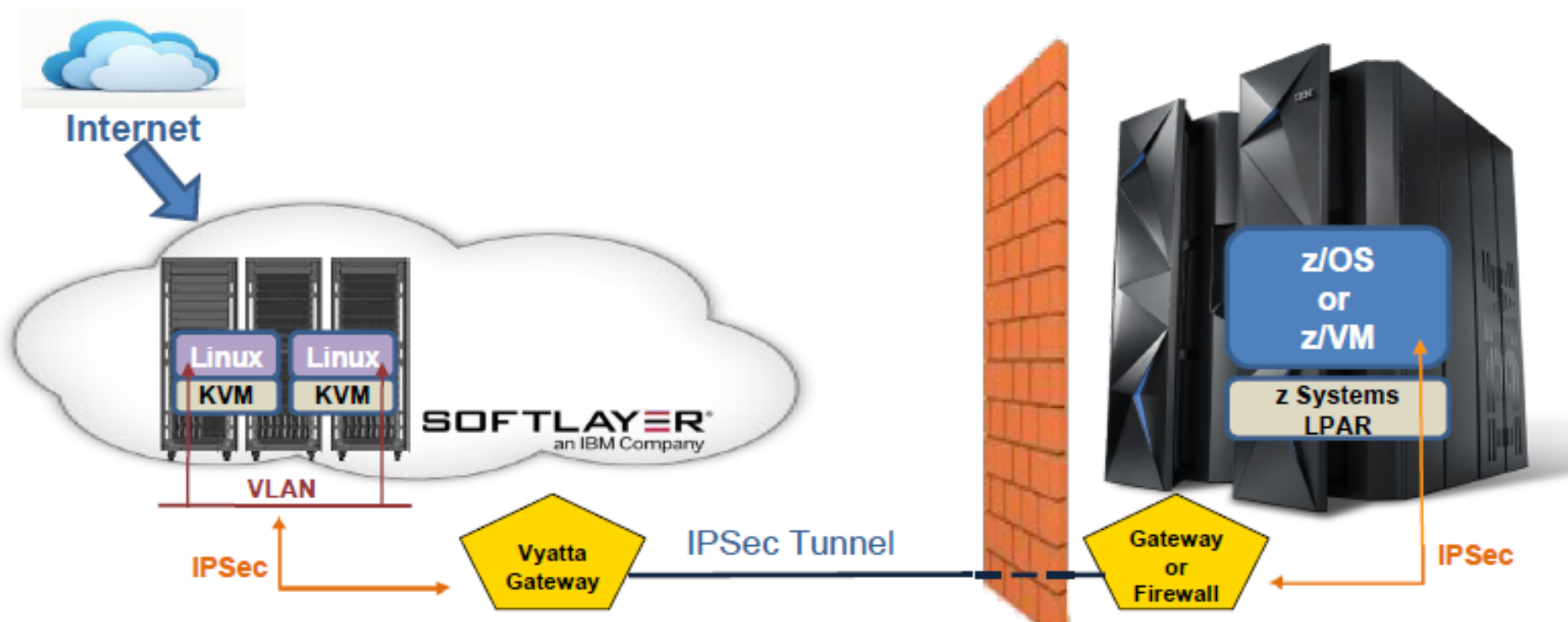


The SoftLayer server, located in Amsterdam, has only the Loyalty program details, thus respecting the privacy laws, sends a push notification to the mobile of M. Majorana

