

# IBM Bluemix PaaS

## Java Developers Dream come true

Presented by:

IBM Cloud Ecosystem  
Development

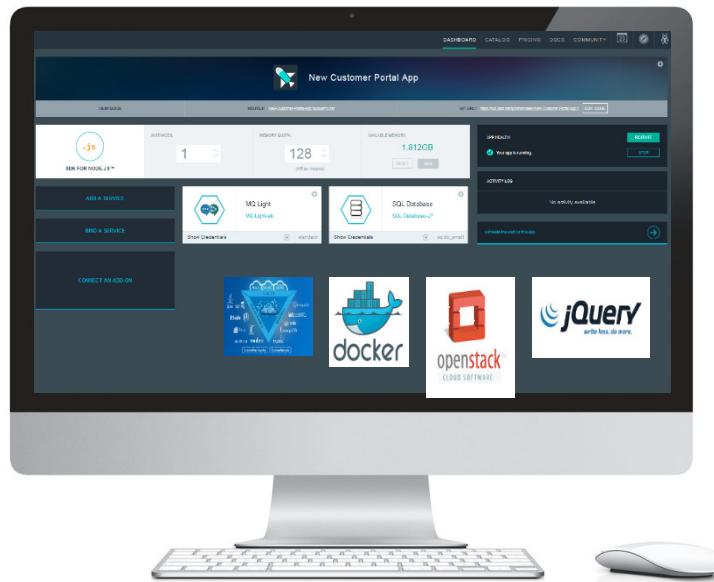
IBM Cloud

# Bluemix: IBM's Cloud Platform

Build, run, scale, manage, integrate & secure applications in the cloud

## Developer experience

- Rapidly deploy and scale applications in any language.
- Compose applications quickly with useful APIs and services and avoid tedious backend config.
- Realize fast time-to-value with simplicity, flexibility and clear documentation.



Built on a foundation of **open technology**.

## Bluemix service categories

- DevOps
- Big Data
- Mobile
- Watson
- Business Analytics
- Database
- Web and application
- Security
- Internet of Things
- Cloud Integration

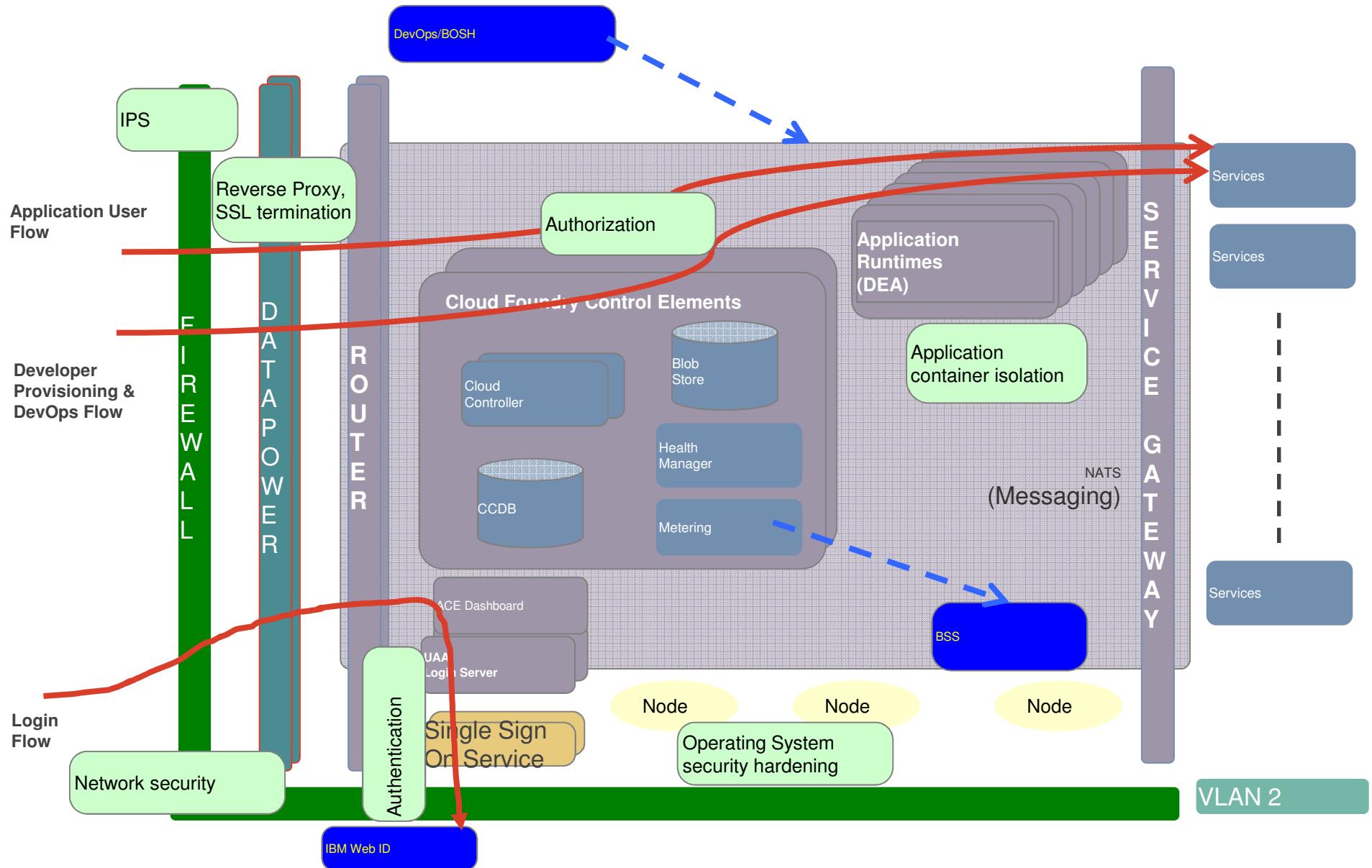
## Enterprise capability

- Securely integrate with existing on-prem data and systems.
- Choose from flexible deployment models.
- Manage the full application lifecycle with DevOps.
- Develop and deploy on a platform built on a foundation of open technology.

# Bluemix is becoming part of complex solutions

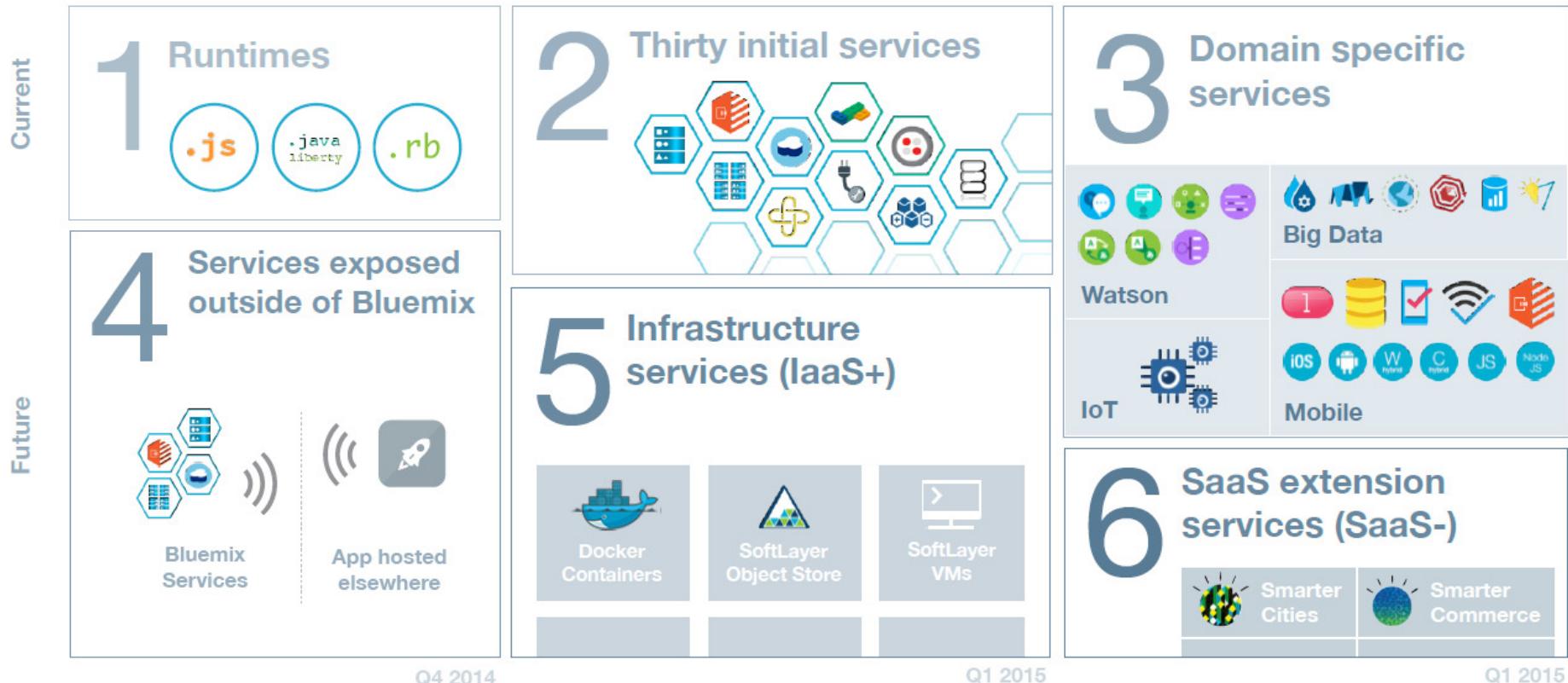


# Bluemix Platform Architecture



# Evolving Bluemix Platform

## Bring full range of developer services into Bluemix



## Services

# Integration

# Portability

# Flexible Deployment Models

# Java Buildpacks

## Intro to Liberty Buildpack

# Responsibility of Liberty Buildpack

- Converts VCAP\_SERVICES and VCAP\_APPLICATION environment variables into configuration variables
  - Variables end up in runtime-vars.xml
- Generates configuration for bound services
- Auto wires resource references to cloud services
- Starts & Stops Liberty server
  - Controls process environment
- Generates Liberty server configuration for war/ear files
- Pulls additional Liberty runtime packages as needed by Apps

