cloudsoft

Bringing Business to the Cloud

Wide-area Migration with Monterey, AS7, Seam and jclouds

JUDCon 2011:London

31 October 2011

Alex Heneveld, CTO & **Aled Sage**, VP Engineering

Cloudsoft Corporation

cloudsoft
Bringing Business to the Cloud

Company Intro

- **Who are Cloudsoft?**
  - **Venture-backed** software company headquartered in **UK**
  - **Experienced team**
    - **Management team:** Duncan Johnston-Watt (CEO & founder), Alex Heneveld (CTO & co-founder), Adrian Cole (Chief Evangelist; jclouds.org founder), Aled Sage (VP Engineering)
    - **Non-exec team:** Derek Gray (UK), Shawn Findlan (NY), Linda Bernardi (MA), Rich Miller (CA), John Mathon (CA), Erik Troan (NC), Lawrence Rosen (CA)
  - **Recognised by**
    - **Gartner:** Cool Vendor 2011 (application and integration platforms)
    - **Forrester:** leading enabler of Elastic Application Fabrics
    - **Patents:** holder of several patents in key jurisdictions

- **What do we do?**
  - Cloudsoft's mission is to **help enterprises** make the most of cloud
    - **Large-scale, globally distributed** applications
    - **Complex, multi-tier, transactional, high-throughput** applications
  - We develop, sell, and support a **middleware platform** that simplifies the **development** and **runtime deployment and management** of such applications

Overview

Combining CDI, JBoss, Monterey, and jclouds:

- **CDI is great for wiring business logic together**
- **JBoss is a modular, powerful app-server**
- **Monterey is a dynamic wide-area processing fabric**
- **jclouds abstracts cloud APIs**

Combination gives you best-of-breed for different parts of app

- **Eliminate database performance bottlenecks**
- **Get consistency and transactional guarantees without sacrificing high-throughput**
- **Continue to focus on just writing business logic**

Simple CDI Example

```
public class Printer {

    @Inject Hello hello;

    public void printHello() {
        System.out.println( hello.sayHello("world") );
    }
}
```

`@Inject` defines an injection point. `@Default` qualifier is assumed

Simple JPA Example

```
@Entity
public class Customer { ... }
```

Annotations describe the persist mapping.

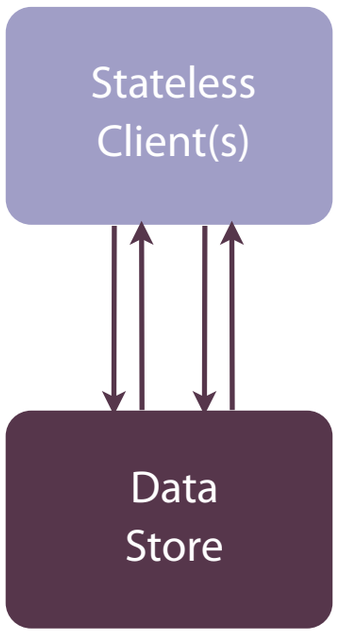
Transactions are often “get” then “put”.

```
em.createQuery("select c ... ");
Customer c = (Customer)q.getSingleResult();

// modify customer

em.persist(c)
```

The Challenge



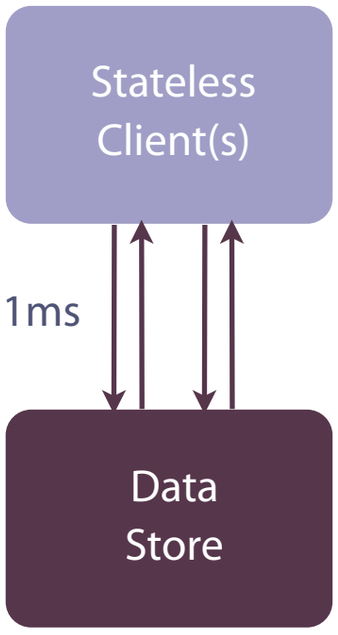
State lives in a data store, not in code.

This paradigm works well for a certain class of problems…

…but it transfers several hard problems to the data tier.

For volatile data, for full-consistency, or for real-time data-intensive analysis, it hits scalability issues.

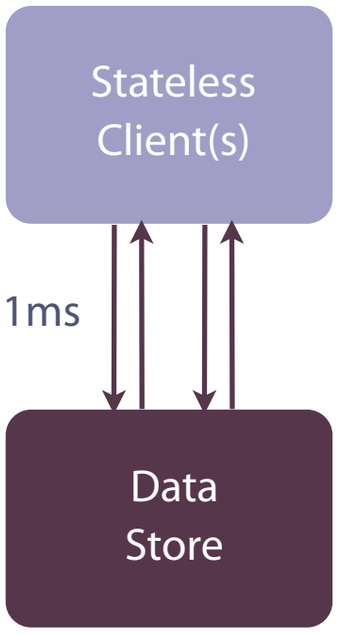
The Challenge



1ms x 4 = 4ms

…that's maybe okay.

