Pivotal

BUILT FOR THE SPEED OF BUSINESS

Part 1: Spring Update

Paul Chapman
Senior Consultant, Pivotal

pchapman@gopivotal.com

GitHub: paulc4 – these slides and demos

Pivotal

© Copyright 2013 Pivotal. All rights reserved.

Spring Update

- What is Pivotal?
- Spring V4
- Spring IO
  - Introduction to selected sub-projects
  - Spring Data
  - Spring Integration
  - Spring Data
  - Spring Batch, Spring XD

© Copyright 2013 Pivotal. All rights reserved.

Spring Update

- **What is Pivotal?**
- Spring V4
- Spring IO
  - Introduction to selected sub-projects
  - Spring Data
  - Spring Integration
  - Spring Data
  - Spring Batch, Spring XD

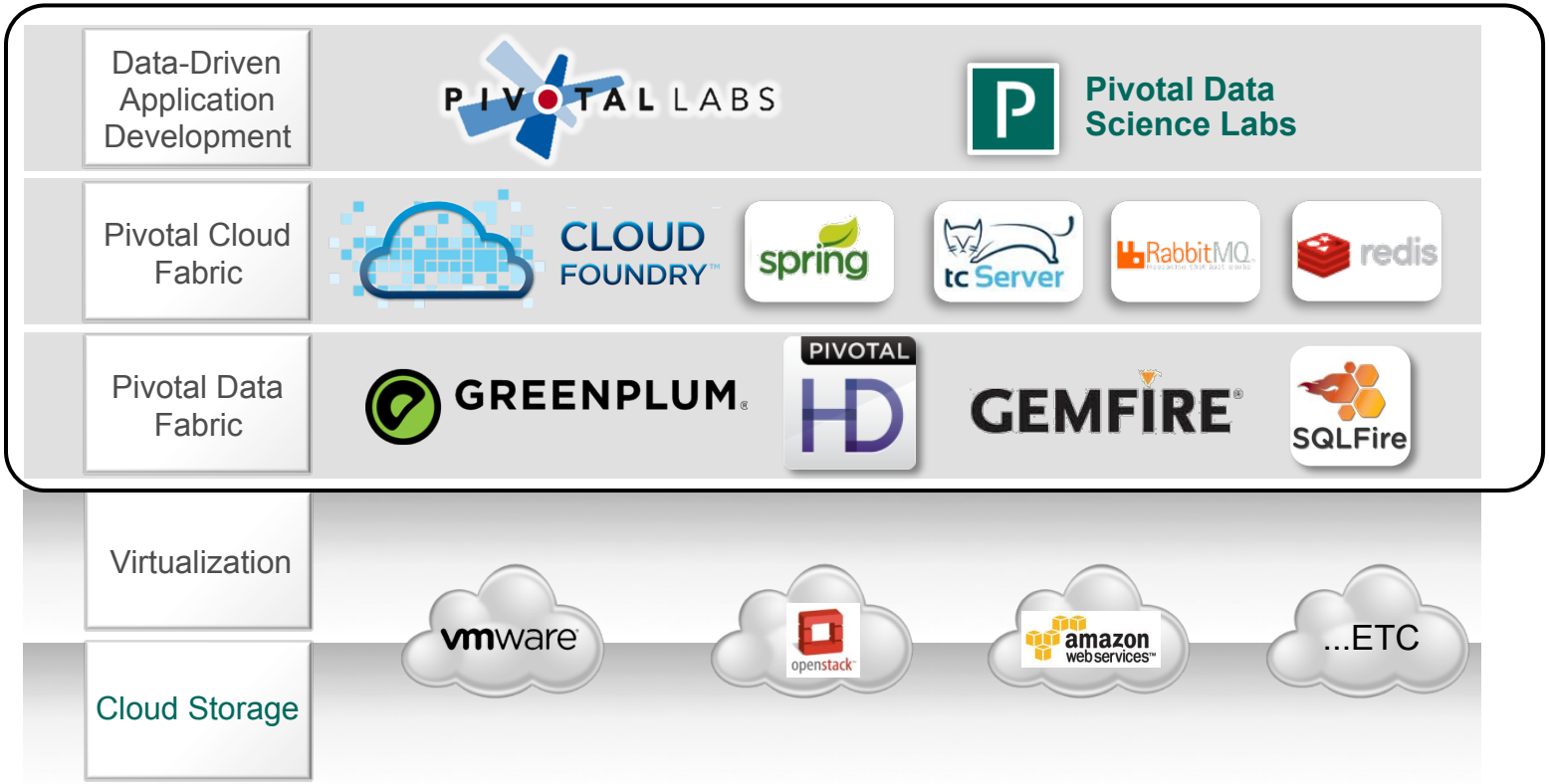
© Copyright 2013 Pivotal. All rights reserved.

Pivotal – the Company

- The new home for several products from VMware and EMC
  - Joint venture: VMware, EMC, GE
- Assets
  - Open-source products including Spring, Rabbit MQ, Redis
  - Data-management tools including Greenplum “sharded” RDB, Pivotal HD (was GPHD), GemFire, SQLFire, Cetas
  - Cloud Foundry
  - Pivotal Labs – agile development specialists

© Copyright 2013 Pivotal. All rights reserved.

Integrating EMC and VMware Assets



© Copyright 2013 Pivotal. All rights reserved.

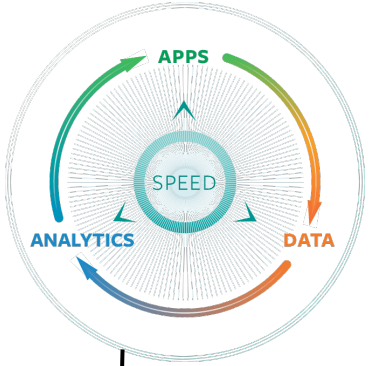
Pivotal – the Platform

- Pivotal CF (aka Pivotal One)
  - Open PAAS: run internally *and/or* on public cloud providers
- Why?
  - Makes available to *any company* the type of environment Internet companies like Google, Amazon are using
  - Virtualized Infrastructure
  - Flexible, dynamic, on-demand cloud deployment
  - Sophisticated data-analytics
    - Spot market trends, real-time system performance

© Copyright 2013 Pivotal. All rights reserved.

Popular Use-Cases

mobile/social refresh

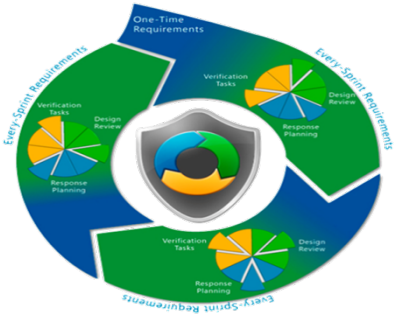


data analytics

dev-ops: dev, test, trial

app modernization

CLOUD FOUNDRY™



agile transformation

© Copyright 2013 Pivotal. All rights reserved.