# Liberty Buildpack vs Java Buildpack

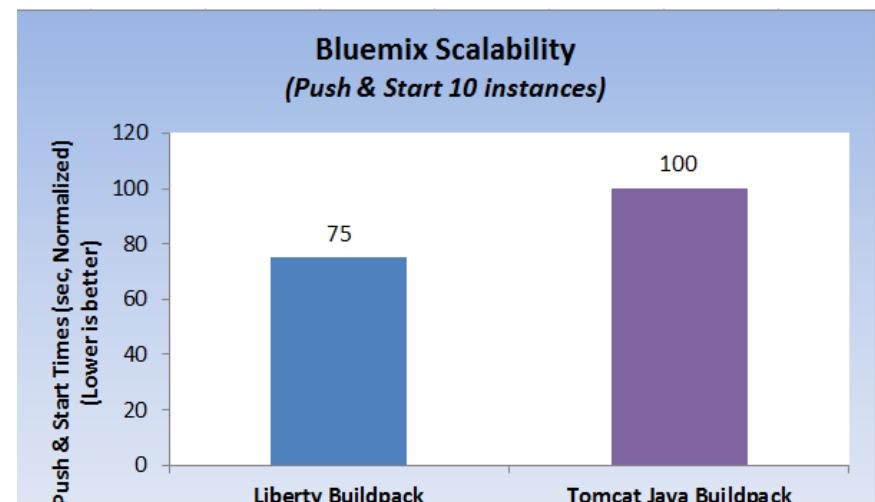
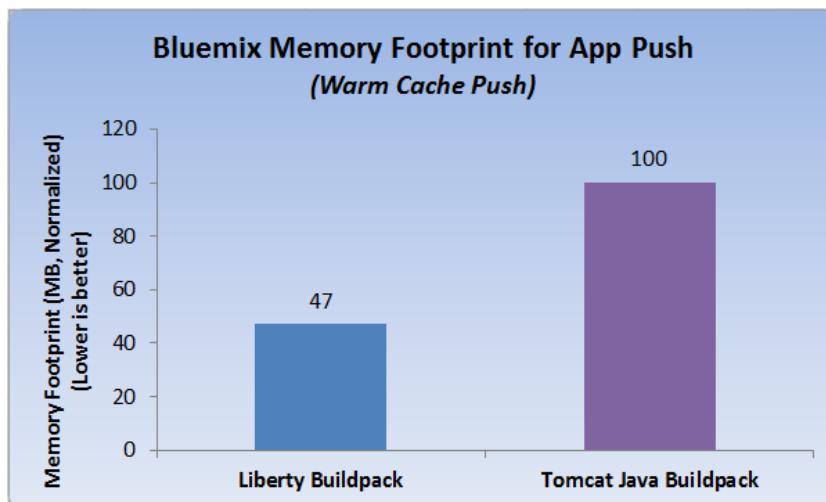
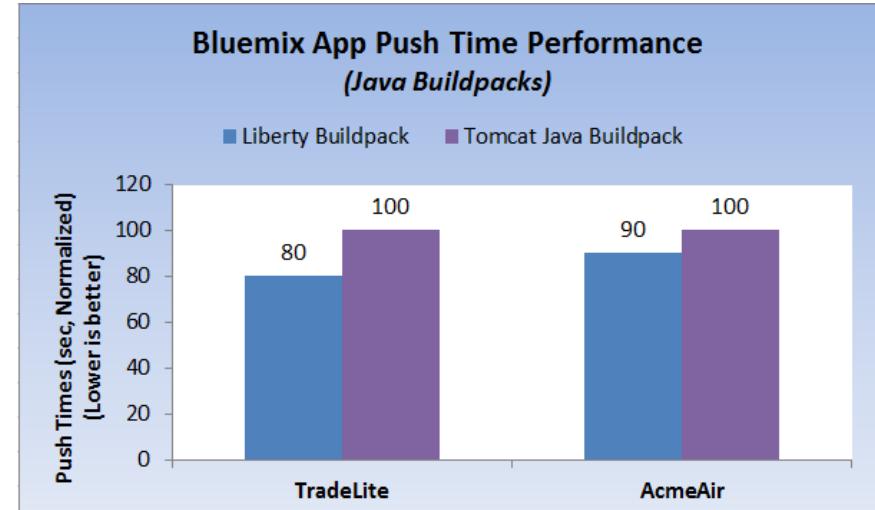
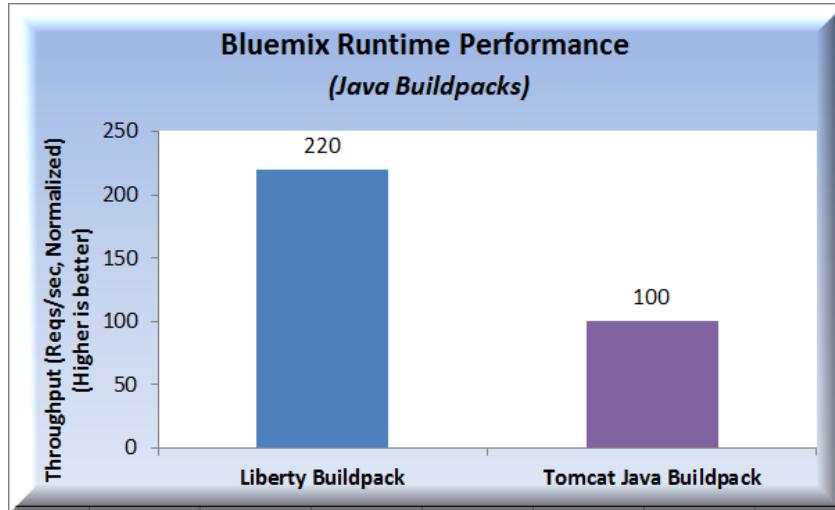
	Liberty buildpack	Java buildpack
<b>Application server</b>	WebSphere Liberty profile – Java EE 6/7 Web Profile & Beyond	Apache Tomcat 8 – Servlet 3.1, JSP 2.3, EL 3.0, WebSocket 1.1
<b>Java runtime</b>	IBM JRE or OpenJDK	OpenJDK
<b>Supported Java Web applications</b>	WAR, EAR, server directory or package	WAR
<b>Configuration</b>	Configuration can be provided with application or via environment variables	Must fork to customize configurations for many scenarios
<b>Service integration</b>	Auto-configuration for 14 services	Auto-configuration for 5 services
<b>Support statement</b>	IBM supported	Community supported
<b>Performance</b>	Better performance (push time, memory footprint, throughput, scaling time)	
<b>Developer aids</b>	Remote debugging & incremental update support, health center access, shell access	

# Easier configuration with Liberty buildpack

What it takes to make the following configuration change:

Configuration	with Liberty buildpack	with Java buildpack
JRE choice	Repush: IBM JRE or OpenJDK	Fork the buildpack
JRE overlay	Repush: include overlay files in app package	Fork the buildpack
Server customization	Repush: use server package	Fork the buildpack
JVM options	Repush or Restart	Repush

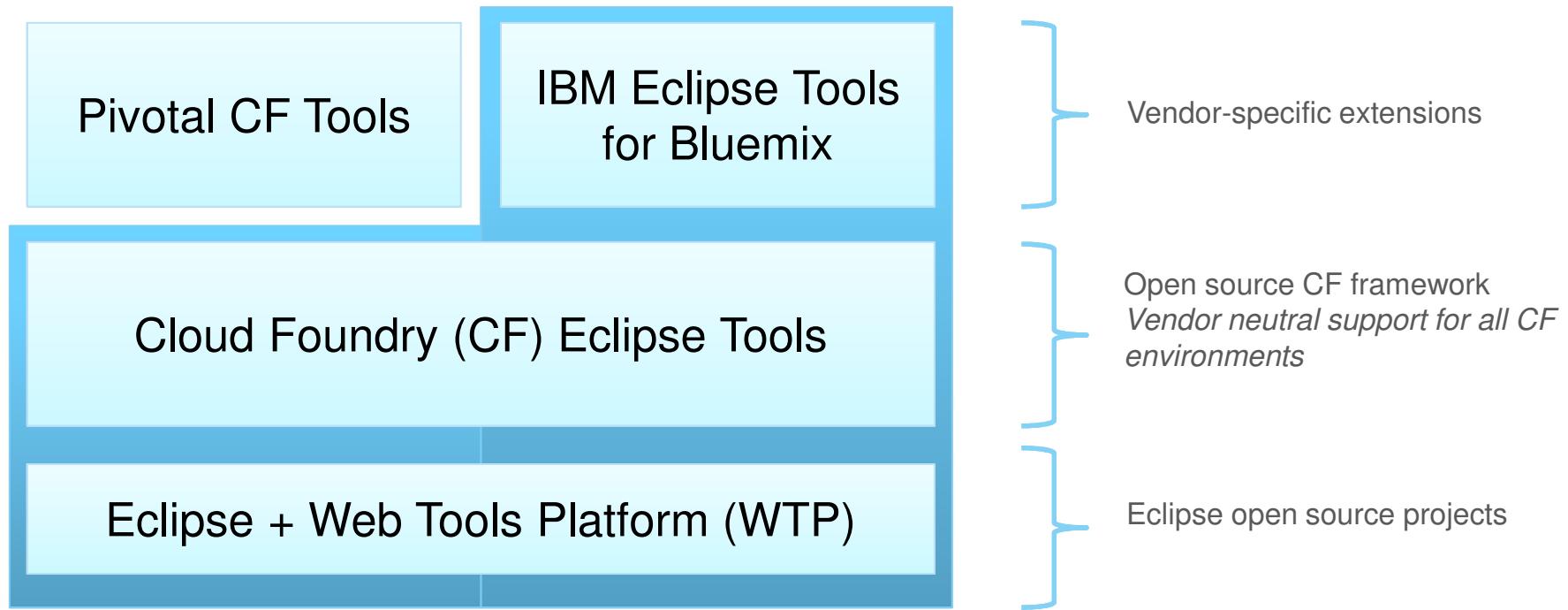
# Performance comparison



# A Java Developer Perspective

To the Cloud..

# Bluemix Tools at a glance



***We develop and contribute to all 3 layers (WTP, CF Eclipse Tools and IBM Eclipse Tools for Bluemix)***

# IBM Eclipse Tools for Bluemix

- Eclipse is IDE of choice
- Develop, build, deploy apps to Cloud quickly & easily
- Currently target IBM supported (Tier 1) runtimes
  - Liberty for Java
  - Node.js