The Challenge



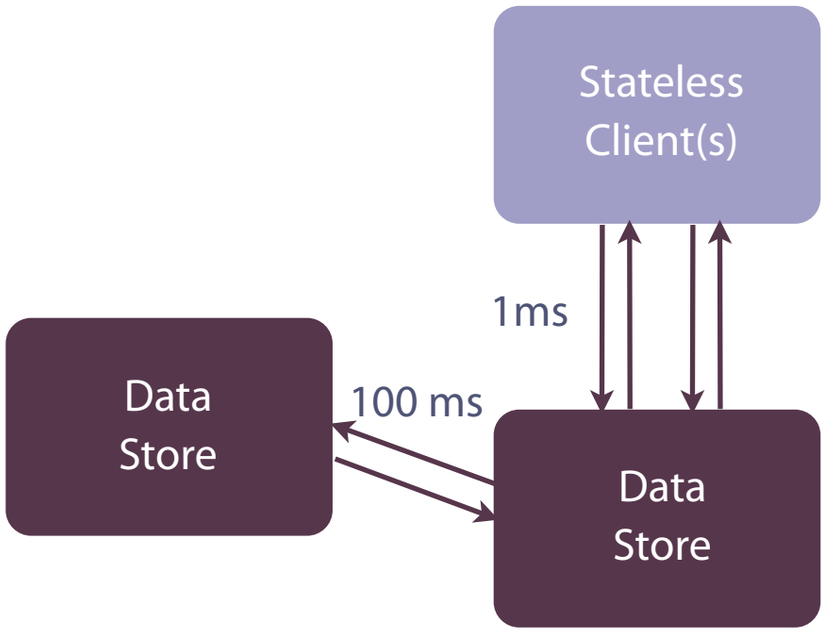
1ms x 4 = 4ms

…that's maybe okay.

**But we max out at 250 txns/sec max,
even less if our computation is expensive,**

and we thrash as we approach that limit!

The Challenge



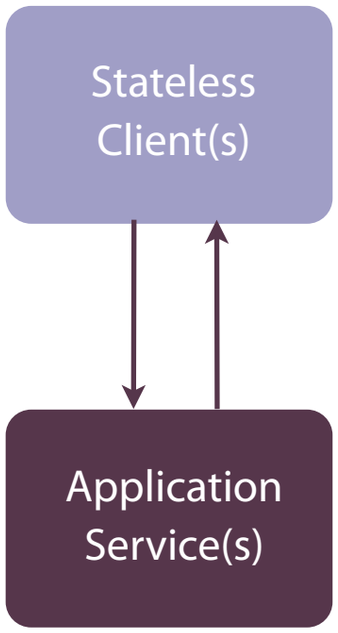
Wide area, or replicating, we're at ~200 ms per transaction…

…5 txns/sec max…

…still with thrashing and blocking and backlog as we approach that limit.

cloudsoft
Bringing Business to the Cloud

How Monterey Works



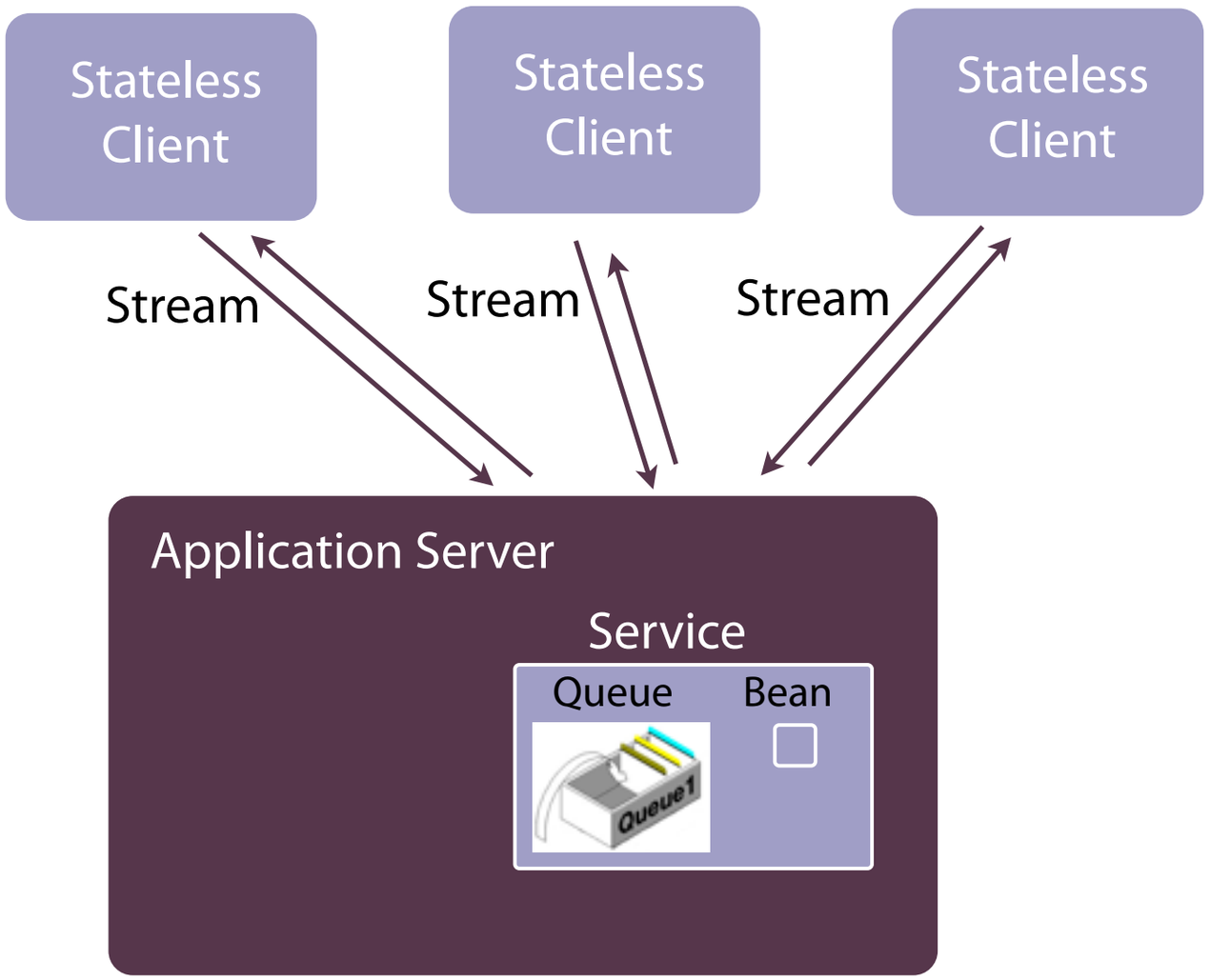
Monterey makes stateful processing available to all client instances, implemented as normal code…