Spring Update

- What is Pivotal?
- **Spring V4**
- Spring IO
  - Introduction to selected sub-projects
  - Spring Data
  - Spring Integration
  - Spring Data
  - Spring Batch, Spring XD

© Copyright 2013 Pivotal. All rights reserved.

Spring – V4

- Due end 2013 – but depends on Java 8
  - Java 8 delayed to Q1 2014
  - Byte-codes currently incompatible with ASM
- Highlights
  - Support for Java 8 (Java 6 minimum)
  - Core messaging package (from Spring Integration)
  - @Conditional, @Lazy injection
  - Web-sockets and `AsyncRestTemplate`
  - JMS 2.0, JTA 1.2, JPA 2.1, Servlet 3.0 (2.5 supported) …

© Copyright 2013 Pivotal. All rights reserved.

Java 8 – Closures (Lamdas)

- Can pass code around like data
  - Ideal for Spring template callbacks

```
JdbcTemplate jt = new JdbcTemplate(dataSource);

jt.query("SELECT name, age FROM person WHERE dep = ?", new RowMapper<Person>() {
    @Override
    public Person mapRow(ResultSet rs, int rows) throws SQLException {
        return new Person(rs.getString(1), rs.getInt(2));
    }
}, 112);

jt.query("SELECT name, age FROM person WHERE dep = ?",
        (rs, rowNum) -> new Person(rs.getString(1), rs.getInt(2)),
        112);
```

Old

New

© Copyright 2013 Pivotal. All rights reserved.

Spring Update

- What is Pivotal?
- Spring V4
- **Spring IO**
  - Introduction to selected sub-projects
  - Spring Data
  - Spring Integration
  - Spring Data
  - Spring Batch, Spring XD

© Copyright 2013 Pivotal. All rights reserved.

Spring IO

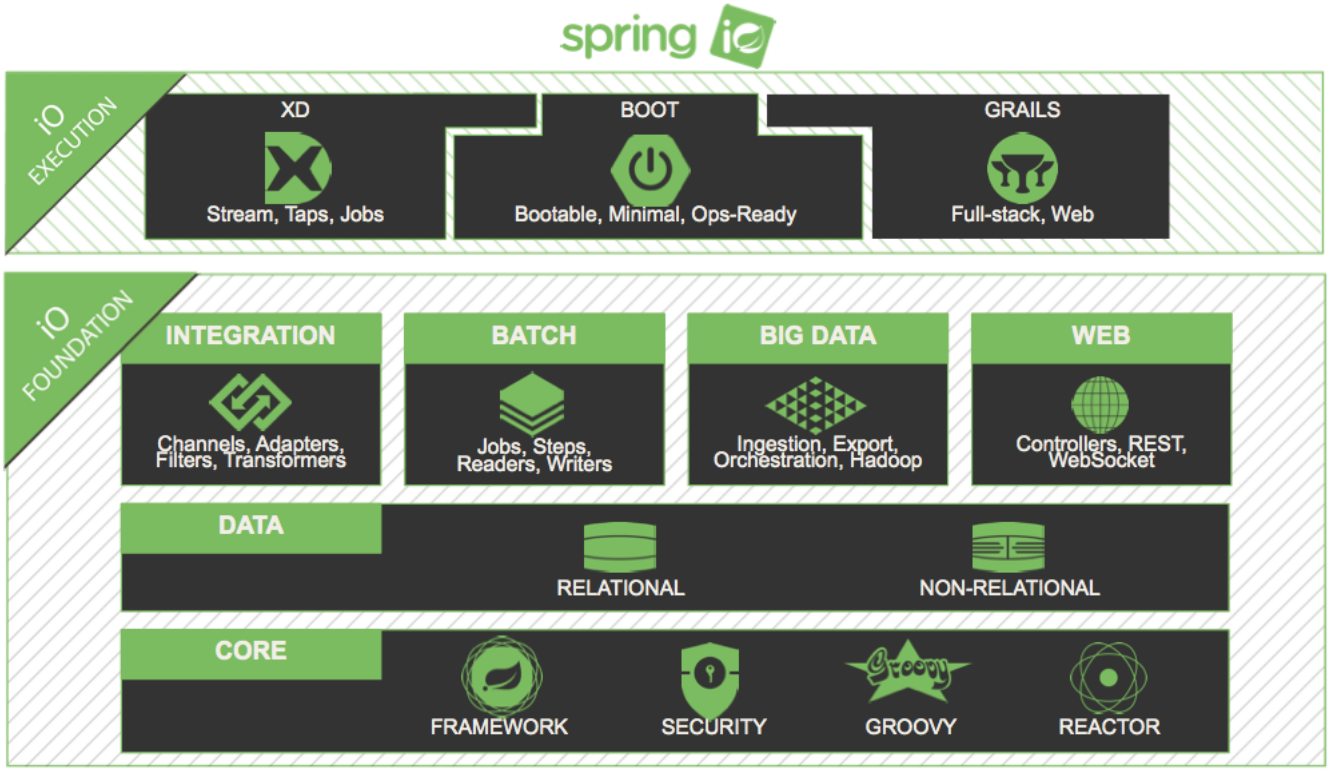
© Copyright 2013 Pivotal. All rights reserved.

Spring IO

- Spring is no longer just a framework, it is now a *platform*
  - *Brings together the Spring family of projects into a cohesive and versioned foundational platform for modern applications*
  - Co-ordinated release trains
- New web-site: spring.io
  - All Spring's old web-sites redirect here
  - Incorporates product pages, blog and forums
  - Plus new *HOWTO* guides

© Copyright 2013 Pivotal. All rights reserved.

Spring IO Platform



© Copyright 2013 Pivotal. All rights reserved.