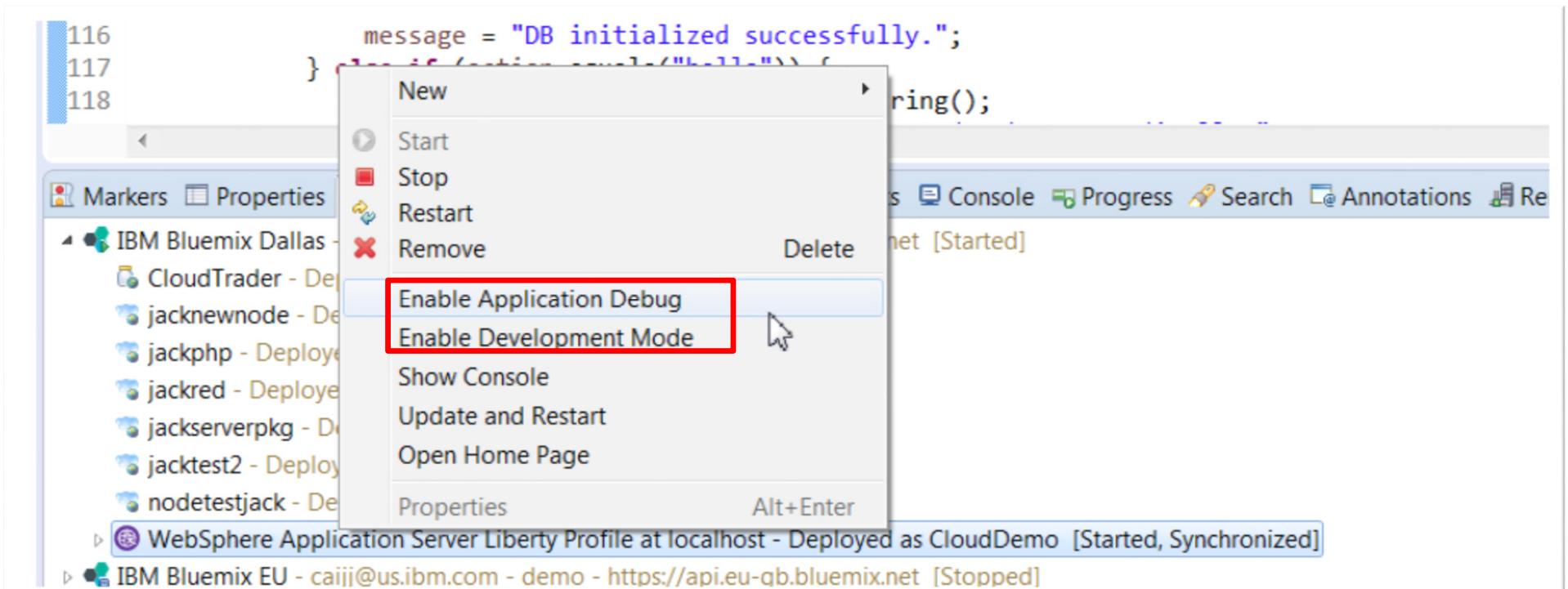
# IBM Eclipse Tools for Bluemix

## Powerful & Versatile DevOps tools

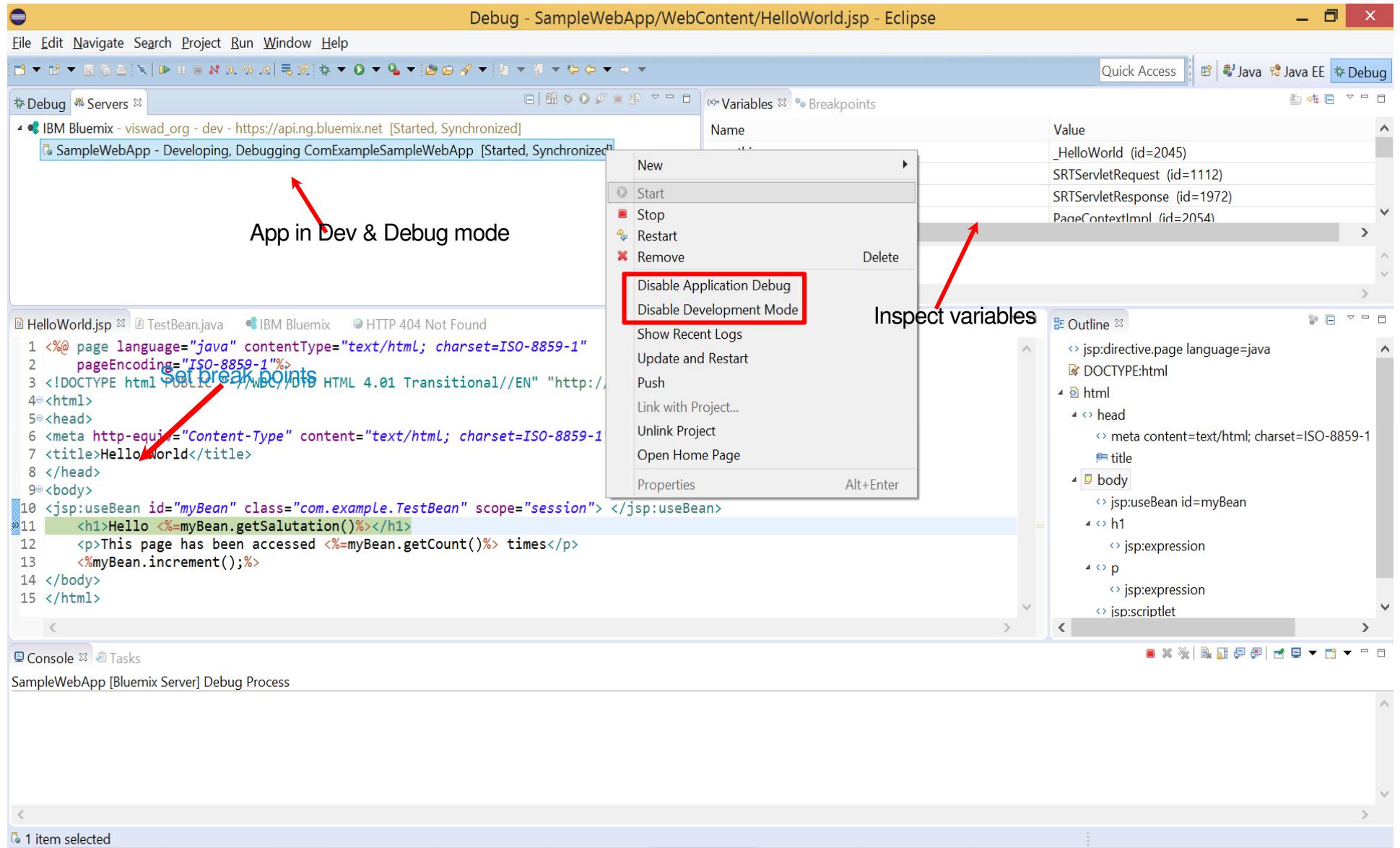
- Ops Tools for Server lifecycle:
  - Creation, Removal, Start, Stop, Restart
- Ops Tools for lifecycle of Applications
- Deploy, Remove, Update, Restart
- Dev Tools supporting Architect & Developer
  - Compose applications by creating and binding services
  - Run, Debug, Profile
  - Tune
- Local & Remote Support

# Development Mode

- Special mode of application instance
- Allows dev operations
  - Remote debugging
  - Incremental update

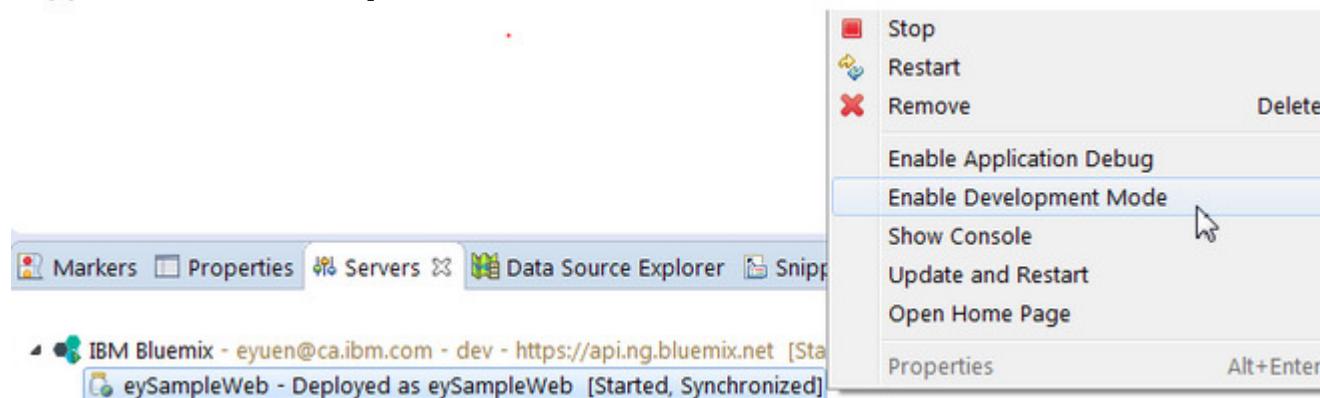


# Remotely Debug Applications in Bluemix



# Incremental Publish of Apps in Bluemix

- Enable Development Mode



- Enabling Development Mode provides access to
  - Push incremental file updates
  - Run additional tools inside app container like SSH web console

# Three Hands-on Labs

# Consuming Services

From Java Applications

# Using a service in Bluemix

- Create & Bind service

- Command line
    - *cf marketplace* to see available services
    - *cf create-service* to create a service instance
    - *cf bind-service* to bind the service instance to your application
    - *cf restart*, or *cf push* again
  - Web portal (bluemix.net)

- Read service connection and credentials

Sample VCAP\_SERVICES environment variable

```
{  
  "SQLDB-1.0": [  
    {  
      "name": "SQLDB-myDB",  
      "credentials": {  
        "hostname": "75.126.155.139",  
        "host": "75.126.155.139",  
        "port": 50000,  
        "username": "u123456",  
        "password": "CasDQ5v72u",  
        "db": "I_012345",  
        "jdbcurl": "jdbc:db2://75.126.155.139:50000/I_012345",  
        "uri": "db2://u123456:CasDQ5v72u@75.126.155.139:50000/I_012345"  
      }  
    }  
  ]  
}
```

# Sample code to connect to a SQL service

## Read the credentials

```
import com.ibm.nosql.json.api.*;
import com.ibm.nosql.json.util.*;

String VCAP_SERVICES = System.getenv("VCAP_SERVICES");
if (VCAP_SERVICES != null) {
    BasicDBObject obj =
        (BasicDBObject) JSON.parse(VCAP_SERVICES);
    String thekey = null;
    Set<String> keys = obj.keySet();
    for (String eachkey : keys)
        if (eachkey.contains("SQLDB"))
            thekey = eachkey;
    BasicDBList list = (BasicDBList) obj.get(thekey);
    obj = (BasicDBObject) list.get("0");
    obj = (BasicDBObject) obj.get("credentials");
    databaseHost = (String) obj.get("host");
    databaseName = (String) obj.get("db");
    port = (String) obj.get("port").toString();
    user = (String) obj.get("username");
    password = (String) obj.get("password");
    url = (String) obj.get("jdbcurl");
    // Use the jdbcurl or construct your own
    databaseUrl = "jdbc:db2://" + databaseHost + ":" +
                  port + "/" + databaseName;
```

# Sample code to connect to a SQL service

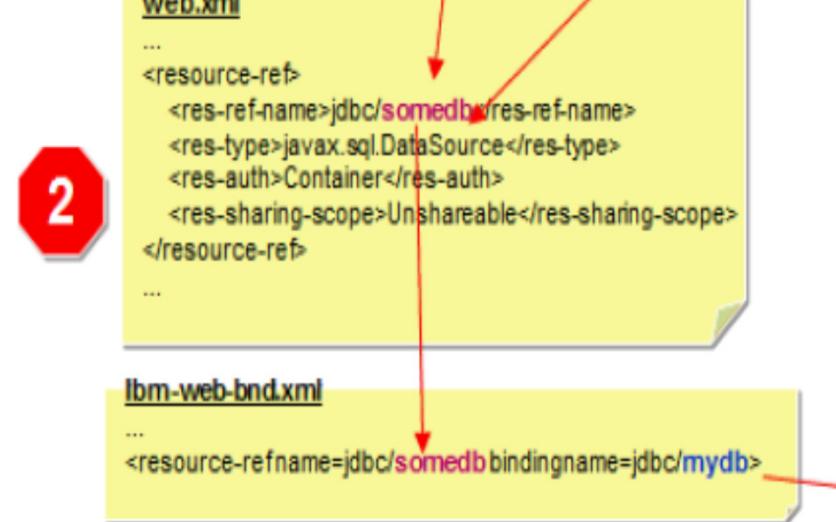
## Establish the connection

```
Connection con = null;
try {
    Class.forName("com.ibm.db2.jcc.DB2Driver");
    con = DriverManager.getConnection(databaseUrl, user, password);
    con.setAutoCommit(false);
} catch (SQLException e) {
    writer.println("SQL Exception: " + e);
    return;
} catch (ClassNotFoundException e) {
    writer.println("ClassNotFoundException: " + e);
    return;
}
Statement stmt = null;
try {
    stmt = con.createStatement();
    sqlStatement = "CREATE TABLE SALES.CUSTOMER (NAME VARCHAR(20))";
    stmt.executeUpdate(sqlStatement);
} catch (SQLException e) {
    writer.println("Error creating table: " + e);
}
```

# Java EE Resource Auto-wiring for services

```
Import javax.sql.DataSource;

public class MyServlet extends HttpServlet
{
    @Resource(name="jdbc/somedb")
    private DataSource myDataSource;
    ...
    InitialContext ctx = new InitialContext();
    DataSource ds = (DataSource) ctx.lookup("jdbc/somedb");
}
```



- ① Developer uses standard methods to access Data Source (JNDI lookup or injection).
- ② Developer provides standard JEE resource references to declare dependency on external resources.
- ③ Developer binds database services to the application.