…called *segments* or *actors*…

…without the penalties of optimistic locking, and with the added benefits of *application mobility* for *runtime agility.*

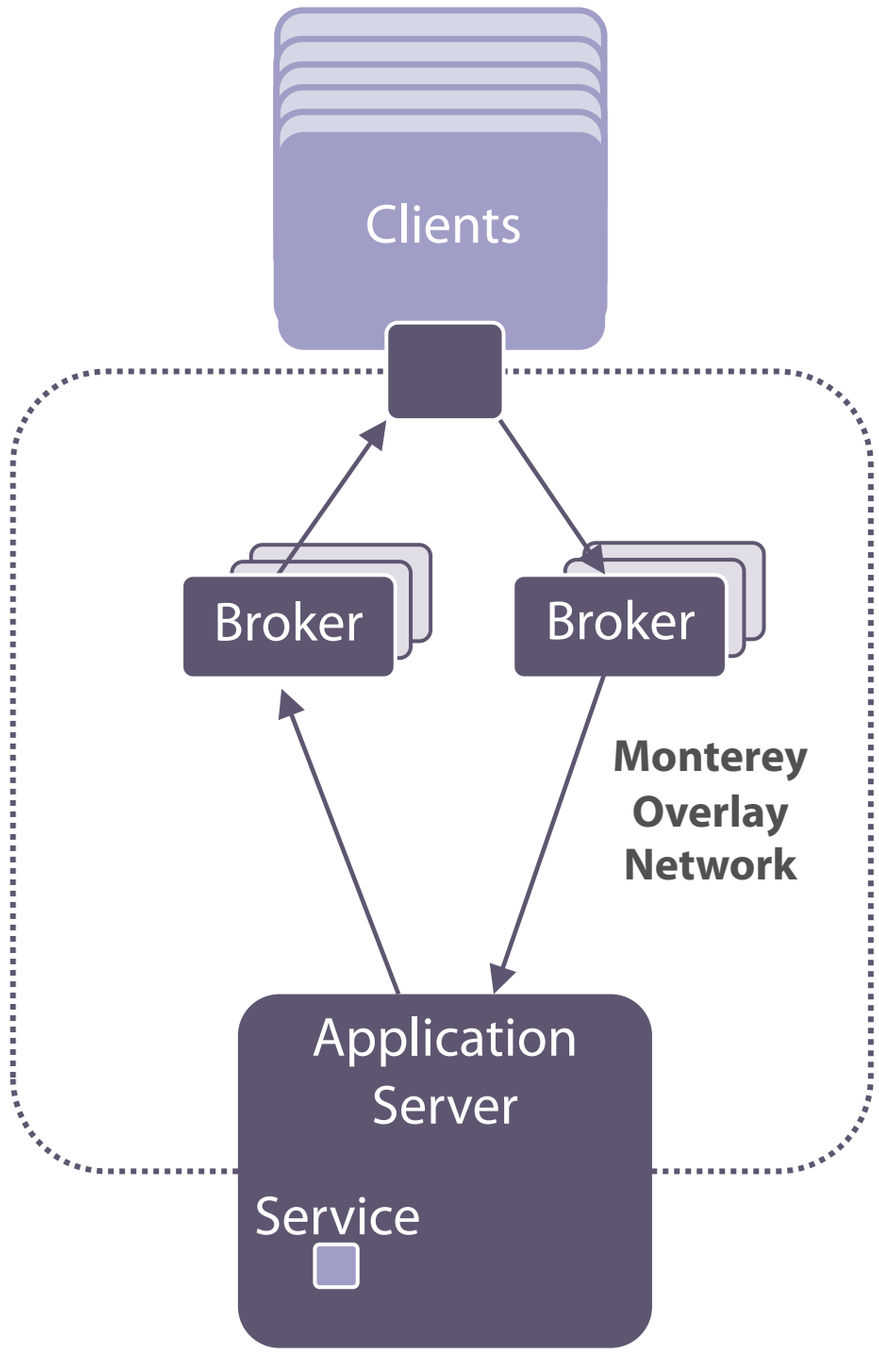
cloudsoft
Bringing Business to the Cloud