Spring IO Foundation – 1

- Projects you probably already know:


SPRING FRAMEWORK


SPRING WEB FLOW


SPRING WEB SERVICES


SPRING INTEGRATION

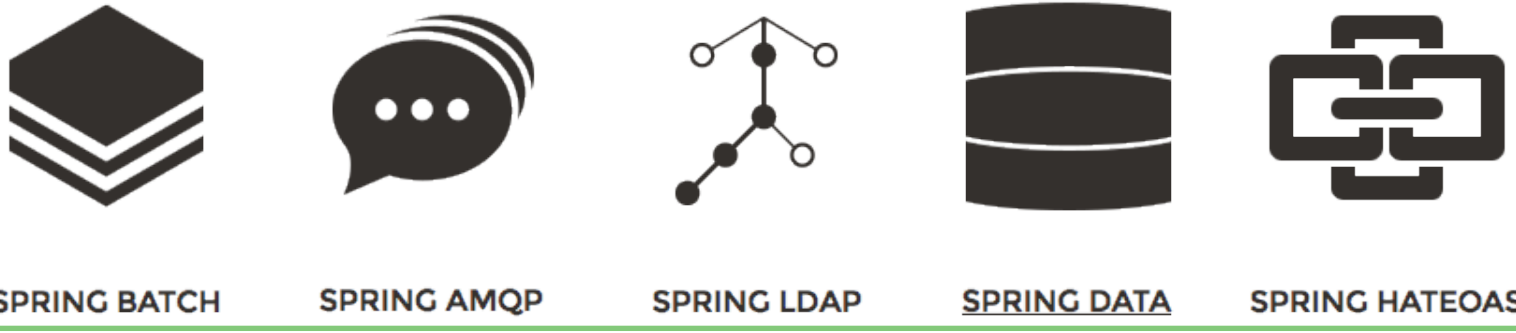

SPRING SECURITY

SPRING WEB – really just a big, optional sub-module of Framework

© Copyright 2013 Pivotal. All rights reserved.

Spring IO Foundation – 2

- Projects you may not know so well



© Copyright 2013 Pivotal. All rights reserved.

Spring Batch

- Run jobs to a schedule
  - Ideal for downsizing from mainframe to mid-range
  - Equivalent to mainframe job-control language (JCL)
  - Easier for Java developers than shell scripts
  - Run multiple processes (steps) in sequence, in parallel
  - Recover after failed step(s)
  - Traditional ELT: Export Transform Load
  - Often used to pull data into specialist databases
    - Data Warehouse, Hadoop

© Copyright 2013 Pivotal. All rights reserved.

Spring AMQP

- Advanced Message Queuing Protocol
  - Common protocol for messaging
    - Like SMTP for email
  - Several products: including Pivotal's open-source Rabbit/MQ
  - Fast, light, interoperable
  - How do I get data in/out of my cloud(s)?
    - Without paying huge licensing fees
    - Need a common messaging infrastructure
  - `AmqpTemplate`: similar to `JmsTemplate`
  - `<rabbit:listener-container>` similar to JMS

© Copyright 2013 Pivotal. All rights reserved.

Spring LDAP

- Lightweight Directory Access Protocol
  - Both a protocol and an old-style hierarchical database
- Project Features
  - LDAP template (like `JdbcTemplate`) hides LDAP API
  - Throws Spring's unchecked `DataAccessExceptions`
  - Classes for dynamically building LDAP filters and Distinguished Names (DNs)
  - Client-side LDAP transaction management (LDAP has none)

```
dn: cn=chapman, ou=employees, dc=pivotal, dc=com
```


© Copyright 2013 Pivotal. All rights reserved.

Spring Data

SPRING DATA

- Support for multiple no-SQL databases
  - JPA, Mongo/DB, Redis, Gemfire, Neo4j, Hadoop, JDBC Extensions
  - Template classes for database access (like `JdbcTemplate`)
  - Annotations for automated persistence (like JPA `@Entity`)
  - Spring name-spaces for configuration

© Copyright 2013 Pivotal. All rights reserved.

Spring Data – Instant Repositories

- Just one feature of Spring Data
- How?
  - Annotate domain classes suitably to define keys and enable persistence
  - Define your repository as an *interface*
- Spring will implement it at run-time
  - Scans for interfaces extending Repository <T, K>

© Copyright 2013 Pivotal. All rights reserved.

Spring Data – Mark-up Classes

- You do this for JPA anyway
  - Spring Data defines similar annotations for other storage products

```
@Entity @Table(...)
public class Customer {

  @Id
  @GeneratedValue(strategy = GenerationType.AUTO)
  private Long id;

  private Date firstOrderDate;
  private String email;

  // Other data-members
}
```

JPA – map to a Table

```
@Document
public class Customer {
   ...
```

MongoDB – map to a JSON document

```
@Region
public class Customer {
   ...
```

Gemfire – map to a region

© Copyright 2013 Pivotal. All rights reserved.

Spring Data – Extend Predefined Interfaces

```
public interface CrudRepository<T, ID
        extends Serializable> extends Repository<T, ID> {

    public <S extends T> save(S entity);
    public <S extends T> Iterable<S> save(Iterable<S> entities);

    public T findOne(ID id);
    public Iterable<T> findAll();

    public void delete(ID id);
    public void delete(T entity);
    public void deleteAll();
}
```

Marker interface – add any methods from CrudRepository or finders

```
public interface Repository<Customer, Long> {  }
```


© Copyright 2013 Pivotal. All rights reserved.

Spring Data – Add Finders

- Auto-generated finders obey naming convention
  - findBy<*DataMember*><*Op*> : <Op> is Gt, Lt, Ne, Between, Like …

```
public interface CustomerRepository
       extends CrudRepository<Customer, Long> {

    public Customer findByEmail(String someEmail);     // No <Op> for Equals
    public Customer findByFirstOrderDateGt(Date someDate);
    public Customer findByFirstOrderDateBetween(Date d1, Date d2);

    @Query("SELECT c FROM Customer c WHERE c.email NOT LIKE '%@%'")
    public List<Customer> findInvalidEmails();
}
```

ID

Custom query uses query-language of underlying product (here JPQL)

© Copyright 2013 Pivotal. All rights reserved.

Spring Data – Instant Repositories – 1

- Get Spring to scan for interfaces
  - Implemented on-the-fly

```
<jpa:repositories base-package="com.acme.**.repository" />

<mongo:repositories base-package="com.acme.**.repository" />

<gfe:repositories base-package="com.acme.**.repository" />
```

- Now just inject a dependency of type `CustomerRepository`

```
@Autowired CustomerRespository custRepo;
```


© Copyright 2013 Pivotal. All rights reserved.

Spring Hateoas

SPRING HATEOAS

- HTTP As The Engine of Application State
  - Possibly the world's *worst* acronym
  - RESTful responses should contain links
    - Like HTML web-pages do
    - Avoids the need for WSDL (SOAP)
  - Spring Hateoas uses HAL (Hypertext Application Language) to generate links in XML or JSON
    - HAL is not yet a standard, just one possible option

See guide for example: https://spring.io/guides/gs/rest-hateoas

© Copyright 2013 Pivotal. All rights reserved.