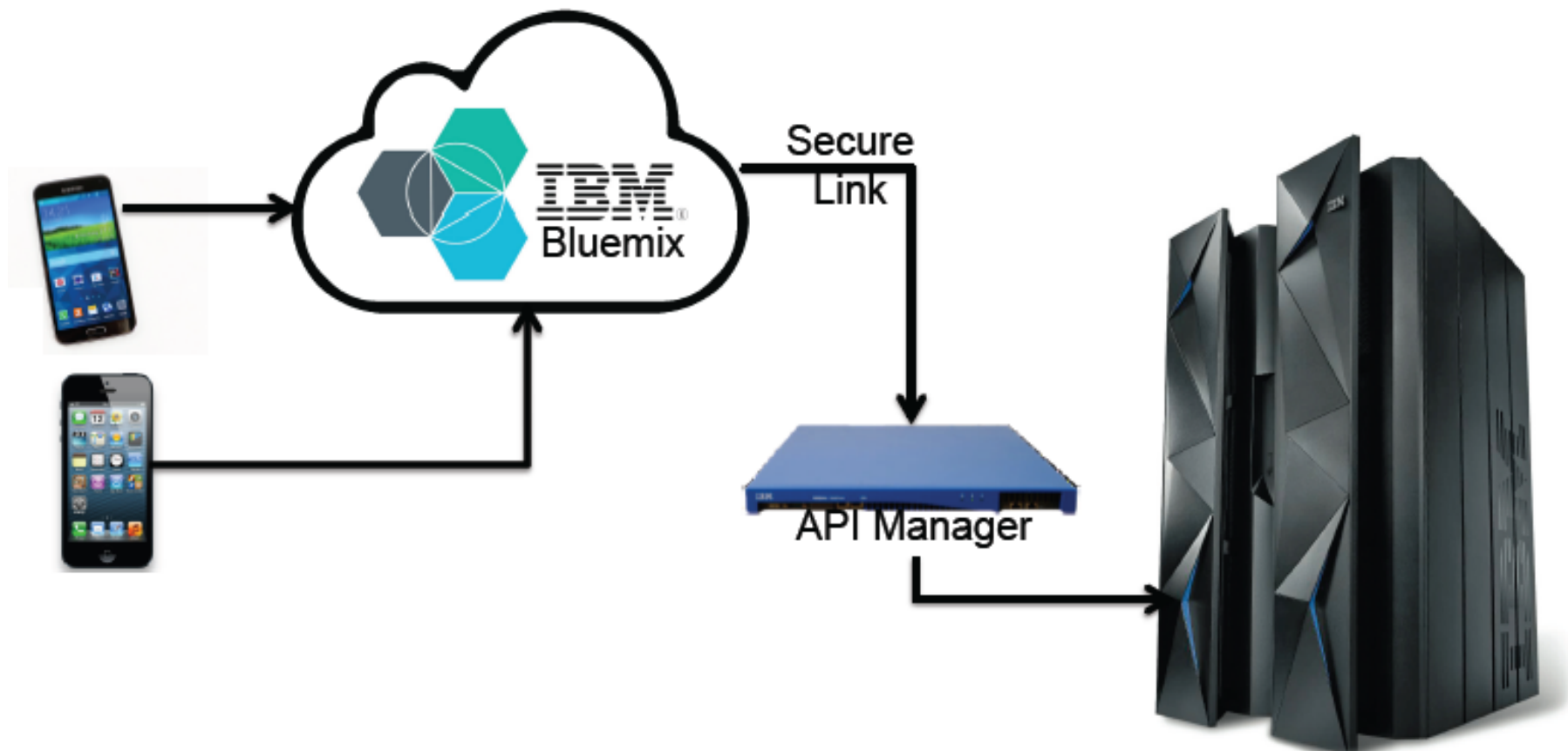
Loyalty program separation



z Systems Hybrid Cloud Example



The mainframe and IBM Bluemix



What is IBM Bluemix?



