- If a single database service is bound, then any/all bindings are bound to it.
- If multiple database services are bound - JNDI

# How does Auto Wiring work ?

- Look for resource using JNDI name  
`java:comp/env/jdbc/myDB`
- Bind service with name ‘myDB’ which is mapped to JNDI name `jdbc/myDB`
- Bound service returned as result of lookup

# Auto configuration of Liberty server

During App Staging :

Buildpack service plugin examines VCAP\_SERVICES →  
Generates server.xml snippets

Before	After
<pre>&lt;server&gt;   &lt;featureManager&gt;     &lt;feature&gt;webProfile-6.0&lt;/feature&gt;   &lt;/featureManager&gt;   ... &lt;/server&gt;</pre>	<pre>&lt;server&gt;   &lt;featureManager&gt;     &lt;feature&gt;webProfile-6.0&lt;/feature&gt;     &lt;feature&gt;eXtremeScale.webapp-   1.1&lt;/feature&gt;   &lt;/featureManager&gt;   ...   &lt;xsWebApp     objectGridName="\${cloud...}"     catalogHostPort="\${cloud...}"     securityEnabled="\${cloud...}" /&gt; &lt;/server&gt;</pre>

# Auto configuration of Liberty server

- Opt-out of Auto configuration when :
  - Generated config is not correct
  - Service plugin cannot discern intent correctly
- Opt-out per service
  - Set services\_autoconfig\_excludes environment variable

# Services easy to consume in IBM Cloud

## Java EE standard resources



SQL Database



BLU Data Warehouse



Elastic MQ

## Modern “resources”



Cloudant



Data Cache



Mongo DB

## Operational services



Session Cache



Auto Scaling



Monitoring and Analytics



Log Analysis

# Using SQLDB service

- SQLDB is on-demand relational database, powered by DB2.
- How you can use it?
  - The plain “VCAP\_SERVICES” way
  - The Java EE way
    - DataSource
    - JPA



# Java EE way – Sample code for SQLDB

```
public class TestServlet extends HttpServlet
{
    @Resource (name = "jdbc/mydb") ←
    private DataSource db;
    ...
    protected void doGet(HttpServletRequest request, HttpServletResponse
    response) throws ServletException, IOException {
        // Alternatively use InitialContext lookup
        DataSource lookup = (DataSource) new InitialContext().lookup("jdbc/mydb");
        ...
    }
}
```

“mydb” is the name of service instance you create in Bluemix

That's it! All familiar code, no changes required to make it work in cloud!

- No need for a server.xml
- Don't need to read VCAP\_SERVICES

# Using ElasticMQ service – familiar way again!

Develop responsive, scalable applications with a fully managed messaging provider in cloud.



```
public class TestServlet extends HttpServlet
{
    @Resource (name = "jms/emq")
    private ConnectionFactory cf;
    ...
    protected void doGet(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
        // Alternatively use InitialContext lookup
        ConnectionFactory lookup =
            (ConnectionFactory) new InitialContext().lookup("jms/myemq");
        ...
    }
}
```

# More services accessible same way



Cloudant

Cloudant's distributed database as a service (DBaaS) allows developers of fast-growing web and mobile apps to focus on building and improving their products, instead of worrying about scaling and managing databases on their own.



DataCache

Improve the performance & user experience of web applications by retrieving information from fast, managed, in-memory caches, instead of relying entirely on slower disk-based databases.



mongodb

MongoDB is an open source document database and the leading NoSQL database that is owned by MongoDB Inc

# Cloudant

```
public class TestServlet extends HttpServlet
{
    @Resource (name = "cloudant/mycloudantdb")
    private CouchDbInstasnce db;
    ...
    protected void doGet(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
        // Alternatively use InitialContext lookup
        CouchDbInstance db = (CouchDbInstance) new
        InitialContext().lookup("java:comp/env/cloudant/mycloudantdb");
        CouchDbConnector dbc = _db.createConnector(DATABASE, true);
        CouchDocument dbentry = new CouchDocument();
        dbentry.setContent("testEntry");
        dbc.create(dbentry);
    }
}
```

“mycloudantdb” is the name of the service instance you create in Bluemix

# DataCache

```
public class TestServlet extends HttpServlet
{
    @Resource (name = "wxs/myGrid")
    private ObjectGrid og;
    ...
    protected void doGet(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
        // Alternatively use InitialContext lookup
        ObjectGrid og = (ObjectGrid) new
        InitialContext().lookup("wxs/myGrid");
        ...
    }
}
```

“myGrid” is the name of the service instance you create in Bluemix

# Mongo

```
import com.mongodb.DB;
```

```
public class TestServlet extends HttpServlet
{
    @Resource (name = "cloudant/mymongo")
    private DB ;

    ...
    protected void doGet(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
        // Alternatively use InitialContext lookup
        db = (DB) new
        InitialContext().lookup("java:comp/env/cloudant/mymongo");
        ...
    }
}
```

“mymongo” is the name of the service instance you create in Bluemix

**IBM Application Server on Cloud for Bluemix** reduces deployment time, increases consistency, and fosters agility. These benefits you likely expect when exploring cloud-based approaches for your WAS environments.



### WebSphere Application Server

Same WAS you Know and Love;  
Same features & capability but  
gaining the Cloud



#### Innovative App Server

Flexible runtime environment for a lightweight, dynamic application runtime or an advanced, large-scale application platform.

#### Rapidly deliver Rich Experiences

Comprehensive set of open standards programming models to better align project needs with capabilities and developer skills.

### Bluemix

Your Route to get your applications to the Cloud. Build, run, deploy and manage applications on the cloud with ease.

#### Extend apps with services

A catalog of IBM, third party, and open source services to grow your solution.

#### Flexible Pricing

Try some services for free and when you're ready, pay only for what you use. Pay as you go and subscription models offer choice and flexibility.

### IBM Application Server on Cloud for Bluemix

Takes WAS to the Next Level with comprehensive cloud orchestration to help you Extend and Evolve.

#### Lift & Shift WAS to Cloud

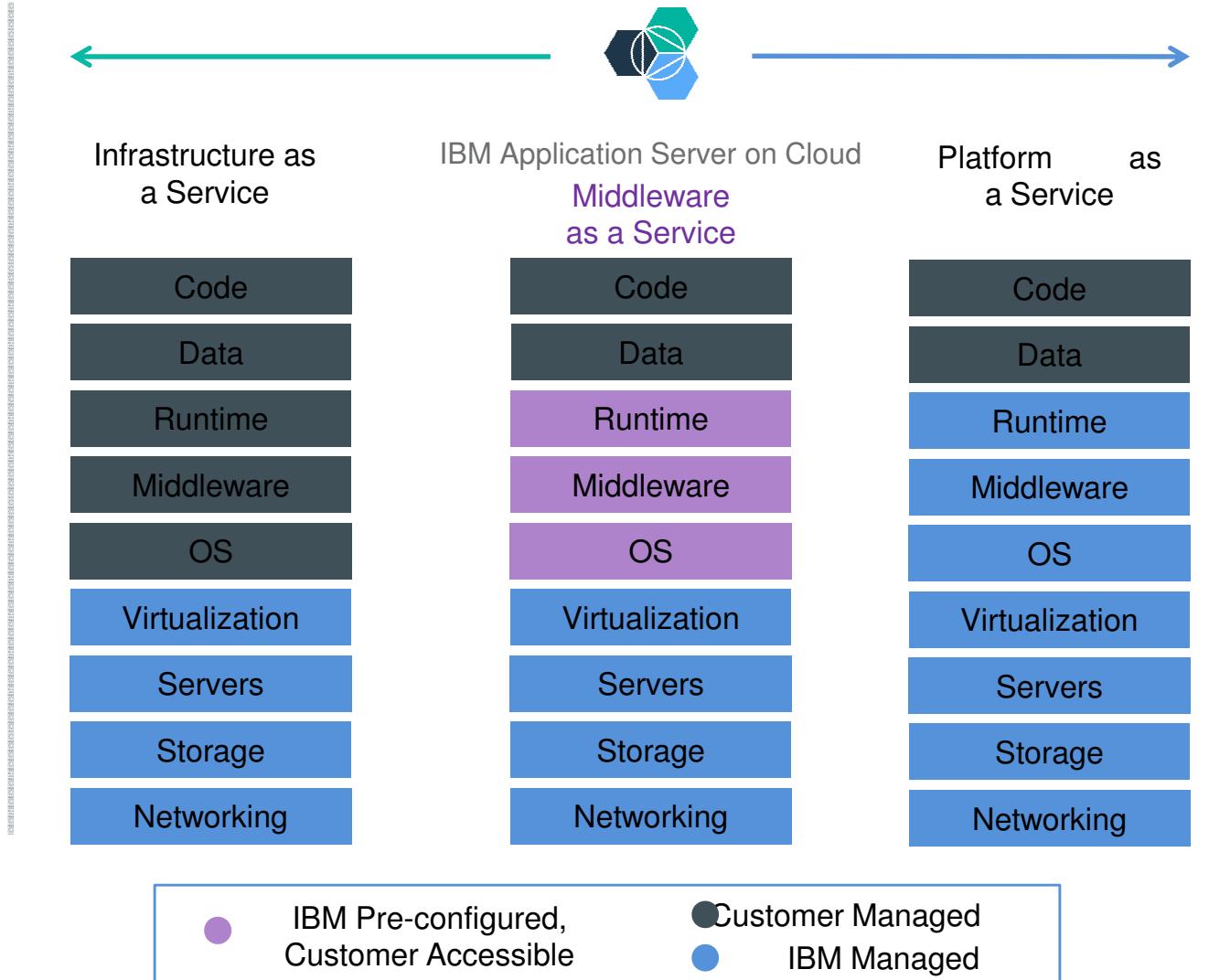
Rapidly bring your Apps to Cloud & Drive faster solution delivery ahead of the competition