Spring Hateoas – JSON Example

```
{
  "_links": {
      "self": { "href": "/orders" },
      "curies": [{ "name": "ea",
           "href": "http://example.com/docs/rels/{rel}", "templated": true }],
      "next": { "href": "/orders?page=2" },
      "ea:find": { "href": "/orders{?id}", "templated": true },
      "ea:admin": [{
          "href": "/admins/2",
          "title": "Fred"
      }, {
          "href": "/admins/5",
          "title": "Kate"
      }]
  },
```

Spring Hateaos provides an API for generating these links in MVC Controller responses

© Copyright 2013 Pivotal. All rights reserved.

Spring Hateoas – JSON Example

```
    "currentlyProcessing": 14,
    "shippedToday": 20,
    "_embedded": {
        "ea:order": [{
            "_links": {
                "self": { "href": "/orders/123" },
                "ea:basket": { "href": "/baskets/98712" },
                "ea:customer": { "href": "/customers/7809" }
            },
            "total": 30.00,
            "currency": "USD",
            "status": "shipped"
        } ... ]
    }
}
```


© Copyright 2013 Pivotal. All rights reserved.

Spring IO Foundation – 3

- More projects you may not know so well

SPRING MOBILE

SPRING SOCIAL

SPRING FOR ANDROID

© Copyright 2013 Pivotal. All rights reserved.

Spring Mobile

- Enables device detection
  - Add `DeviceResolverHandlerInterceptor` filter
- Site preferences
  - `SitePreferenceHandlerInterceptor`
  - `SiteSwitcherHandlerInterceptor`
- Supports different views per device
  - Prepend device type to view name

SPRING MOBILE

© Copyright 2013 Pivotal. All rights reserved.

Spring Mobile – Examples

```
@RequestMapping("/")
public void home(Device device) {
    if (device.isMobile()) {
        // Do mobile stuff
    } else if (device.isTablet()) {
        // Do tablet stuff
    } else {
        // Do big-device stuff

    }

    return "index";    // Index view
}
```

```
<bean name="irvr" class="...InternalResourceViewResolver">
    <property name="prefix" value="WEB-INF/views"/>
    <property name="suffix" value="jsp"/>
</bean>

<bean class="...LiteDeviceDelegatingViewResolver">
    <constructor-arg ref="irvr"/>
    <property name="mobilePrefix" value="mobile/"/>
    <property name="tabletPrefix" value="tablet/"/>
    <!-- No prefix for big devices - render usual JSP -->
</bean>
```

Mobile request:

do mobile stuff executed then *WEB-INF/views/mobile/index.jsp* rendered

© Copyright 2013 Pivotal. All rights reserved.

Spring Social

- Integrate you application with social media
  - Core
    - Handle user-accounts
    - Login using Spring Security OAuth
  - Sub-projects for Facebook, Twitter, LinkedIn
    - Allow access to Facebook graph API
    - Access REST APIs from Twitter and LinkedIn

SPRING SOCIAL

© Copyright 2013 Pivotal. All rights reserved.

Spring Android

- Provides components of Spring for use in Android apps
- Features
  - REST client for Android
  - OAuth support for accessing secure APIs

See: https://spring.io/guides/gs/consuming-rest-android/

SPRING FOR ANDROID

© Copyright 2013 Pivotal. All rights reserved.

Spring IO Execution

- Build on the foundations

SPRING BOOT

SPRING XD

© Copyright 2013 Pivotal. All rights reserved.

Spring Boot

- Write a deployable Spring application in just a few lines
  - Many defaults out of the box
  - Override any/all as needed
  - Still need to setup your build environment (ant, maven, gradle)
  - Can run as a Java application
    - Deploys to embedded container (Tomcat by default)
  - Or deploy as a war in usual way
  - Java Configuration or XML supported plus autowiring and component-scanning

SPRING BOOT

© Copyright 2013 Pivotal. All rights reserved.

Simplest Program (Standalone)

- Just one class – an MVC Controller

```
@Controller
@EnableAutoConfiguration
public class SampleController {

  @RequestMapping("/spring-demo")
  @ResponseBody
  String home() {
    return "<body><h1>Hello World!</h1><p>Instant web-app</p></body>";
  }

  public static void main(String[] args)    throws Exception {
    SpringApplication.run(SampleController.class, args);
  }
}
```

This annotation tells Spring Boot to use all its defaults

Initial class to run

Run as Java application and got to http://localhost:8080/spring-demo

© Copyright 2013 Pivotal. All rights reserved.

Or .. to run as a WAR

- Extend `SpringBootServletInitializer`
  - Use component scanning or XML configuration to define Controllers, Services etc

```
@EnableAutoConfiguration
@ComponentScan({ "org.project.myapp" })
@ImportResource(value = "classpath:config/security.xml")
public class Main extends SpringBootServletInitializer {

  @Override
  protected Class<?>[] getConfigClasses() {
    return new Class<?>[] { Main.class };
  }
}
```

Tells Spring Boot to run up as a war

Initial class to run (itself)

© Copyright 2013 Pivotal. All rights reserved.

Spring XD

- A unified, distributed, and extensible system for data ingestion, real time analytics, batch processing, and data export.
  - Extends Spring Batch and Integration for big-data applications
  - Crossover into Pivotal Hadoop and Pivotal data-fabrics

SPRING XD

© Copyright 2013 Pivotal. All rights reserved.