IBM Bluemix™

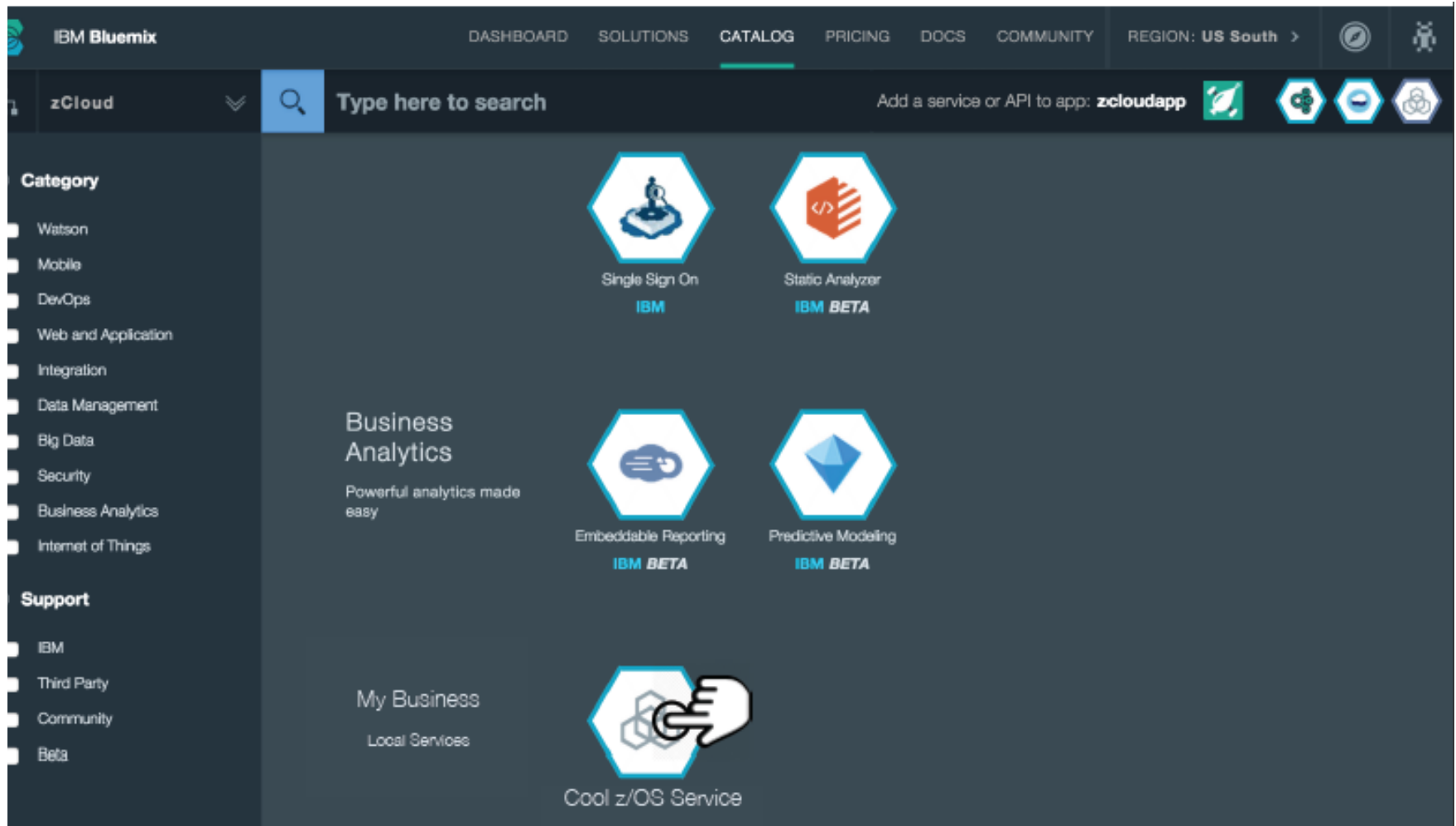
Starters // Choose a package of sample code and services, or start from scratch