#### Best of Both Worlds

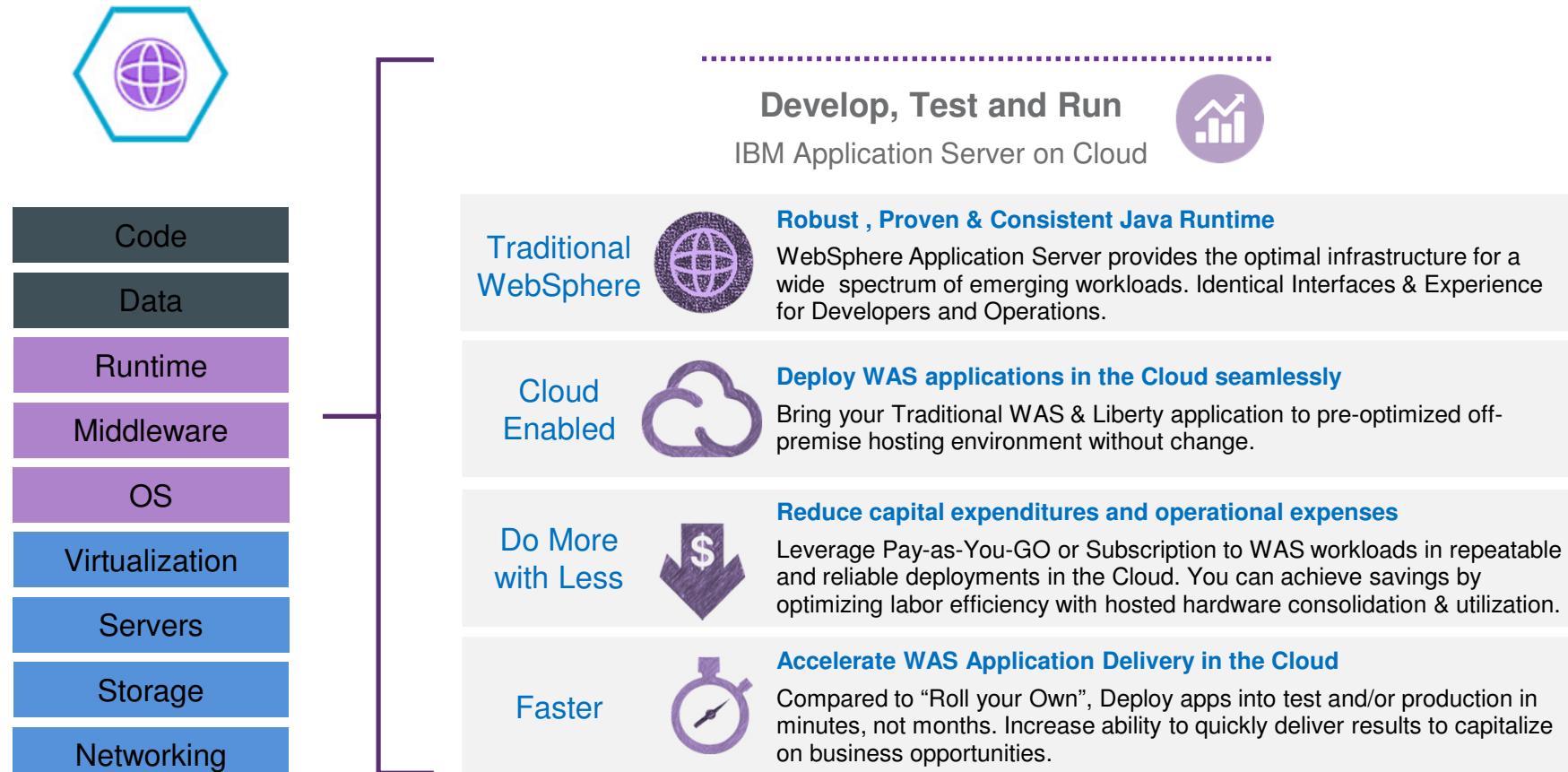
Balance Cloud Economics & Agility with WAS quality, security and governance.

IBM Application Server on Cloud provides **Middleware-as-a Service** for WebSphere Application Server Workloads.

Capabilities of Bluemix now span **IaaS** through **PaaS**.



# IBM Application Server on Cloud provides the **Same** WebSphere Application Server with the **Speed** and **Simplicity** of the Cloud!



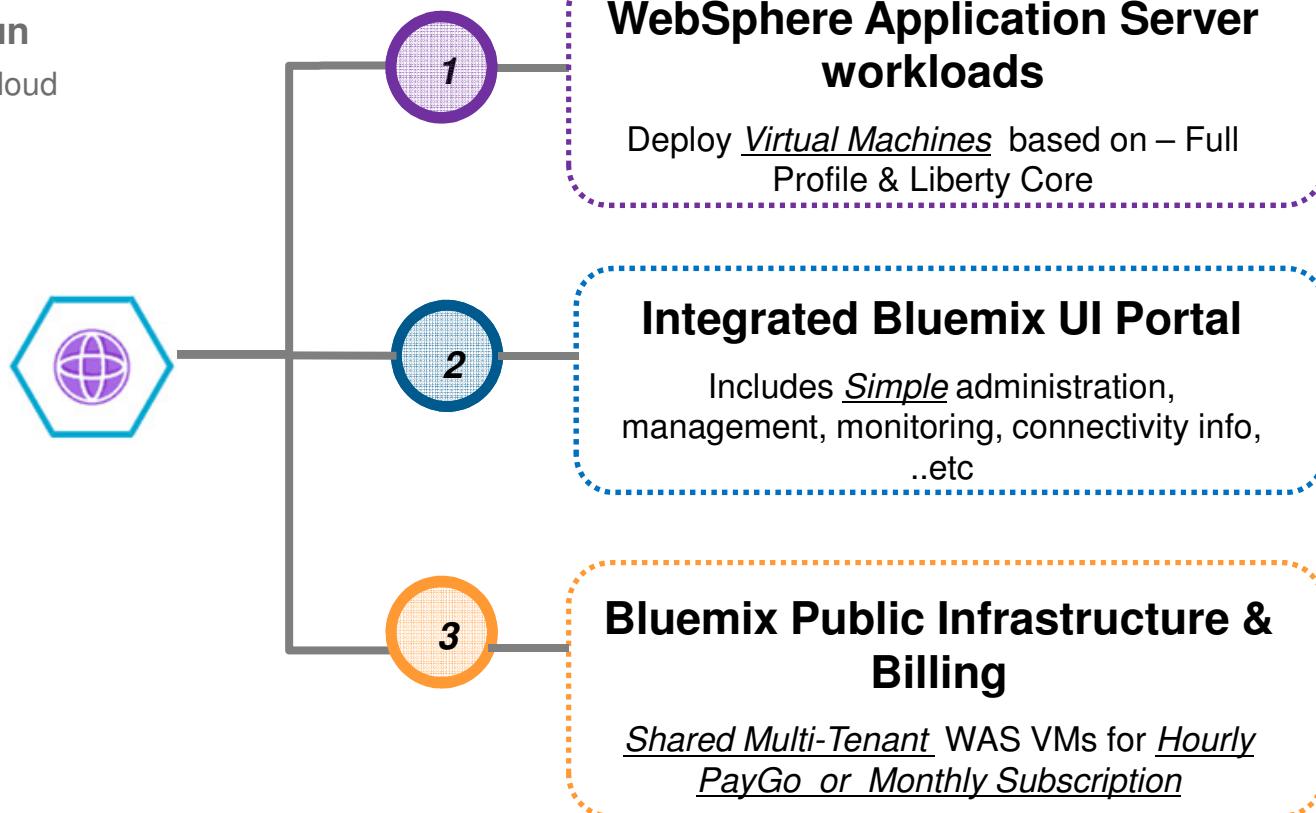
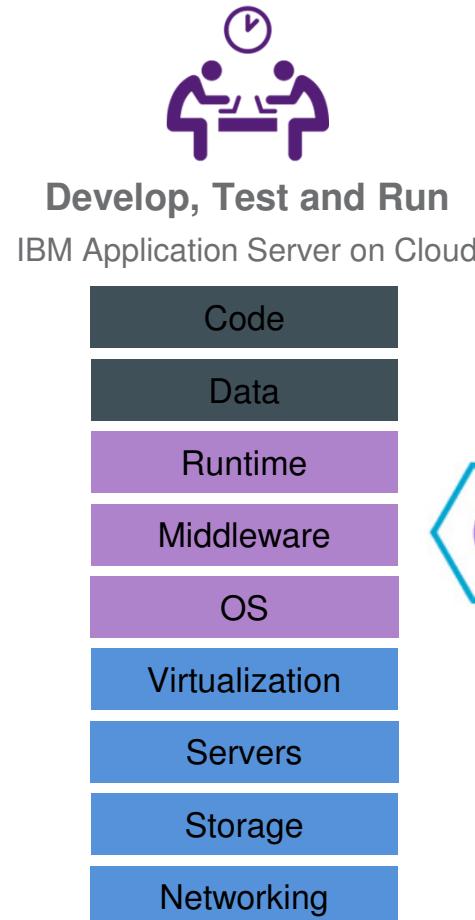
IBM Pre-configured,  
Customer Accessible



Customer Managed  
IBM Managed

# IBM Application Server on Cloud provides the Building Blocks ,

Standardized virtual machines, pre-configured through automation with self-serviced consumption based pricing.



# IBM Application Server on Cloud provides Portability from on-prem

Quickly get up and running on pre-configured WAS Classic and WAS Liberty Virtual Machines in a hosted cloud environment on Bluemix with Full control of the underlying OS & middleware.



 IBM Application Server on Cloud

Code  
Data  
Runtime  
Middleware  
OS  
Virtualization  
Servers  
Storage  
Networking

 IBM Application Server on Cloud  
IBM

PUBLISH DATE  
07/06/2015

TYPE  
Service

[VIEW DOCS](#)

BETA

The IBM Application Server on Cloud gives you the WebSphere Application Server experience in Bluemix. Choose between Full Profile or Liberty Profile of WebSphere Application Server, preconfigured and hosted on your own Red Hat Enterprise Linux guest.

- **WebSphere Experience**  
Reuse your WebSphere administration skills and scripts to ease your transition to cloud.
- **Shell and Admin UI Management**  
Use a secure shell, or log into the Admin Center/Console to control your WebSphere environment.
- **Use. Abuse. Repeat.**  
It's a cloud service after all. Use it for an hour. Drive it hard. Delete it. Grab a new one when you need it; it's up to you.
- **Full-Feature Application Servers**  
Leverage sophisticated enterprise features like transactions, just like you do on-premise.
- **Customize just what you need**  
We've already installed and configured WebSphere, now all you have to do is make those minor changes to get your application just the way you need it.





  
© 2015 IBM. All rights reserved. IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation, and/or its various direct and indirect wholly-owned and/or majority-owned subsidiaries. Other brands and product names might be trademarks of their respective holders. All other brand and product names are trademarks of the their respective holders.

**IBM Application Server on Cloud** includes a **Dashboard** providing **Ease of Administration** of Virtual Machine Environments allowing Customer to focus on Application Development & Application Integration



WebSphere Application  
Server Developers,  
Testers  
Administrators



## Service Dashboard

The screenshot shows the Service Dashboard for an Application Server on Cloud instance named 'w1'. It displays the following sections:

- Service Information:**
  - Service Type: Liberty Profile
  - Creation Time: September 10, 2015 1:07:38 PM
  - Lease Expires: September 24, 2015 1:07:38 PM
- Operating System Resources:**
  - OS Distribution: RHEL 6.6 X64
  - Host: 159.122.209.41
  - Admin Username: root
  - Admin Password: (link)
- WebSphere Resources:**
  - Admin Username: wsadmin
  - Admin Password: (link)
  - Admin Console: http://159.122.209.41/adminCenter

A central message says "Your virtual server is running." with "STOP" and "START" buttons.

## SSH Access to VMs

```

$ jbdDeploy.sh
$ enableProvisioning.sh
$ enableSheller.sh
$ eventBucket.sh
$ eventCatalog.sh
$ eventEmail.sh
$ eventFile.sh
$ eventFilepurgepool.sh
$ eventFilepurge.sh
$ eventFilestatus.sh
$ eventFileupgrade.sh
$ eventFileupgrade082.sh
$ eventFileupgrade082205.sh
$ eventFileupgradeOracle.sh
$ executePath.sh
$ executeQuery.sh
$ executeXSLT.sh
$ Extractor.sh
$ FindEJBTimers.sh
$ genHistoryReport.sh
$ GenPluginCfg.sh
$ genVersionReport.sh
$ hadoopHdi.sh
$ hadoopMapred.sh
$ -bash-4.1$ pud
$ /opt/IBM/WebSphere/Profiles/DefaultAppSrv01/bin
$ -bash-4.1$ ./serverStatus.sh -all
$ RMWU0116I: Tool Information is being logged in file
$ /opt/IBM/WebSphere/Profiles/DefaultAppSrv01/logs/serverStatus.log
$ RMWU0128I: Starting tool with the DefaultAppSrv01 profile
$ RMWU0503I: Retrieving server status for all servers
$ RMWU0505I: Servers found in configuration:
$ RMWU0506I: Server name: server1
$ RMWU0508I: The Application Server "server1" is STARTED
$ -bash-4.1$

```

The screenshot shows two side-by-side log-in pages:

- Liberty Admin Center:** A simple form with fields for "User Name" and "Password", and a "Log In" button.
- WebSphere Integrated Solutions Console:** A more complex form with fields for "User ID", "Password", and "Log In", accompanied by a logo featuring three stylized human figures and a key.