Part 2: Gemfire & Cloud Foundry

Pas Apicella
Senior Field Engineer, Pivotal

papicella@gopivotal.com

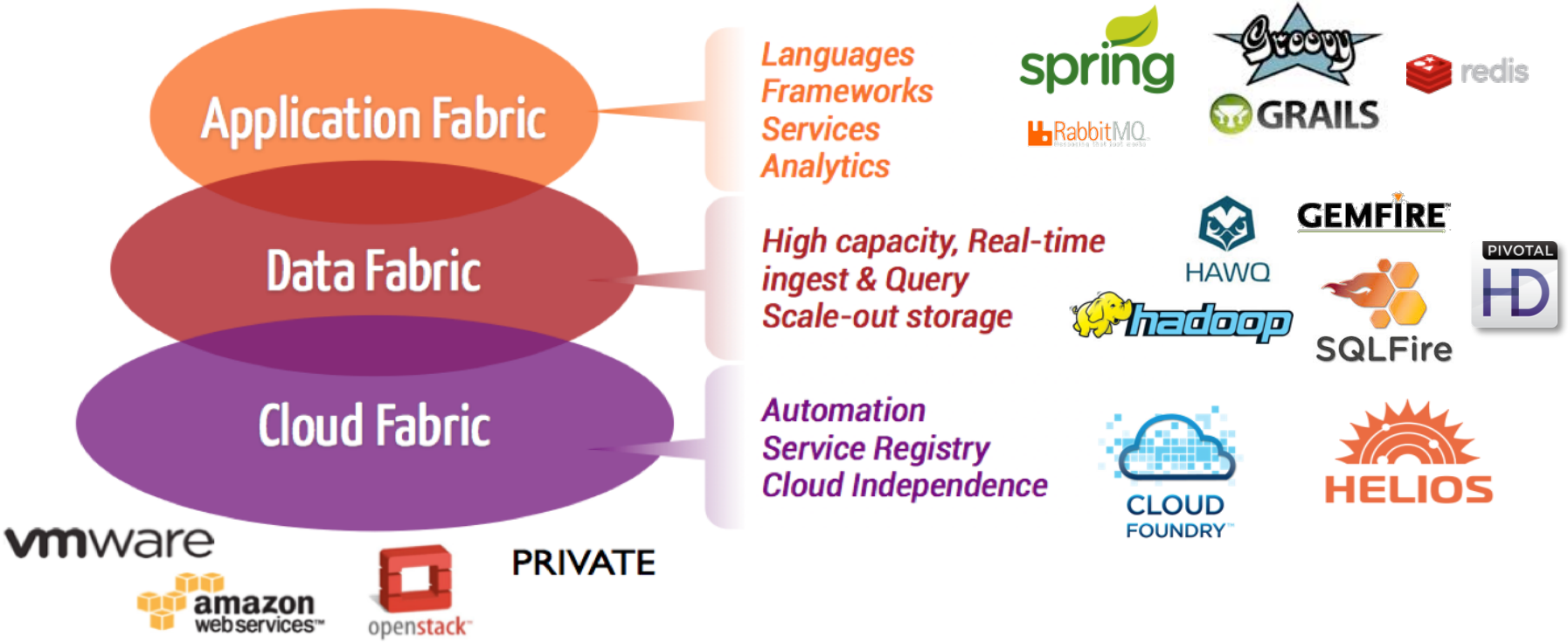
© Copyright 2013 Pivotal. All rights reserved.

Let’s Get Pivotal

- **Pivotal Stack**
- CloudFoundry Open PAAS
- GemFire Data Fabric
- Pivotal Helios

© Copyright 2013 Pivotal. All rights reserved.

Pivotal Stack



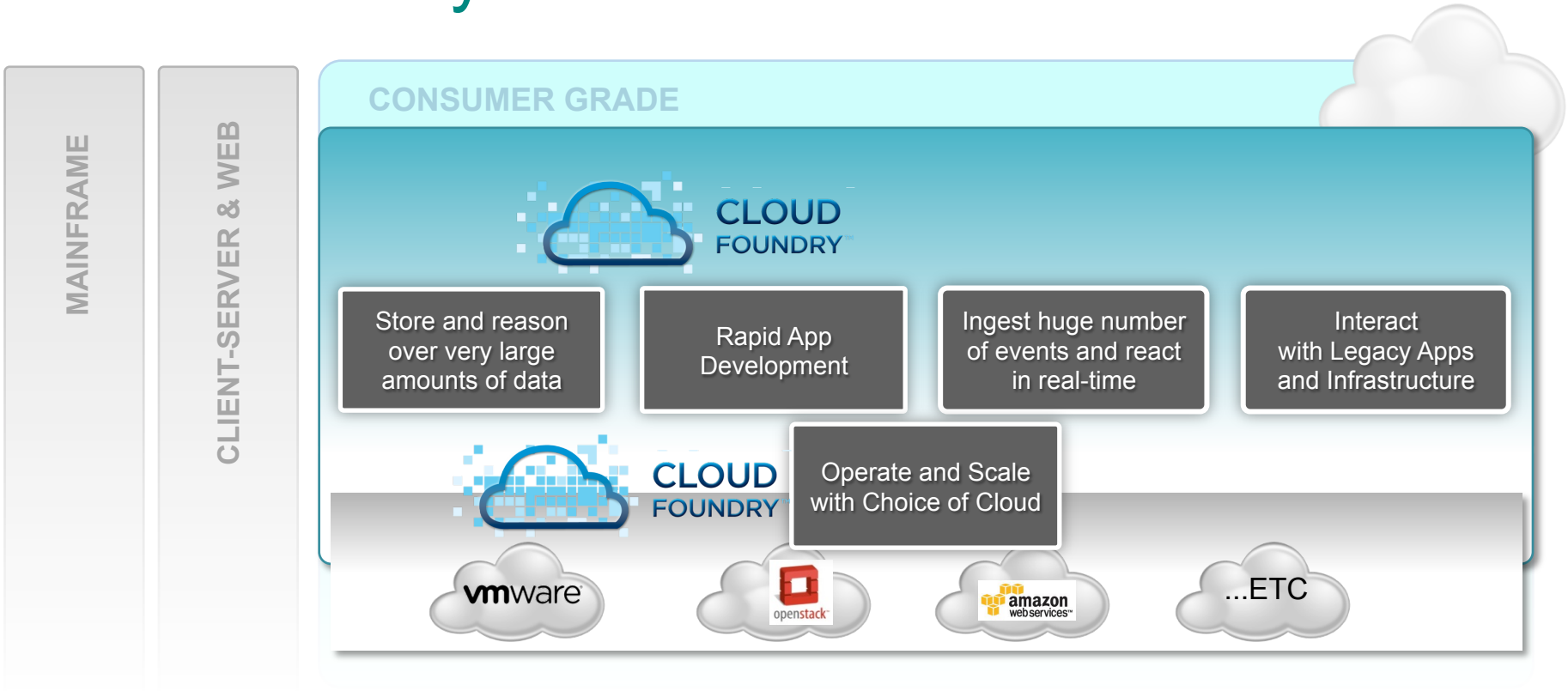
© Copyright 2013 Pivotal. All rights reserved.

Let’s Get Pivotal

- Pivotal Stack
- **CloudFoundry Open PAAS**
- GemFire Data Fabric
- Pivotal Helios

© Copyright 2013 Pivotal. All rights reserved.

Cloud Foundry and Pivotal CF

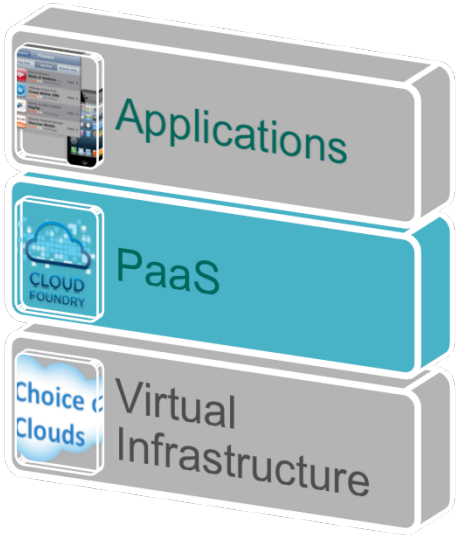


© Copyright 2013 Pivotal. All rights reserved.