Boilerplates

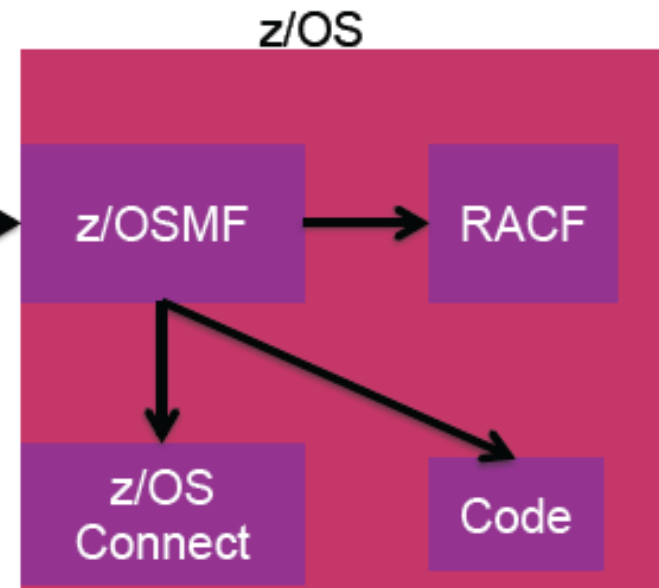
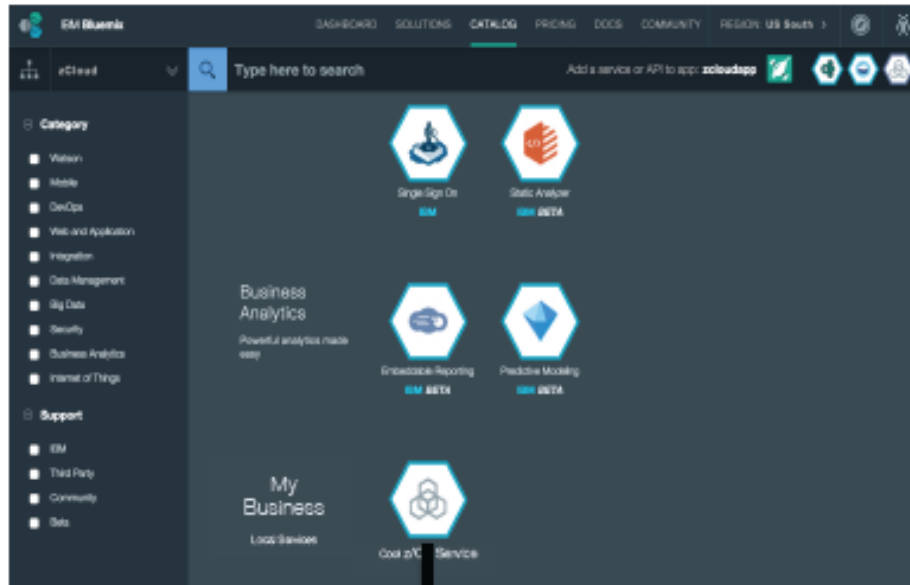
Get started with a new app, now