How Monterey Works



cloudsoft
Bringing Business to the Cloud

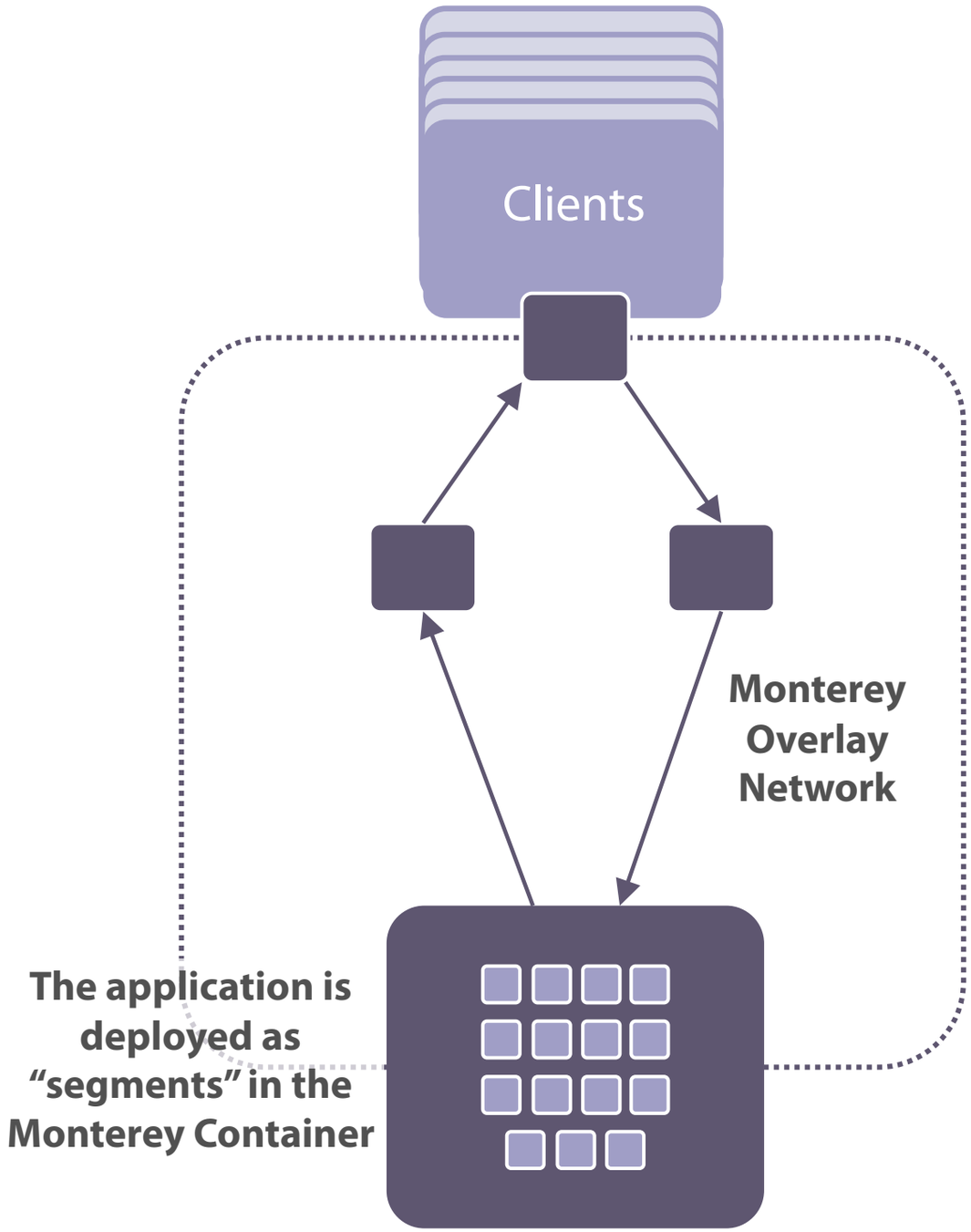
How Monterey Works



- Message brokers route the messages
- Use independent upstream and downstream paths
- Communication is asynchronous: don't block for acks