Both pages include the "WebSphere software" logo at the top.

Licensed Materials - Property of IBM (c) Copyright IBM Corp. 1997, 2011 All Rights Reserved. IBM, the IBM logo, ibm.com and WebSphere are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at [Copyright and trademark information](#).

## Admin Center and Admin Console Links

### Customer access:

- Documentation
- OpenVPN configuration
- Start and Stop virtual machines
- Hostname, Root & OS password.
- Private SSH key
- WebSphere Admin User & Password.
- Admin Center and Admin Console Urls.

# IBM Application Server on Cloud provides easy Data integration and Enterprise Connectivity

Integration and data capabilities are accessible making **hybrid application** architecture even easier and more flexible.



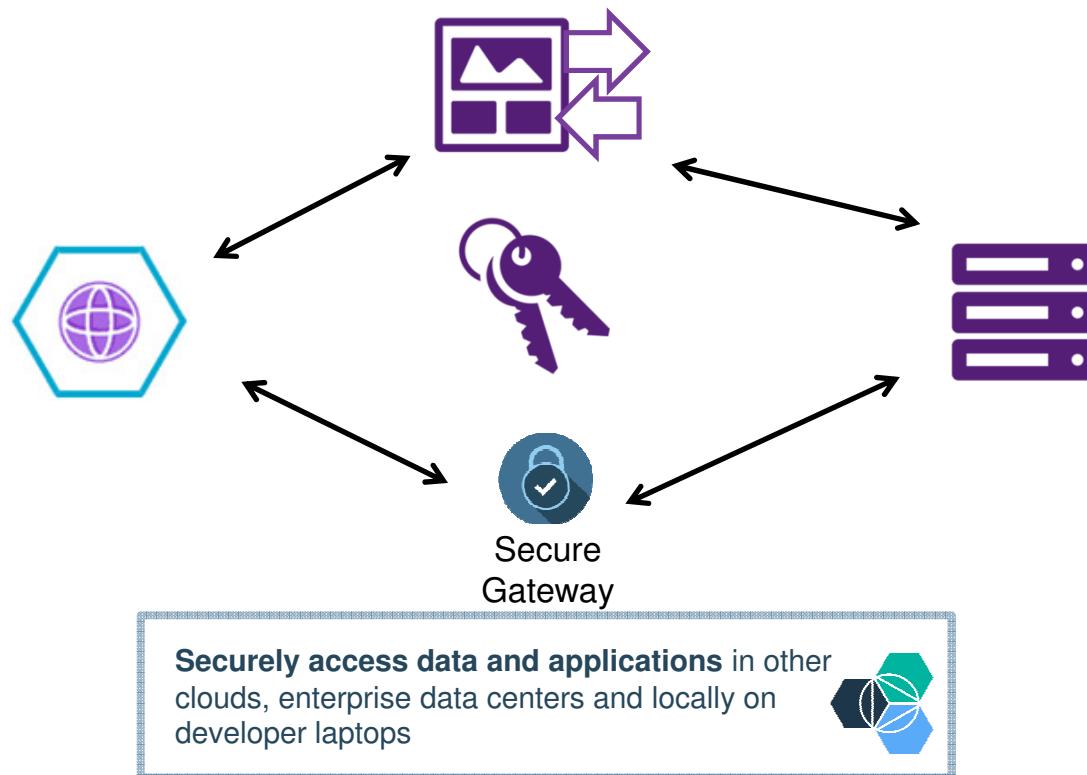
## Develop, Test and Run

IBM Application Server on Cloud

Code
Data
Runtime
Middleware
OS
Virtualization
Servers
Storage
Networking

## VPN

Leverage OpenVPN for secure connection to Systems of Record running in your network

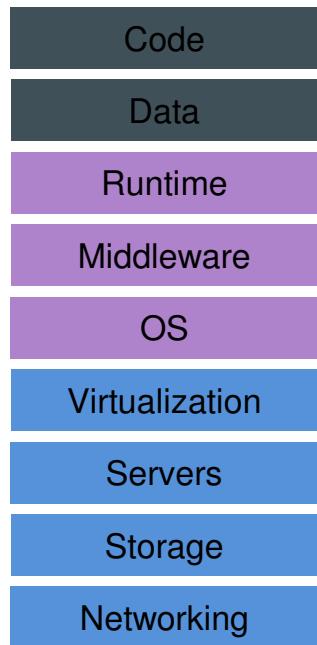


# IBM Application Server on Cloud popular Starter Usecases.



## Develop, Test and Run

IBM Application Server on Cloud



## Internal Pilot project

- Proving the business value of new projects
- Try out a newer version of WebSphere Application Server for short period of time

## Hybrid cloud development

- As a development environment for WebSphere Application Server
- Get started fast during on-prem WebSphere Application Server installation/set-up is in parallel

## Instead of their own WAS Installation

- Prefer Cloud solutions to on-prem Licensed product
  - Organizations with small IT departments
- As a target for clients struggling with their own on-prem or RYO WAS Cloud infrastructure
- Urgency for implementation (Quick Time to Value)



# Sign up in minutes. Pay for what you use.

Cloud based pricing models to serve customer needs.



## Friction free adoption

- **10 Instance Hours Included each Month** - Designed to allow testing of an application on the platform
- **Free tier for many Bluemix service** - encourages experimentation of new services for applications already running on Bluemix

## Multiple Commitment Models

- **Pay-as-you-go** - optimized for flexibility, no term commitment
- **Subscription** - term based optimized for cost, discounted from pay as you go rates

## Self Service

- Zero to coding in **less than 5 minutes**
- **Credit card** over the web in many countries - or through your IBM rep



### IBM manages:

- 24/7 Support
- Platform Configuration
- Platform Uptime
- Platform Monitoring
- Platform High Availability
- High Availability
- WAS Product Updates
- Platform Scheduled Maintenance with advanced notice



### Customer manages

#### **WAS Maintenance**

Version updates on customer's schedule



### **Capacity and scaling**

By default, we provide capacity in customer's regional data center. Additional capacity available by adding my Instances!

# IBM Application Server on Cloud provides Simple Plan Options

One Instance of a plan equates to **1vCPU with 2GB RAM, and 12Gig HD capacity**  
priced by **Instance-Hr**



## Develop, Test and Run

IBM Application Server on Cloud

Code
Data
Runtime
Middleware
OS
Virtualization
Servers
Storage
Networking



<b>BYOL Plan</b> Current Active WAS S&S Customer can bring licenses of WAS to run on this Offering. This plan will deploy Pre-configured WAS based on existing entitlement. ( <b>Subscription Only</b> )	\$0.30/Hr or \$222/Mo
<b>Liberty Core Plan</b> Pre-configured Instances based on Liberty Core	\$0.41/Hr or \$304/Mo
<b>WAS Base Plan</b> Pre-configured Instances based on WebSphere Application Server Full Profile.	\$0.52/Hr or \$389/Mo
<b>WAS ND Plan</b> Pre-configured Instances based on WebSphere Application Server Network Deployment	\$1.14/Hr or \$845/Mo
<b>(Optional) Bluemix Standard Support</b> Developers forum. Ticketing system with 24-hour responsiveness. Connect with IBM technical staff. Covers IBM Bluemix offerings and features.	10% of account charges, with a minimum monthly charge of \$200.00 USD

Approximate monthly Price based on 744 Instance Hours per month.

# Backup

# Using Operational Services

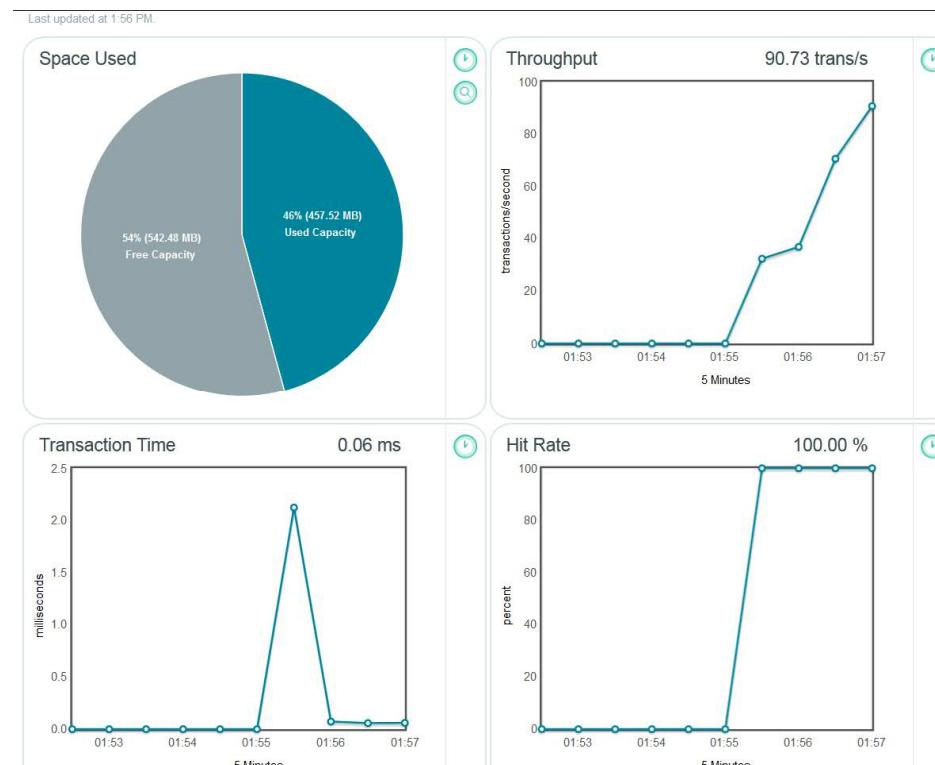
## Using operational services

- Session Cache
- Application Auto-Scaling
- Log Analysis
- Monitoring

## Session Cache

Improve application resiliency by storing session state information across many HTTP requests.

Enable persistence HTTP sessions for your application & seamless session recovery in event of an application failure.



## Liberty server dir/package

- A server package
  - Use “./bin/server package myServer --include=usr” to create it
- Or, simply a server directory



Now you take full control!