 Apache Spark Starter IBM	 Internet of Things Foundation Starter IBM	 Java Cache Web Starter IBM	 Java Cloudant Web Starter IBM	 Java DB Web Starter IBM	 LoopBack Starter IBM
 Mobile Cloud IBM	 MobileFirst Services Starter IBM	 Node.js Cache Web Starter IBM	 Node.js Cloudant DB Web Starter IBM	 Personality Insights Java Web Starter IBM	 Personality Insights Node.js Web Starter IBM
 StrongLoop Arc IBM	 Node-RED Starter Community	 Ruby Sinatra Community	 Vaadin Rich Web Starter Community		

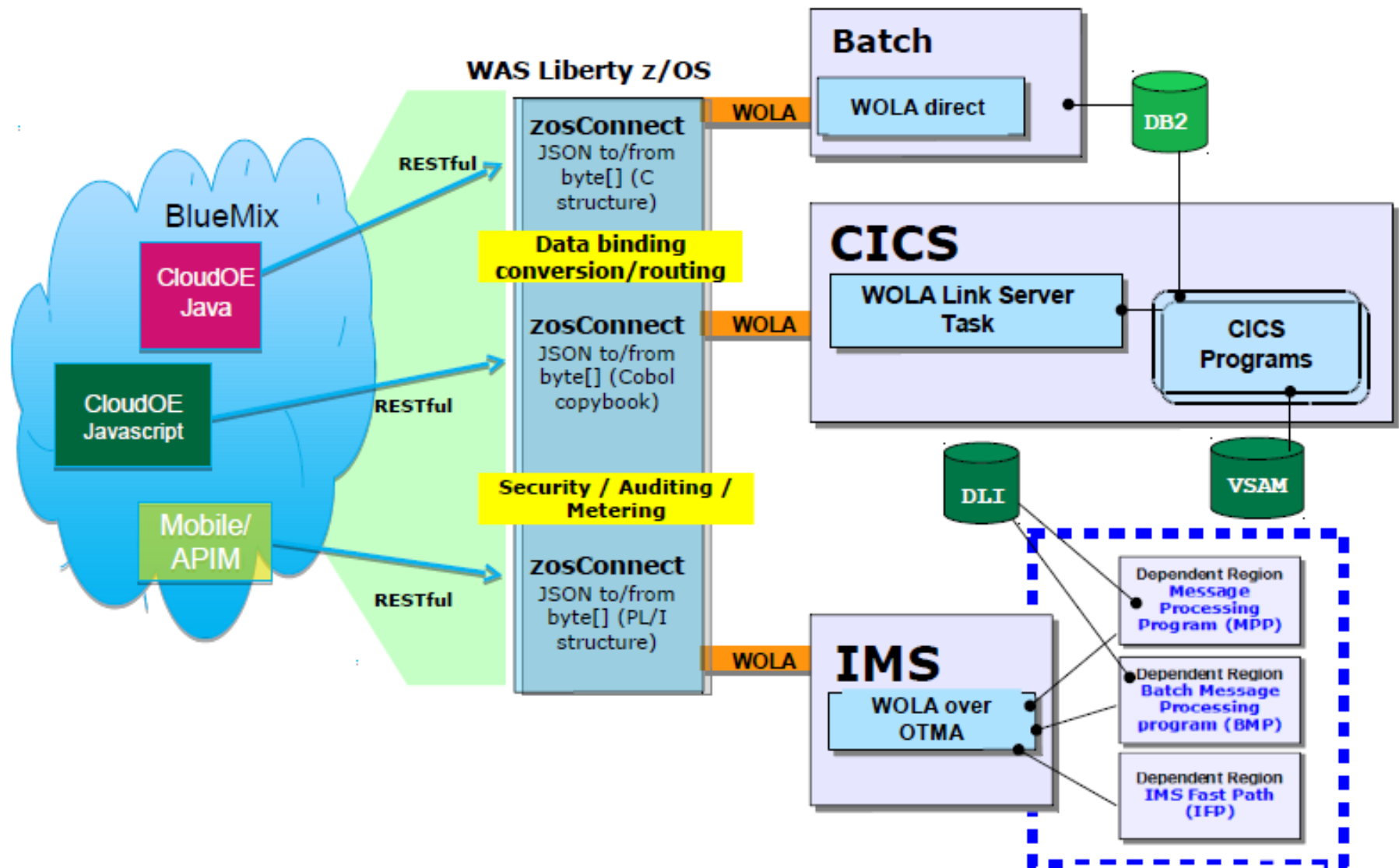
Bluemix services and APIs



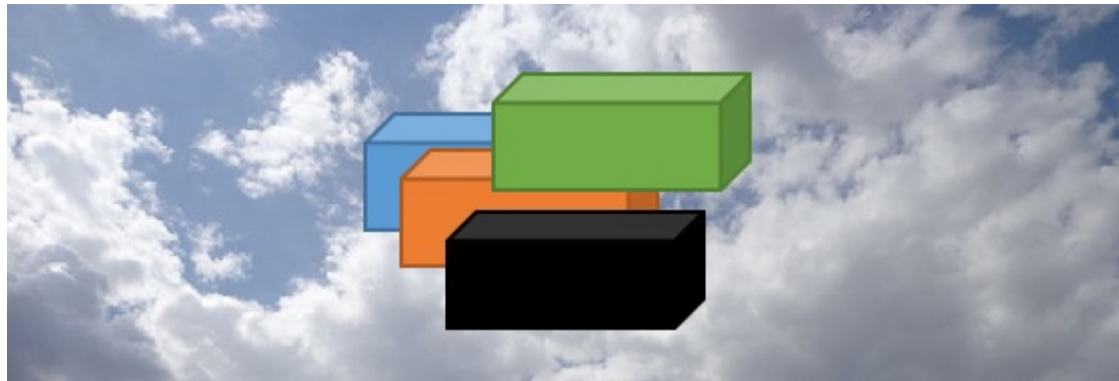
Connecting Bluemix to z/OS



Hybrid cloud: z/OS Connect and Bluemix



- Container technology provides an easy way to make applications more mobile in a hybrid cloud
- Containers are much smaller in size than virtual machines
- Containers provide more freedom in placing workloads
- Container technology will be integrated in OpenStack
- Docker is a consumer of containers