cloudsoft
Bringing Business to the Cloud

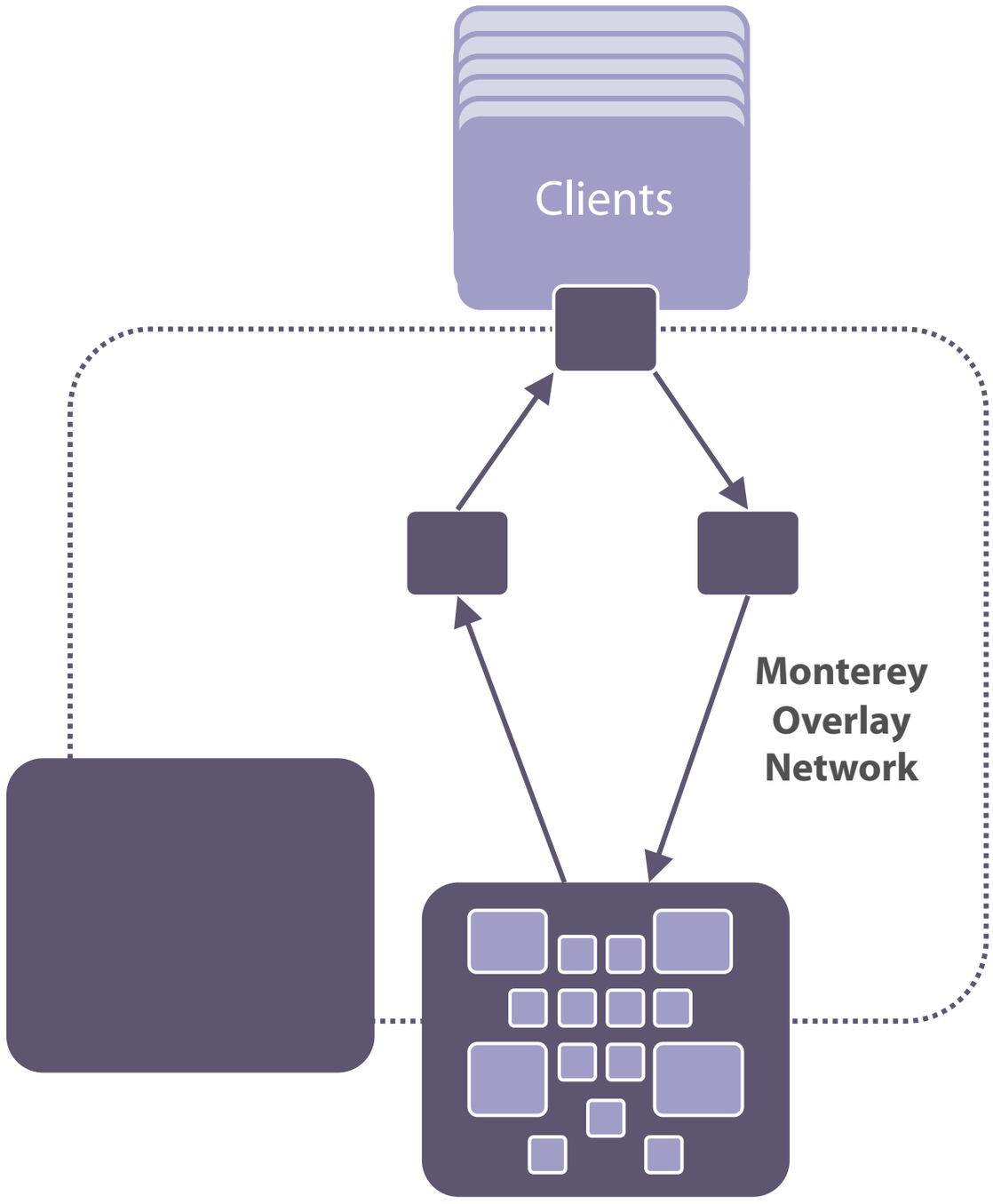
How Monterey Works



- The Service(s) are automatically deployed as very fine-grained components (“segments”) in the Monterey overlay network
- The overlay network transparently manages requests/responses between the Client(s) and segments
- Monterey’s user-defined policies control the running application and dynamically change its deployment as required

cloudsoft
Bringing Business to the Cloud

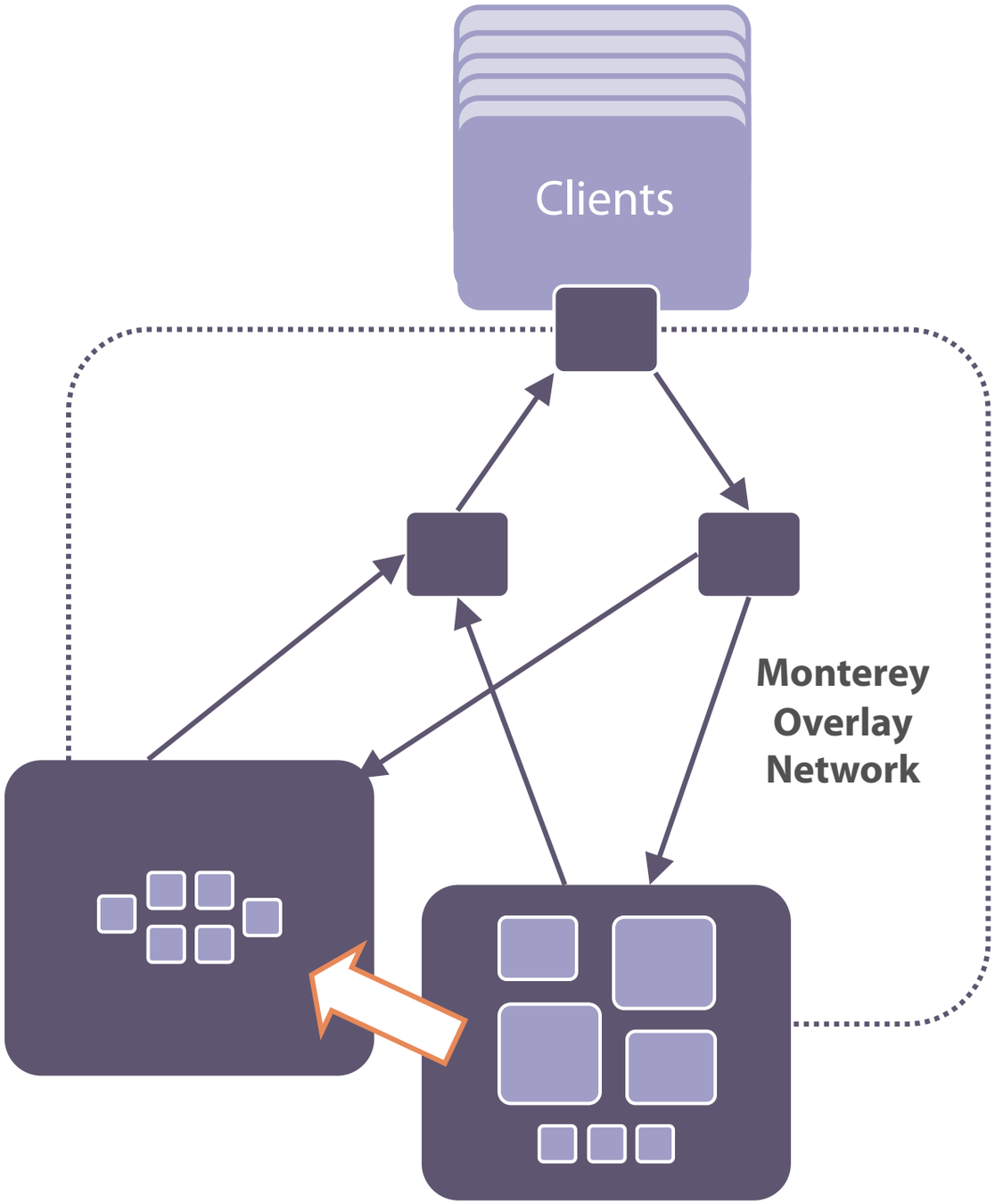
How Monterey Works



- Monterey requests additional resources as necessary (using jclouds to support EC2, cloud.com, vcloud, and many more)
- Decisions are made automatically by Monterey's real-time monitoring working in conjunction with user-defined policies