Cloud Foundry Platform as a Service (aka PaaS)

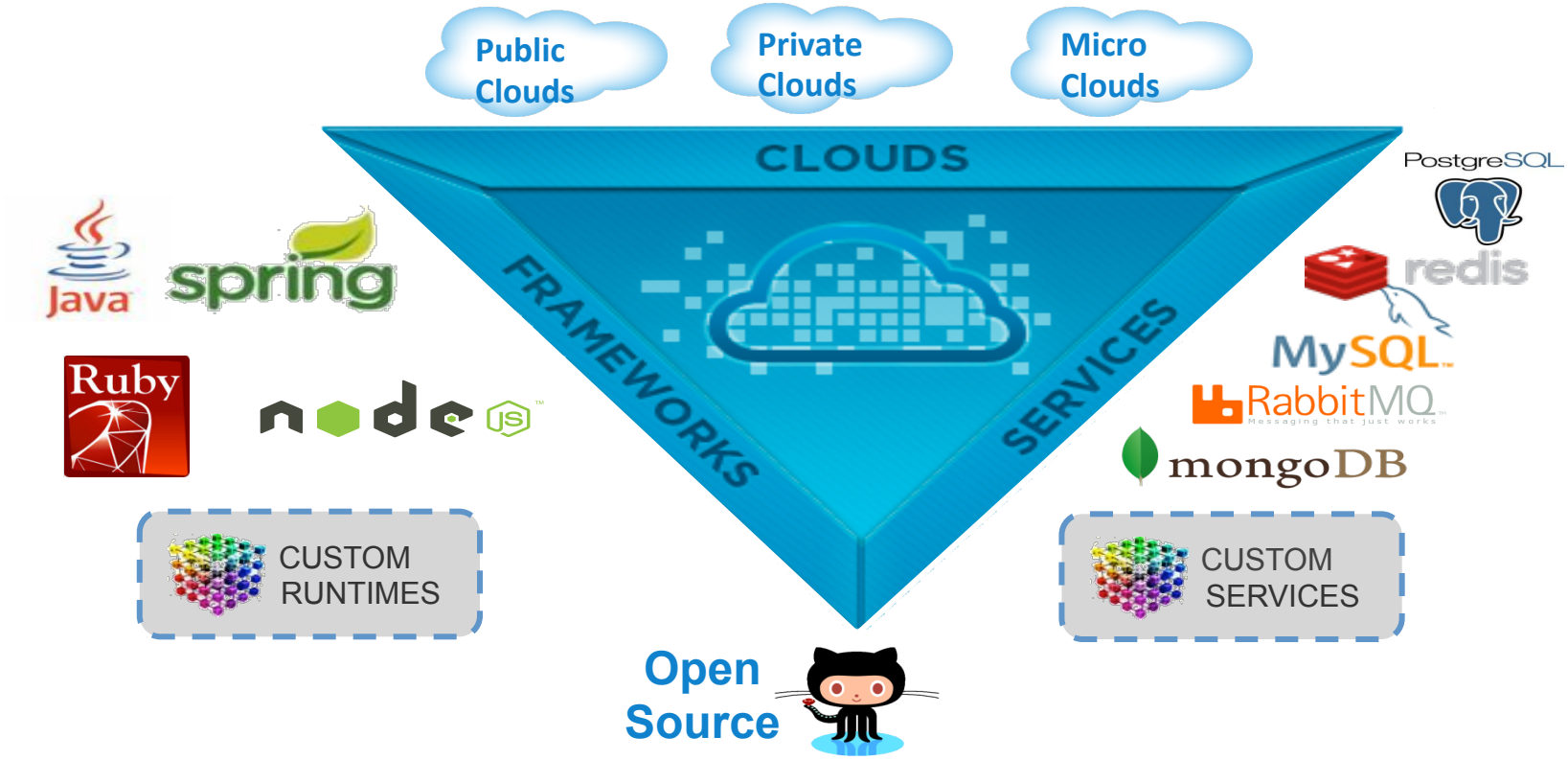
Application is the new unit of deployment and control

- Abstracting VMs and Middleware
- Abstracting containers and processes
- Data as a service
- Eliminate bottleneck of provisioning & deployment



© Copyright 2013 Pivotal. All rights reserved.

What does “Open” PaaS means? Choice!



© Copyright 2013 Pivotal. All rights reserved.

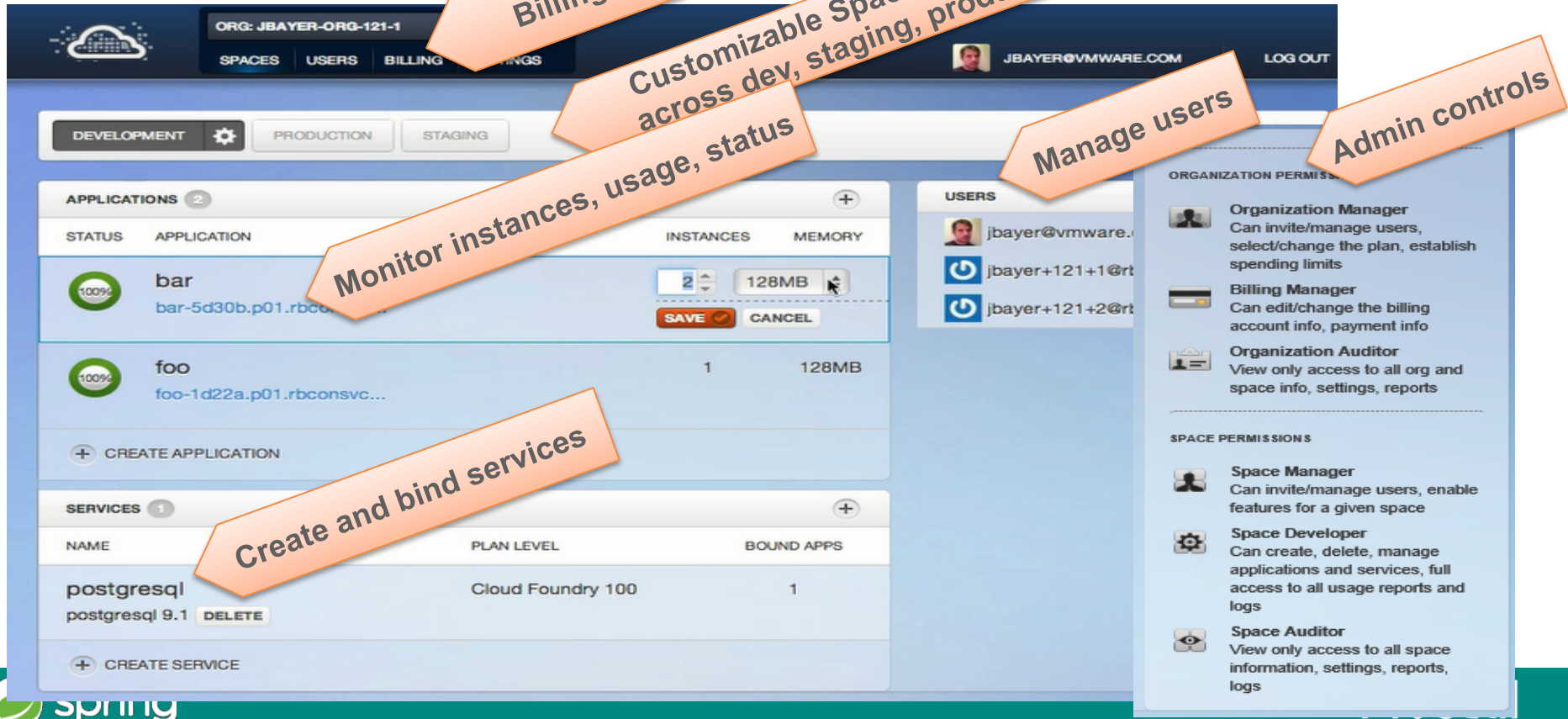
Provision and Deploy in *Less than a Day*!

```
provision <my cloud>
   target <my cloud>
   push <my app>
   bind <my services>
   scale <my app> +100
upgrade <my cloud>
```