The Open Container Project

IBM

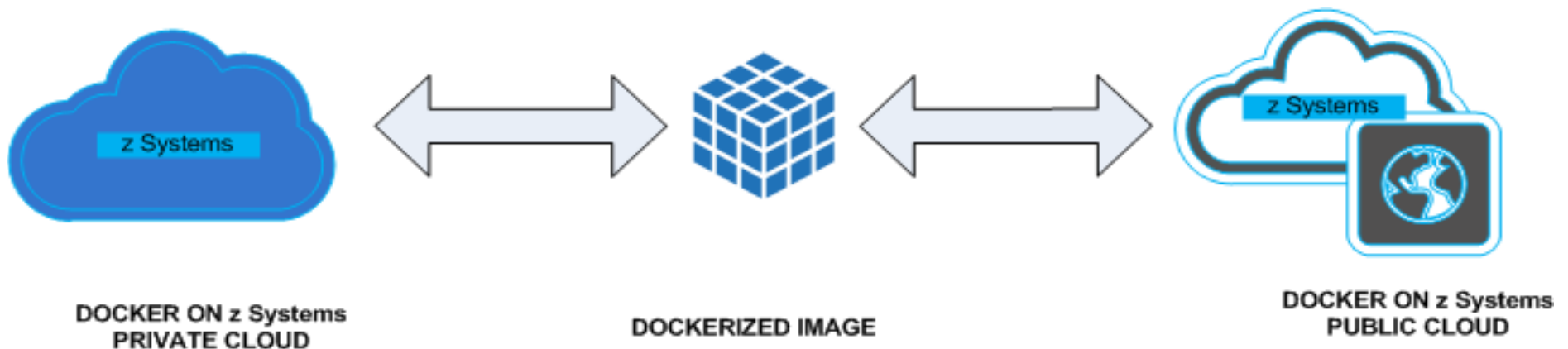


Application and run-time deployment

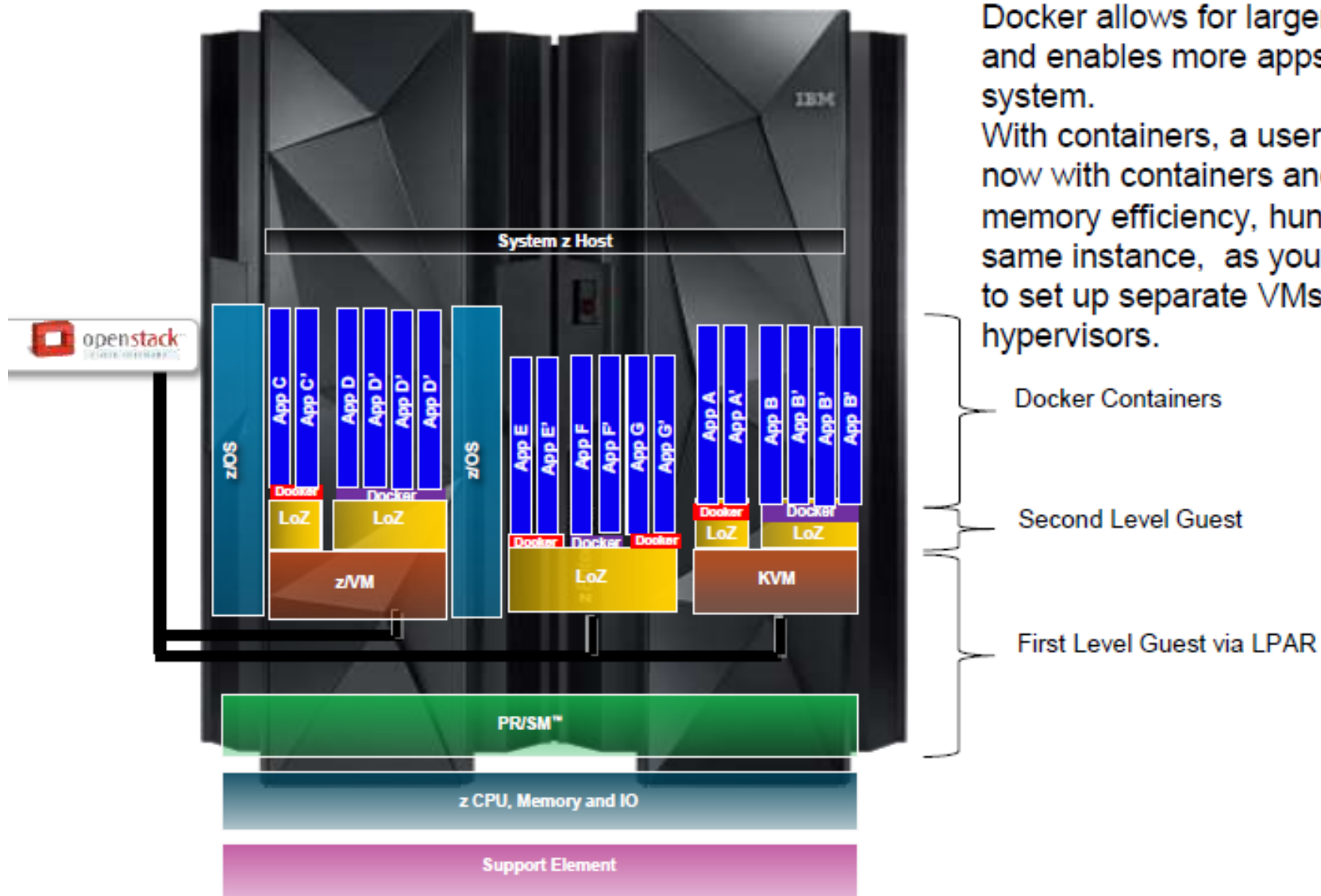


Container technology provides an easy way to make applications more mobile in a hybrid cloud