cloudsoft
Bringing Business to the Cloud

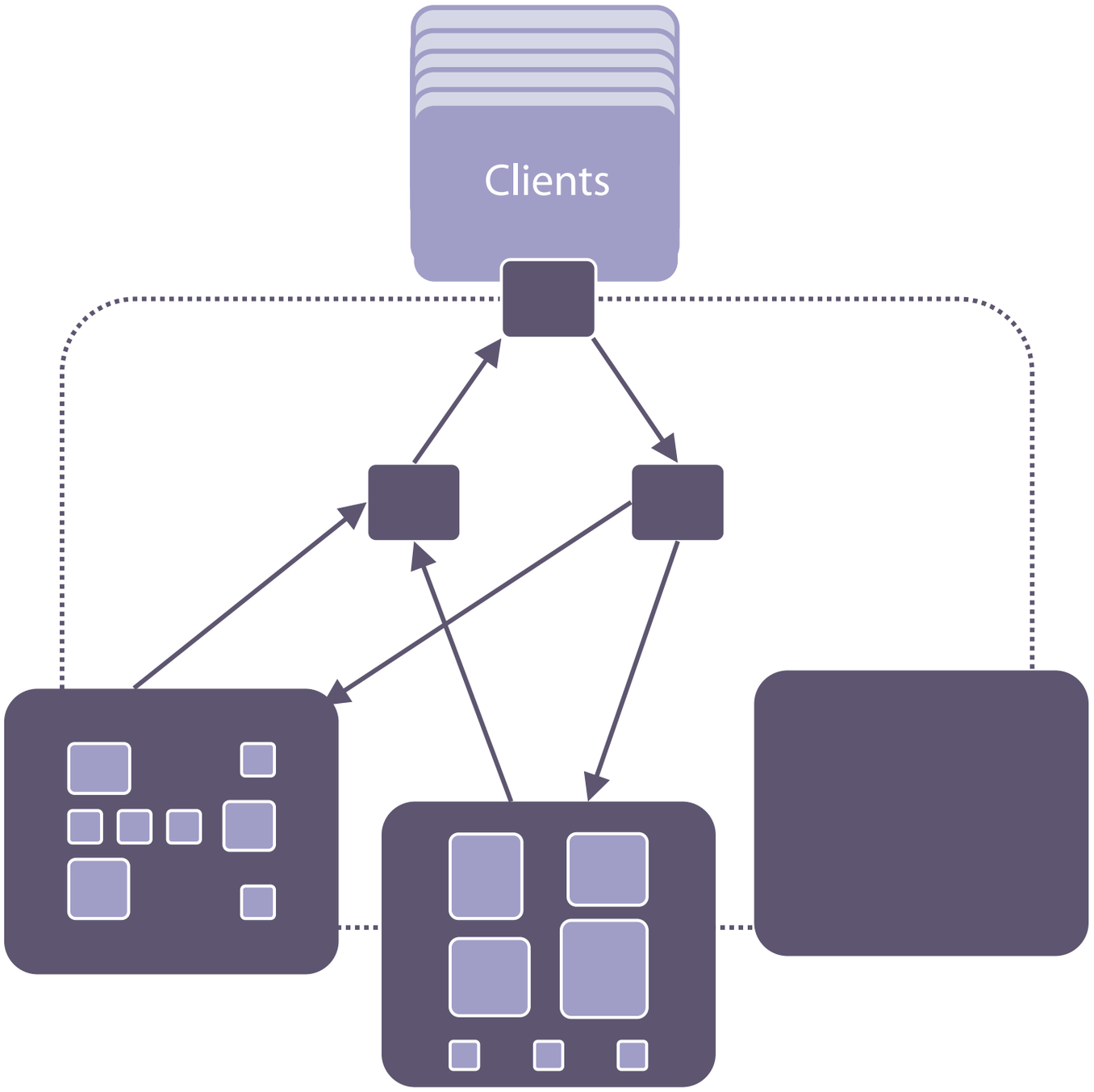
How Monterey Works



- Monterey dynamically re-allocates segments to resources
- Segments move while still running, with zero interruption or degradation to service
- All changes are completely transparent to clients