You can also turn off Auto-Configuration and get the control

- ▶ Connection info in runtime-vars.xml as variables
- ▶ Provide a server.xml that reads these variables
- ▶ Push server package

**myServerPackage.zip**

```
wlp
  usr
    servers
      defaultServer
        server.xml
      apps
        myapp.war
```

You can also turn off Auto-Configuration and get the control variables available via runtime-vars.xml (provided)

```
<server>
  <variable name='cloud.services.myDB.name' value='myDB' />
  <variable name='cloud.services.myDB.label' value='SQLDB-1.0' />
  <variable name='cloud.services.myDB.plan' value='SQLDB_OpenBeta' />
  <variable name='cloud.services.myDB.connection.hostname'
    value='75.126.155.169' />
  <variable name='cloud.services.myDB.connection.host'
    value='75.126.155.169' />
  <variable name='cloud.services.myDB.connection.port'
    value='50000' />
  <variable name='cloud.services.myDB.connection.username'
    value='u816491' />
  <variable name='cloud.services.myDB.connection.password'
    value='7QquHfNjaF' />
  <variable name='cloud.services.myDB.connection.db'
    value='I_819916' />
  <variable name='cloud.services.myDB.connection.jobcurl'
    value='jdbc:db2://75.126.155.169:50000/I_819916' />
  <variable name='cloud.services.myDB.connection.uri'
    value='db2://u816491:7QquHfNjaF@75.126.155.169:50000/I_819916' />
  <variable name='port' value='61477' />
```

## server.xml (you provide)

```
<dataSource id='db2-myDB' jdbcDriverRef='db2-driver'
  jndiName='jdbc/myDB' statementCacheSize='30' transactional='true'>
  <properties.db2.jcc databaseName='${cloud.services.myDB.connection.db}'
    id='db2-myDB-props' password='${cloud.services.myDB.connection.password}'
    portNumber='${cloud.services.myDB.connection.port}'
    serverName='${cloud.services.myDB.connection.host}' user='${cloud.services.myDB.connection.username}' />
</dataSource>
```

## Deploying Liberty applications to Bluemix

- Server centric paradigm
  - Focus on runtime infrastructure available in the cloud
  - server.xml provided along with the application
  - User understands how to configure Liberty for their application
  - User wants that configuration reflected in the cloud deployment
- Cloud centric paradigm
  - Focus on the application, not on the runtime infrastructure
  - server.xml is generated by the buildpack
  - User does not have to configure the Liberty server instance
  - User expects the environment to just do the right thing in order for their application to run

## Supported Application Types

- cf push - Push a single app (with or without a manifest):
  - ▶ `cf push APP [-b BUILDPACK_NAME] [-c COMMAND] [-d DOMAIN] [-f MANIFEST_PATH] [-i NUM_INSTANCES] [-m MEMORY] [-n HOST] [-p PATH] [-s STACK] [-t TIMEOUT] [--no-hostname] [--no-manifest] [--no-route] [--no-start]`
- Standalone jar files (\*.jar)
- Web Applications (\*.war)
- Enterprise Applications (\*.ear)
- Liberty profile server package (\*.zip)
- Liberty profile server directory (dir.)
- <https://www.ng.bluemix.net/docs/Liberty/LibertyApp.html>

## Support Statement for Java apps deployed to Bluemix

- We will keep two Liberty buildpacks in bluemix
  - Latest and one back
- Buildpacks can be explicitly targeted with the –b option
- Default Liberty Buildpack will always be the latest valid service level
- Cumulative service will only be provided on the latest Buildpack level
- ▶ We will retire any level of the buildpack as we see fit
- Provide scripts & documentation to perform the “blue-green” application update at eGA
- ▶ ***IBM reserves the right to restart and in very rare situations restage applications***  
due to operational processes or maintenance requirements

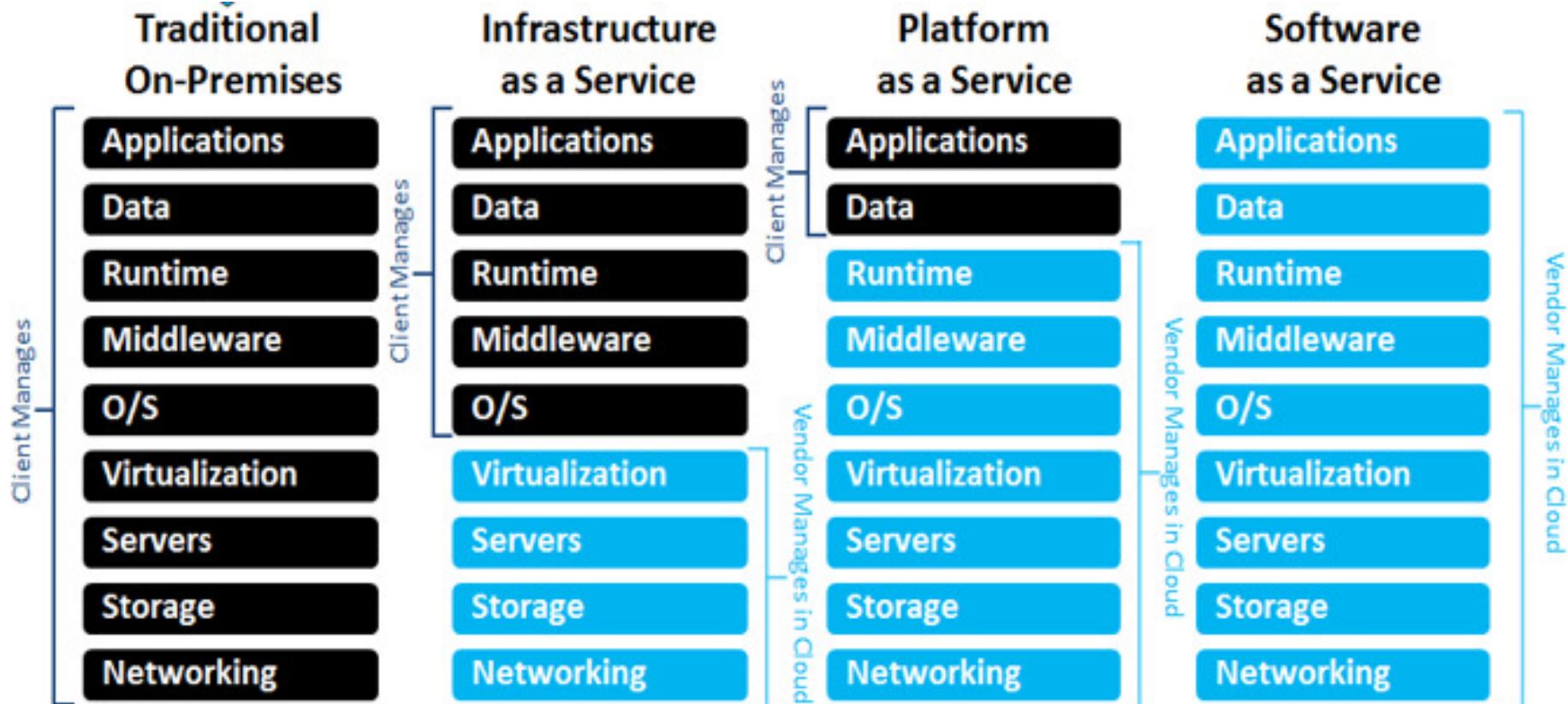
## Debugging and Troubleshooting apps pushed with Liberty buildpack

- cf push with the latest production client explicitly specifying location with the –p
- If you see timeouts during push increase environment variable values
  - CF\_STAGING\_TIMEOUT=15 Max wait time for buildpack staging, in minutes
  - CF\_STARTUP\_TIMEOUT=5 Max wait time for app instance startup, in minutes
- Check to see if you are inadvertently pushing with a manifest file
- Liberty logs can be viewed and downloaded using cf commands
  - cf files, cf logs, cf logs --recent, cf events
- Logs and Files can also be looked at from ACE
  - Liberty logs: messages.log, stderr.log, stdout.log, FFDCs
  - env.log: environment variables of the application process
  - staging\_task.log: staging logs from the app staging process
- Set CF\_TRACE=true and then use application guid to drive cf curl commands
- Bind to the Configuration service to dynamically enable trace
- Set trace specification in server.xml and push the server package again
- Buildpack logging enabled if you push after setting the JBP\_LOG\_LEVEL environment variable in manifest.yml or using the cf set-env command – logs available in the diagnostics directory
- Explicitly set the liberty buildpack start command with –command
- Inspect the VCAP\_SERVICES environment variable and connect directly to the service from the service console



# Cloud Service Models and IBM Cloud Offerings

## Cloud Services Models and IBM Cloud Offerings



Customization; higher cost; slower time to value

Standardization; lower cost; faster time to value

## Traditional On-Premises – Core IT

● Client Manages

### Core IT



### Benefits

- Fully customizable.
- Few limitations.
- Necessary for some solutions.
- Existing Investments.

### Time Commitment

- Weeks to setup and deploy.
- Maintenance/upgrades of hardware and software.

Time to initial deployment



~ Weeks

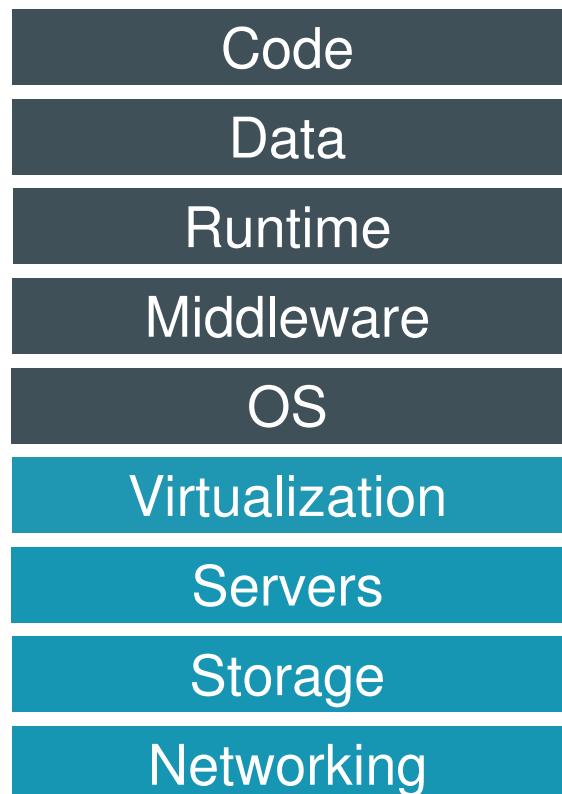
~ Days

~ Minutes

## Infrastructure as a Services (IaaS)

- Client Manages
- Vendor Manages in Cloud

Infrastructure  
as a Service



### Benefits

- Most control in the cloud.
- Necessary for some solutions.
- Infrastructure managed by SP.

### Time Commitment

- Minutes to provision VM.
- Time to configure software and apps varies.
- Maintenance/upgrades of OS, middleware, runtime.

**SOFTLAYER®**  
an IBM Company



## IBM SoftLayer IaaS Offerings



- Highly flexible architecture
- One platform for public cloud servers, private clouds, bare metal servers
- Unified systems management & API
- Technology-neutral platform
- Support for broad range of operating systems, virtualization platforms

Infrastructure Management System provides orchestration and automation

Hyper-V

VMWare

Citrix Xen

OpenStack

CloudStack

Parallels



Bare Metal Servers



Virtual Server Instances

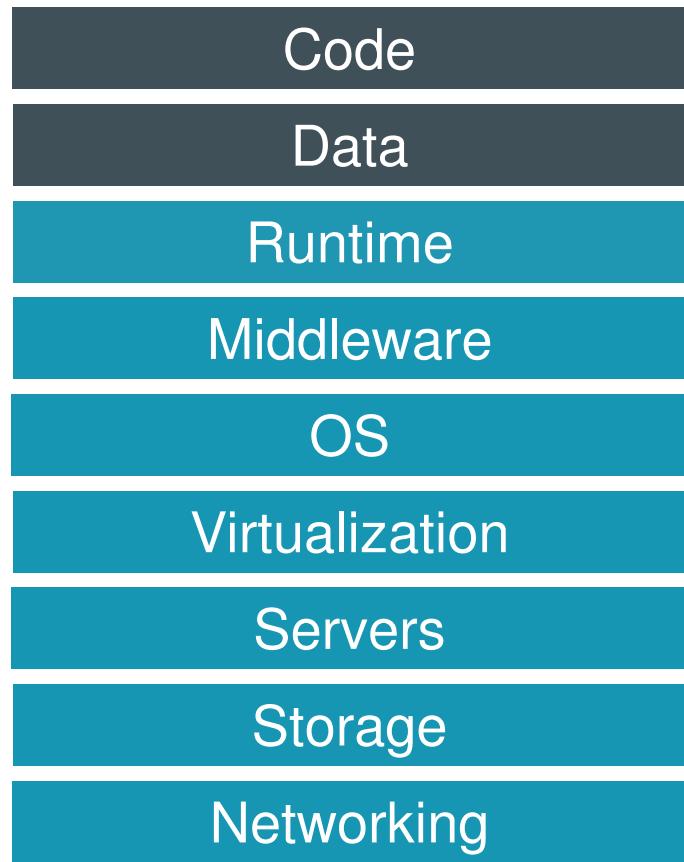


Private Clouds

## Platform as a Services (PaaS)

- Client Manages
- Vendor Manages in Cloud

### Platform as a Service



### Benefits

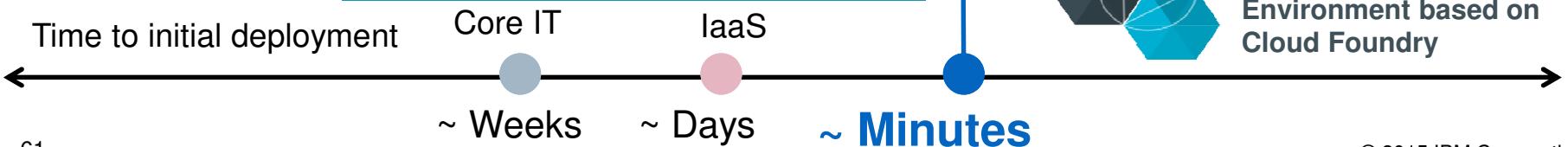
- Setup environments and deploy apps *very quickly*.
- Infrastructure and platform managed by Service Provider.