DevOps Hybrid Cloud : Docker Architecture on z Systems



Virtualization options and Docker on z Systems



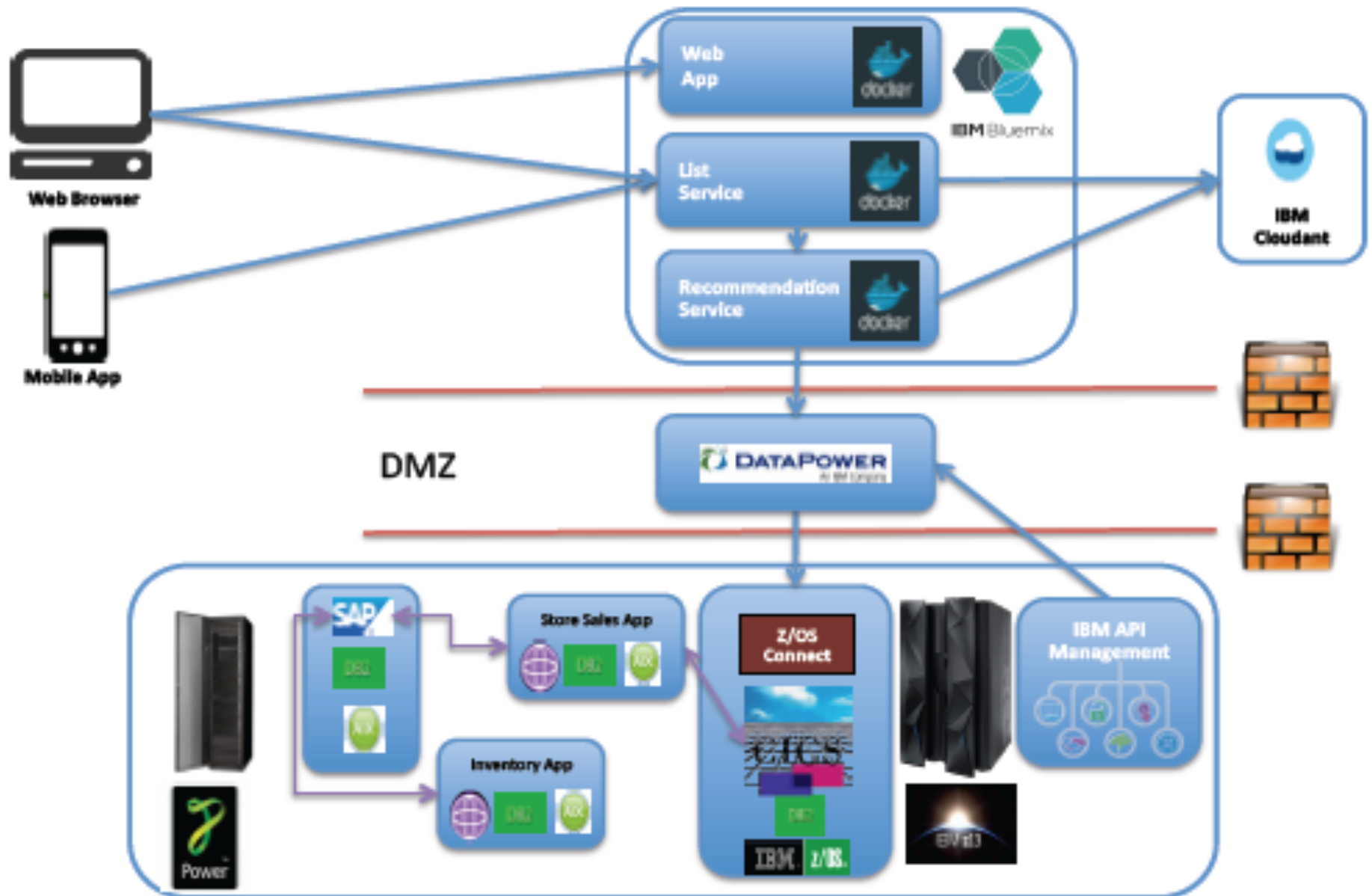
Docker allows for larger density and enables more apps in one system. With containers, a user could run now with containers and its memory efficiency, hundreds of the same instance, as you don't have to set up separate VMs with hypervisors.