cloudsoft
Bringing Business to the Cloud

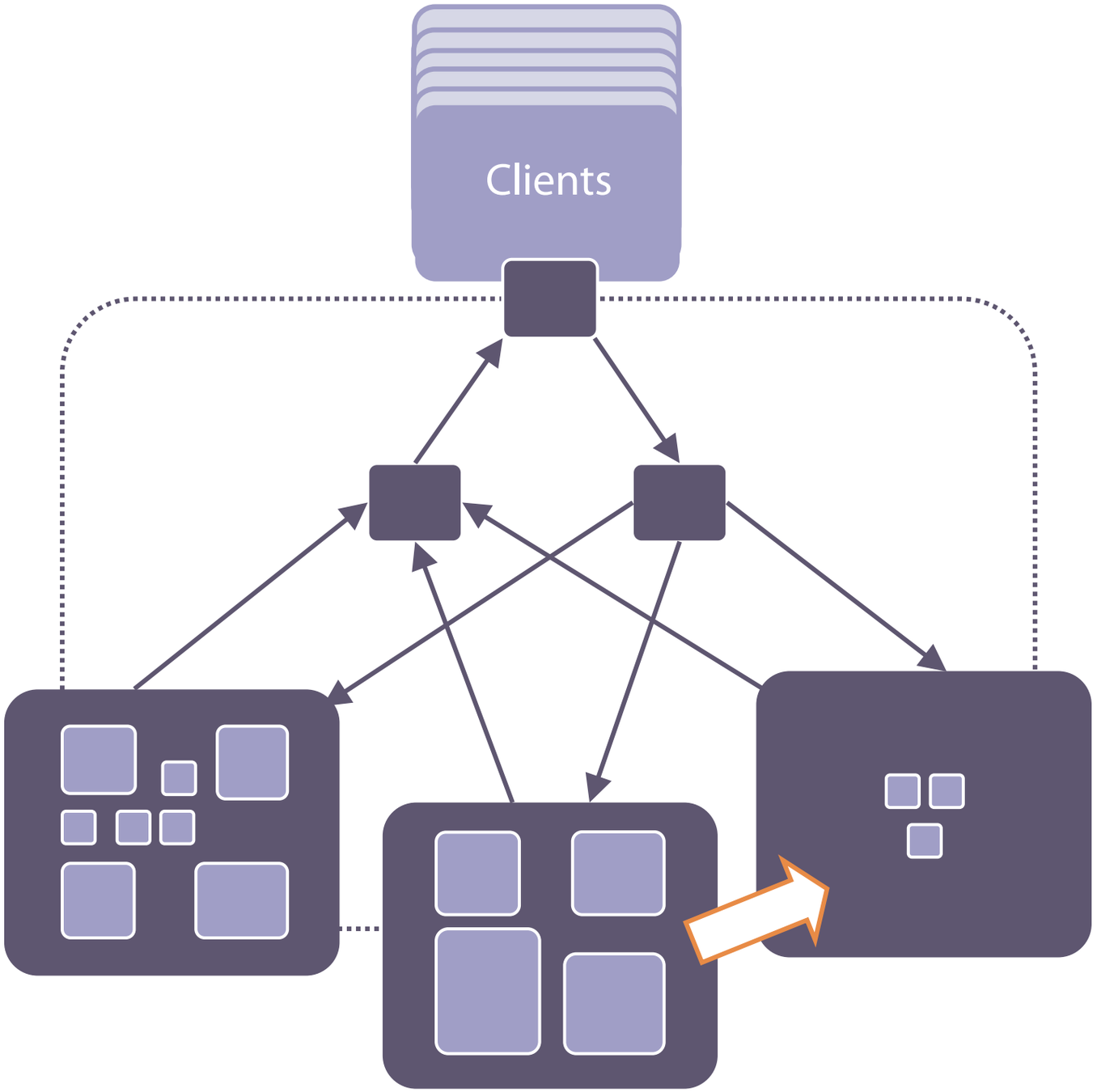
How Monterey Works



- The amount of available resource can be continually adjusted in line with workloads