Operator
Developer

© Copyright 2013 Pivotal. All rights reserved.

Cloud Foundry Web Console



© Copyright 2013 Pivotal. All rights reserved.

Let’s Get Pivotal

- Pivotal Stack
- CloudFoundry Open PAAS
- **GemFire Data Fabric**
- Pivotal Helios

© Copyright 2013 Pivotal. All rights reserved.

Pivotal GemFire & SQLFire

Extreme Data Scalability & Elasticity

Application Data *Lives* Here

Application Data *Sleeps* Here

Greenplum

File Systems

Databases

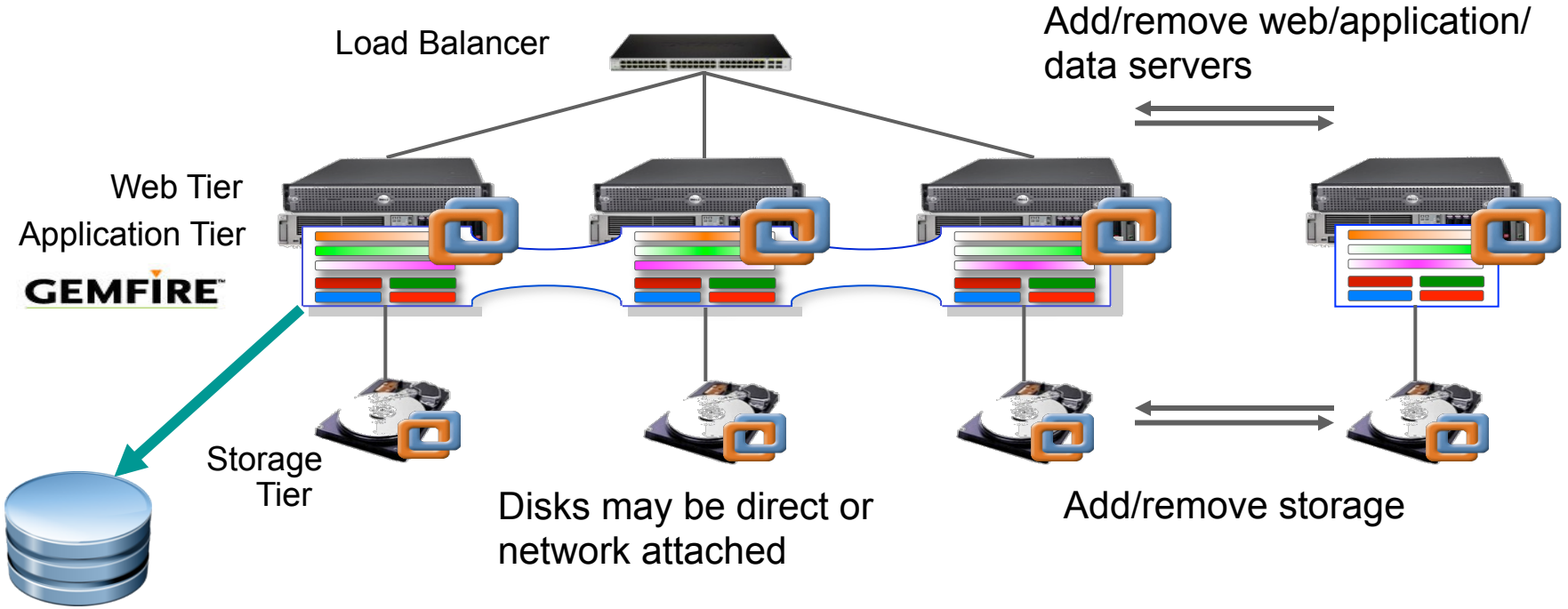
Mainframes / other

Primary Use Cases

- **Web Session Cache, L2 Cache**
  - Shopping Cart state management
- **App Data Cache, In-memory DB**
  - High performance OLTP
- **Grid Data Fabric**
  - Shared data grid accessed by many clients executing app logic
  - … or where app logic is executed within the data fabric itself
  - Synchronise data *across remote* data centres in *near* real time

© Copyright 2013 Pivotal. All rights reserved.

Moving Data to the Middle Tier



Optional reliable, asynchronous feed to Data Warehouse or archival Database

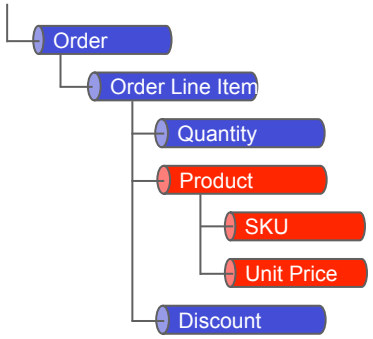
© Copyright 2013 Pivotal. All rights reserved.