Docker Containers

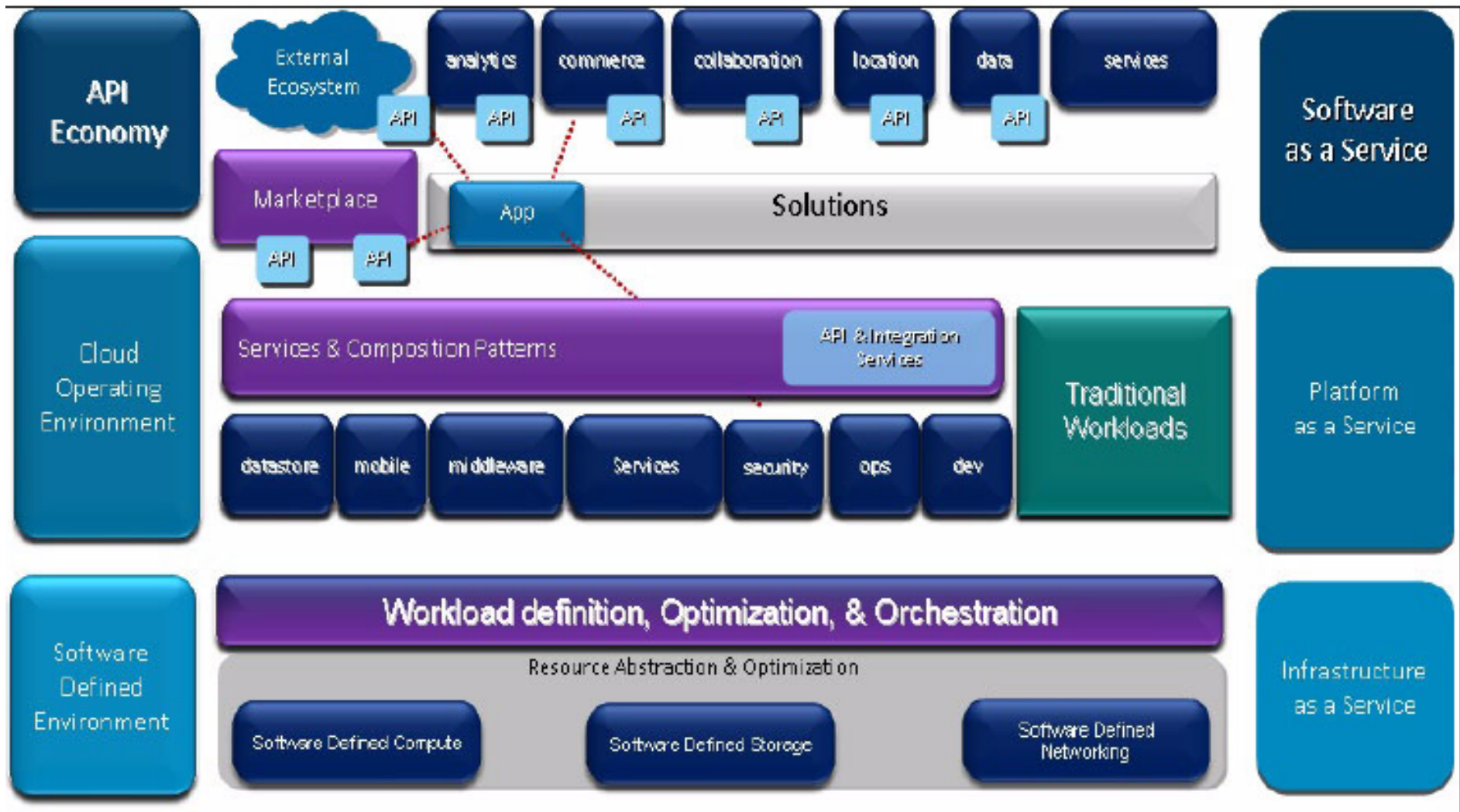
Second Level Guest

First Level Guest via LPAR

Docker containers on Bluemix



The cloud service delivery model.....



Things to Remember.....



- **What is the problem you are trying to solve?**
- **What is meant by “We should be doing cloud computing?” (ask the three questions)**
- **Focus on the future directions of cloud: hybrid cloud, containers and the API economy.**
- **z Systems can provide cloud-like service delivery if that is what your business requires.**
- **Make sure you, and z Systems, are a part of the cloud conversation at your shop.**



Don't Let the Cloud Fog Your Vision!





2016

IBMTechU Comes to You

IBM z Systems • IBM Power Systems • IBM Storage

12-14 April 2016

Atlanta | Georgia

10-12 May 2016

San Francisco | California

18-20 May 2016

Chicago | Illinois

7-9 June 2016

Boston | Massachusetts

2-4 November 2016

Baltimore | Maryland

14-16 November 2016

Austin | Texas

ibm.com/training/events