### Time Commitment

- Minutes to setup and deploy.
- Focus on your apps and their data.



IBM Bluemix PaaS  
Environment based on  
Cloud Foundry



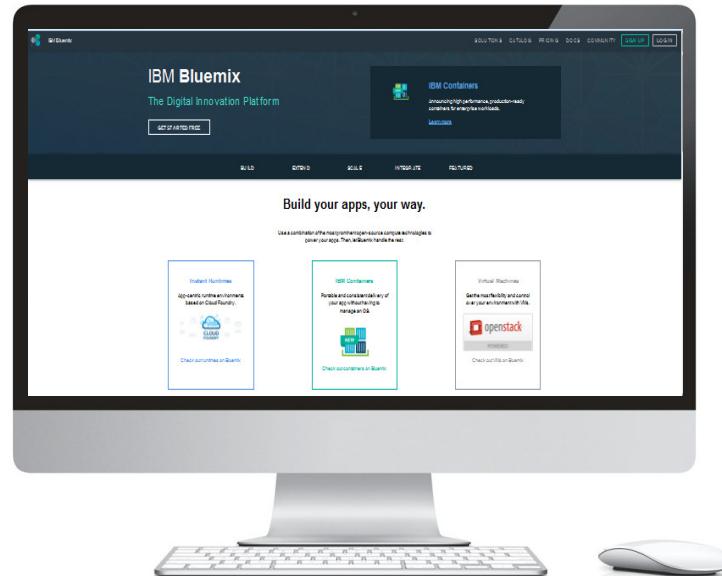
## IBM Platform as a Service Offerings – Bluemix

### Service catalog available on Bluemix

- **DevOps** - From development to deployment
- **Big Data** – Harness the power
- **Mobile** – Quickly get started with your next app
- **Watson** – Enable building cognitive apps
- **Business Analytics** – Powerful analytics made easy
- **Data Management** – Agile data mgmt. & refinement
- **Web and Application** – Deliver new web & mobile apps
- **Security** – Build security into application design
- **Internet of Things** – A new generation of apps
- **Integration** – Extend existing investments & Infrastructure

### Developer experience

- Rapidly deploy and scale applications in any language.
- Compose applications quickly with useful APIs and services and avoid tedious backend config.
- Realize fast time-to-value with simplicity, flexibility and clear documentation.



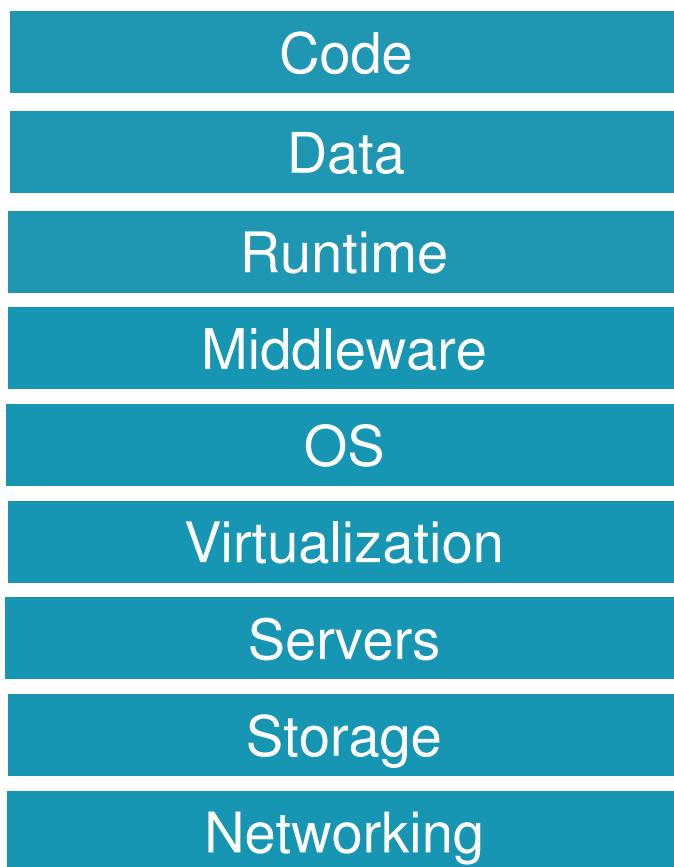
Built on a foundation of **open** technology.

### Enterprise capability

- Securely integrate with existing on-prem data and systems.
- Choose from flexible deployment models.
- Manage the full application lifecycle with DevOps.
- Develop and deploy on a platform built on a foundation of open technology.

## Software as a Services (SaaS)

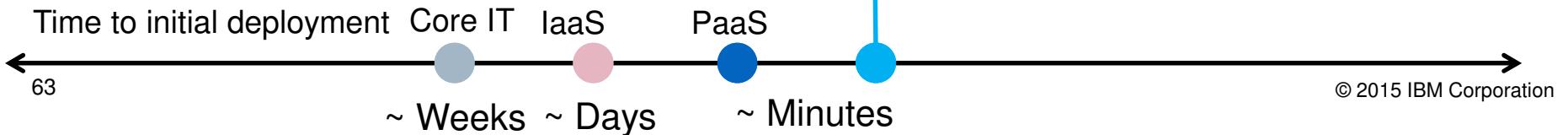
- Vendor Manages in Cloud



### Software as a Service

#### Benefits

- Sign up and rapidly start using business applications.
- Access apps and data from any connected end points
- No data is lost if computer breaks
- The service is able to dynamically scale to usage needs
- No longer have to install, update and maintain the software



## IBM Cloud SaaS Portfolio

### Software as a Service



IBM Cloud for Smarter Process



IBM Cloud  
for Smarter  
Commerce



IBM Cloud  
for Smarter  
Cities



IBM Cloud  
for Smarter  
Workforce



IBM Cloud for Collaboration



IBM Cloud for Big Data & Analytics



IBM Cloud for IT Operations

## IBM Cloud Marketplace – Cloud Services Store

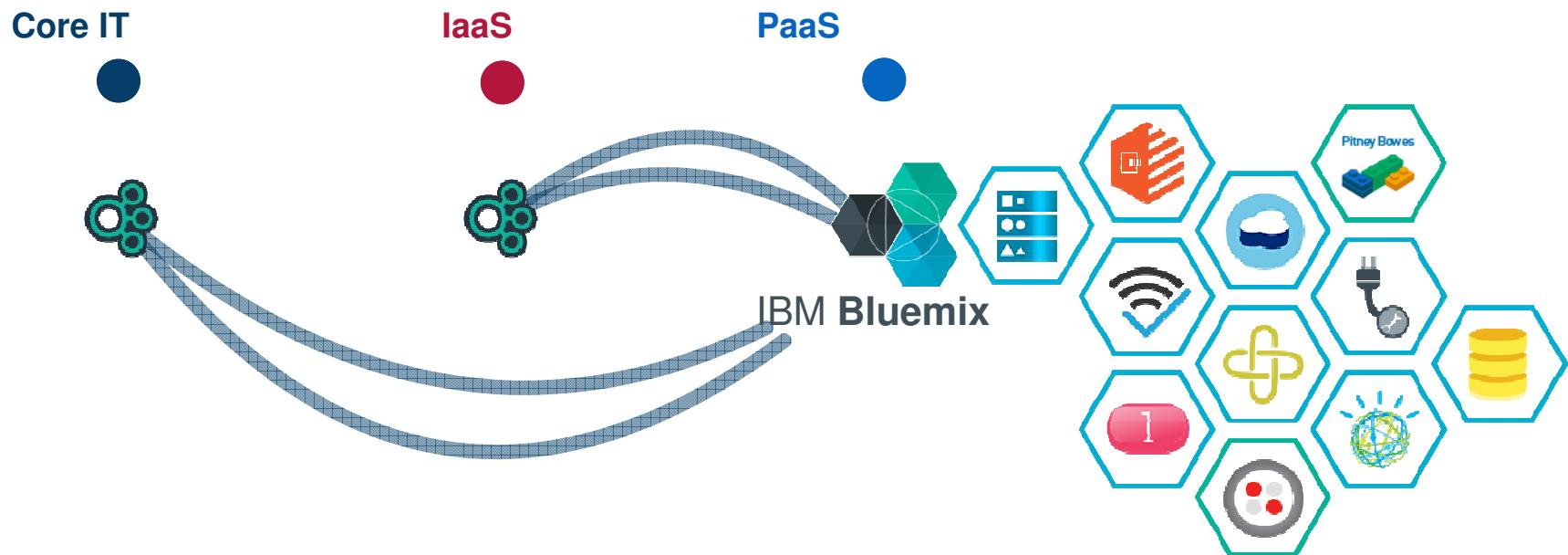
Data Stores	Development Tools	Security	Operations Support
    <b>AppFirst™</b> 	       	     	      
Messaging	Mobile	Analytics	Business Support

- Is a dedicated area on [ibm.com/cloud](http://ibm.com/cloud) for IBM customers to discover enterprise cloud services that run on or are deployable on SoftLayer or integrated with Bluemix and are available from IBM & Business Partners.

[ibm.biz/newway4partners](http://ibm.biz/newway4partners)

Timing is Critical ...so are all of your other investments

Leverage the power of Bluemix **without abandoning** what you already use.



## Bookmarks & References

<http://www.ibm.com/cloud-computing/us/en/>

<http://www.ibm.com/cloud-computing/us/en/iaas.html>

<http://www.ibm.com/cloud-computing/us/en/paas.html>

<http://www.ibm.com/cloud-computing/us/en/saas.html>

<http://www.ibm.com/developerworks/cloud/library/cl-open-architecture/>

<http://www.ibm.com/cloud-computing/us/en/open-cloud-essentials.html>

<https://www.bluemix.net>

<https://www.ng.bluemix.net/docs/#starters/index-gentopic4.html#genTopProId5>

[https://www.ng.bluemix.net/docs/#starters/index-gentopic3.html#container\\_overview](https://www.ng.bluemix.net/docs/#starters/index-gentopic3.html#container_overview)

<https://www.docker.com/>

## QUESTIONS?

[HTTPS://IBM.BIZ/BDXGB1](https://ibm.biz/bdxgb1)

Get help from Bluemix  
development team and Stack  
Overflow community