Pivotal Gemfire Interfaces

The GEMFIRE family of products support two data interfaces:

GemFire Enterprise

- Key-Value store with OQL Queries
- Stores Objects (Java, C++, C#, .NET) or unstructured data
- Spring-GemFire sub-project
- L2 Cache plugin for Hibernate
- HTTP Session replication module



Name	Thread pitch (mm)	Minor diameter tolerance	Nominal diameter (mm)	Head shape	Price for 50 screws	Available at factory outlet?	Number in stock	Flat or Phillips head?
M4	0.7	4g	4	Pan	$10.08	Yes	276	Flat
M5	0.8	4g	5	Round	$13.89	Yes	183	Both
M6	1	5g	6	Button	$10.42	Yes	1043	Flat
M8	1.25	5g	8	Pan	$11.98	No	298	Phillips
M10	1.5	6g	10	Round	$16.74	Yes	488	Phillips
M12	1.75	7g	12	Pan	$18.26	No	998	Flat
M14	2	7g	14	Round	$21.19	No	235	Phillips
M16	2	8g	16	Button	$23.57	Yes	292	Both

SQLFire Enterprise

- Stores Relational Data with SQL interface
- Supports JDBC, ODBC, Java and .NET interfaces
- Use existing relational tools

© Copyright 2013 Pivotal. All rights reserved.

Gemfire Basic Concepts

- ***Cache*** represents a *member* of a GemFire Distributed System (a node, a JVM)
  - Data is stored in one or more ***Regions***
  - Effectively a distributed key/value store, implementing java.util.Map
  - Has own OQL syntax for queries

```
Person person = new Person(1,"David","Turanski");
region.put(1,person);
person = region.get(1);
region.query("SELECT * FROM /Person WHERE lastname = 'Turanski'");
```


© Copyright 2013 Pivotal. All rights reserved.

Spring Data Gemfire Entity Mapping

- Mapping simpler than JPA
  - Objects stored directly, no columns

```
@Region
public class Contact implements Serializable {

    private static final long serialVersionUID = 1L;

    @Id
    private Long id;

    private String firstname;

    private String lastname;

    private Set<Phone> phones = new HashSet<Phone>();

    private Set<EmailAddress> emailAddresses = new HashSet<EmailAddress>();
```


© Copyright 2013 Pivotal. All rights reserved.

Spring Data Gemfire Repositories

- Based on repository infrastructure in Spring Data Commons
  - Define repository interface and namespace configuration
  - Just like JPA example earlier

```
public interface ContactRepository extends GemfireRepository<Contact, Long> {
    List<Contact> findByFirstname(String firstname);
    List<Contact> findByLastnameStartsWith(String lastname);
}
```

```
<gfe:repositories base-package="com.acme.**.repository" />
```


© Copyright 2013 Pivotal. All rights reserved.

Use OQL For Query

```
public interface ProductRepository extends CrudRepository<Product, Long> {

    @Query("SELECT * FROM /Product where attributes[$1] = $2")
    List<Product> findByAttributes(String key, String value);
```

JPA example used JPQL,
Gemfire requires its OQL

```
@Region
public class Product extends AbstractPersistentEntity {

    private String name, description;
    private BigDecimal price;
    private Map<String, String> attributes = new HashMap<String, String>();
```


© Copyright 2013 Pivotal. All rights reserved.

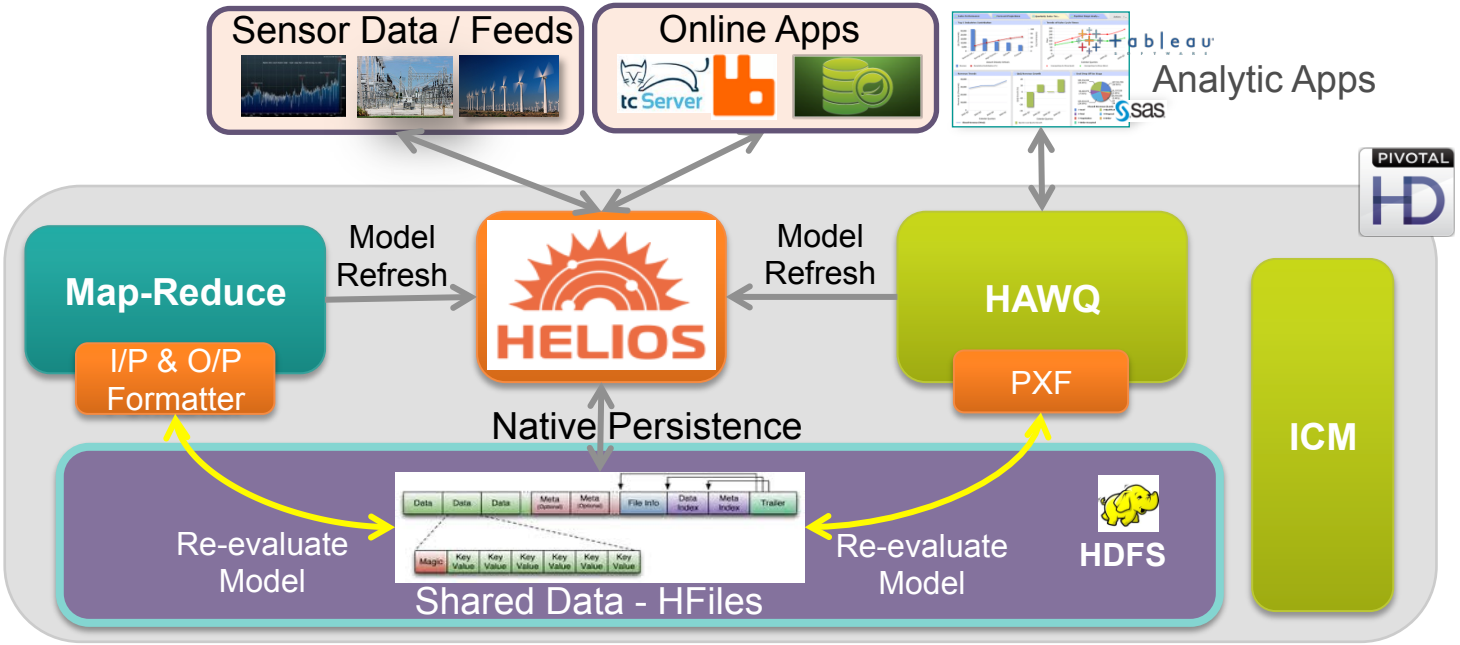
Let’s Get Pivotal

- Pivotal Stack
- CloudFoundry Open PAAS
- GemFire Data Fabric
- **Pivotal Helios**

© Copyright 2013 Pivotal. All rights reserved.

Pivotal Helios

- Enterprise Real-Time Data Service on Pivotal HD



© Copyright 2013 Pivotal. All rights reserved.

Questions?

© Copyright 2013 Pivotal. All rights reserved.

Thank You

© Copyright 2013 Pivotal. All rights reserved.

Pivotal

BUILT FOR THE SPEED OF BUSINESS