cloudsoft
Bringing Business to the Cloud

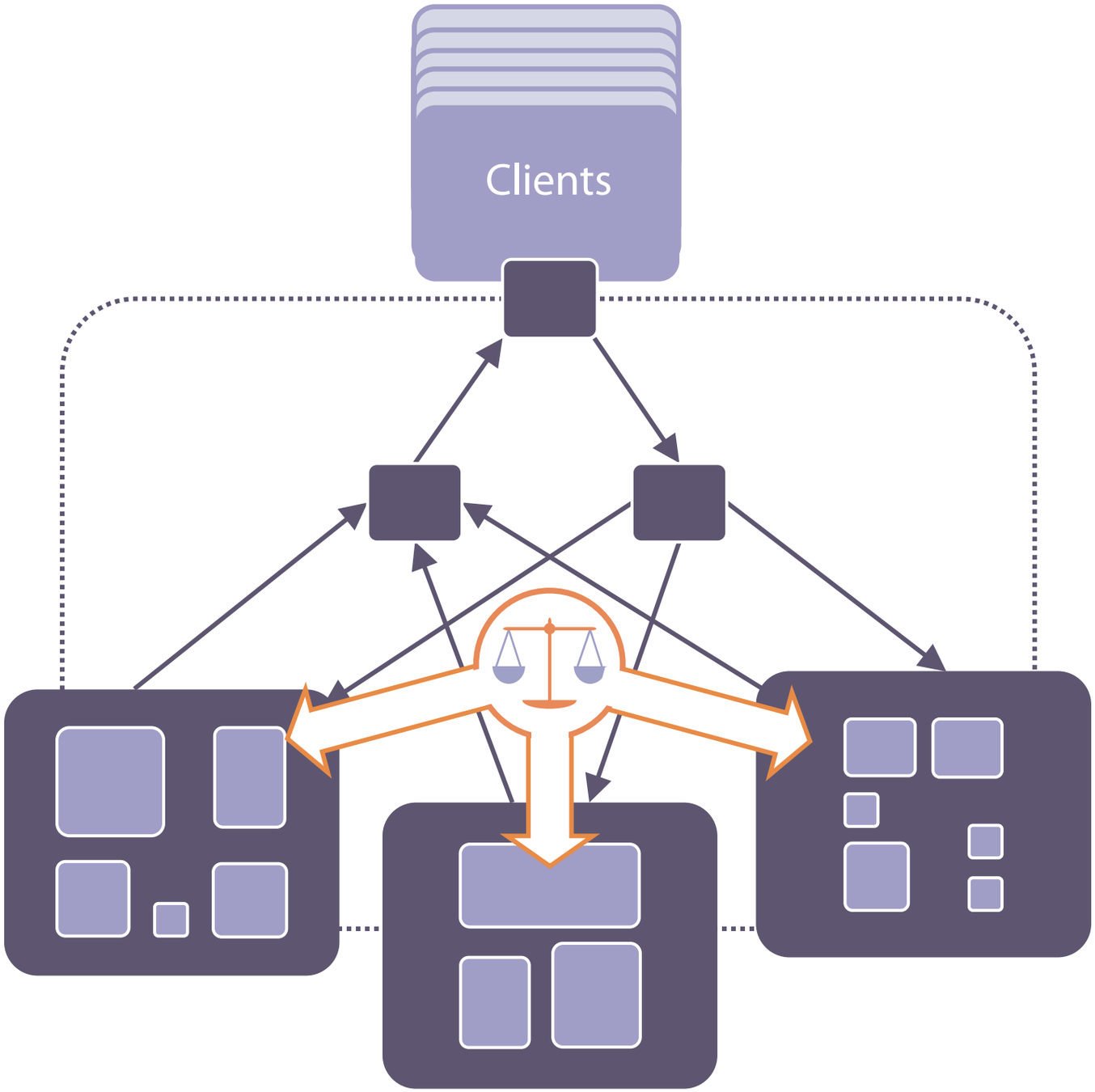
How Monterey Works



- The service can be further scaled-out by re-distributing segments – with no pause in their operation
- At all times transactional integrity is maintained

cloudsoft
Bringing Business to the Cloud

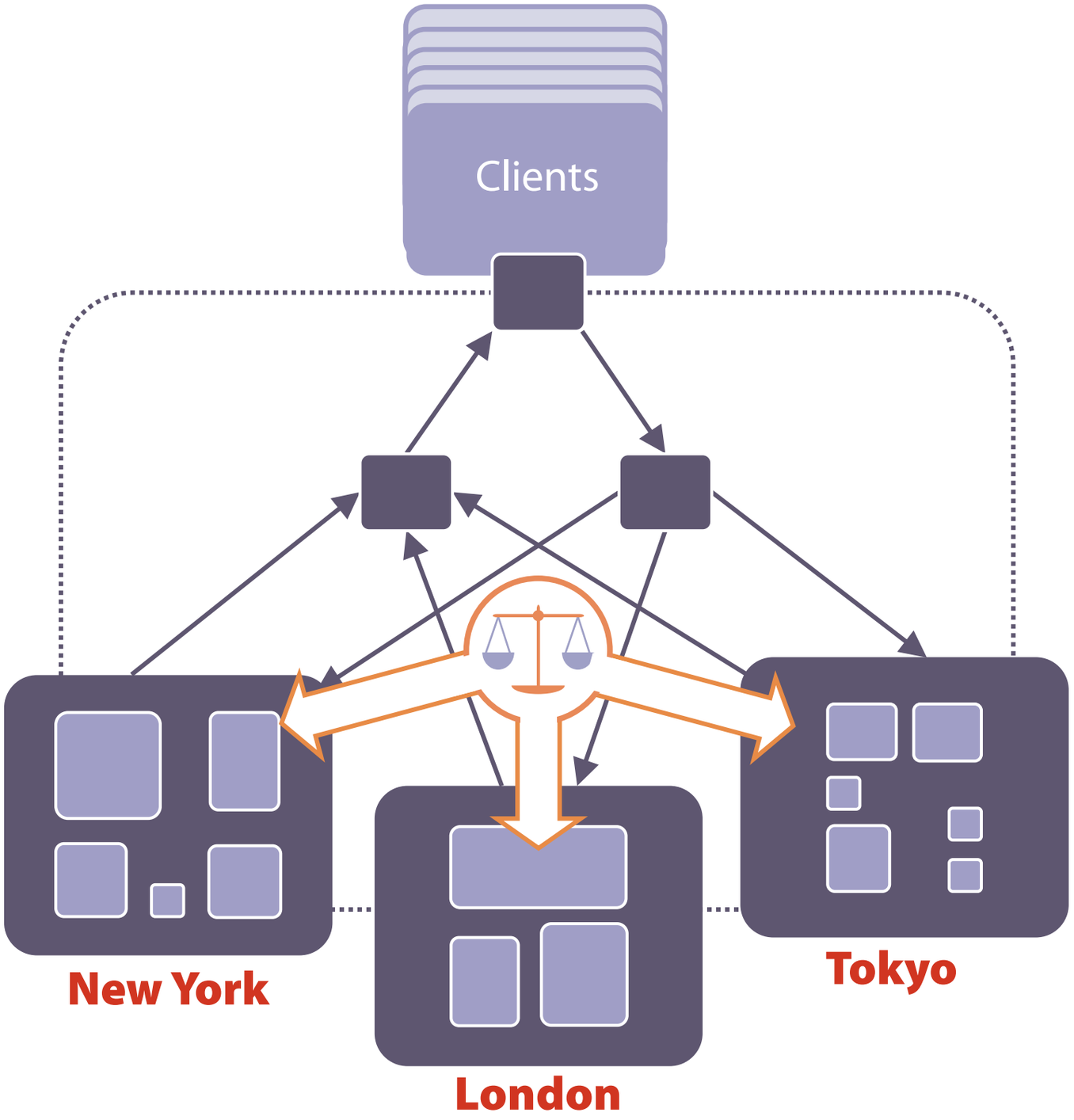
How Monterey Works



- The matching of segment workloads to resources is monitored and managed in real-time
- Dynamic re-configuration and relocation of segments is completed within milliseconds, creating near-instant responsiveness

cloudsoft
Bringing Business to the Cloud

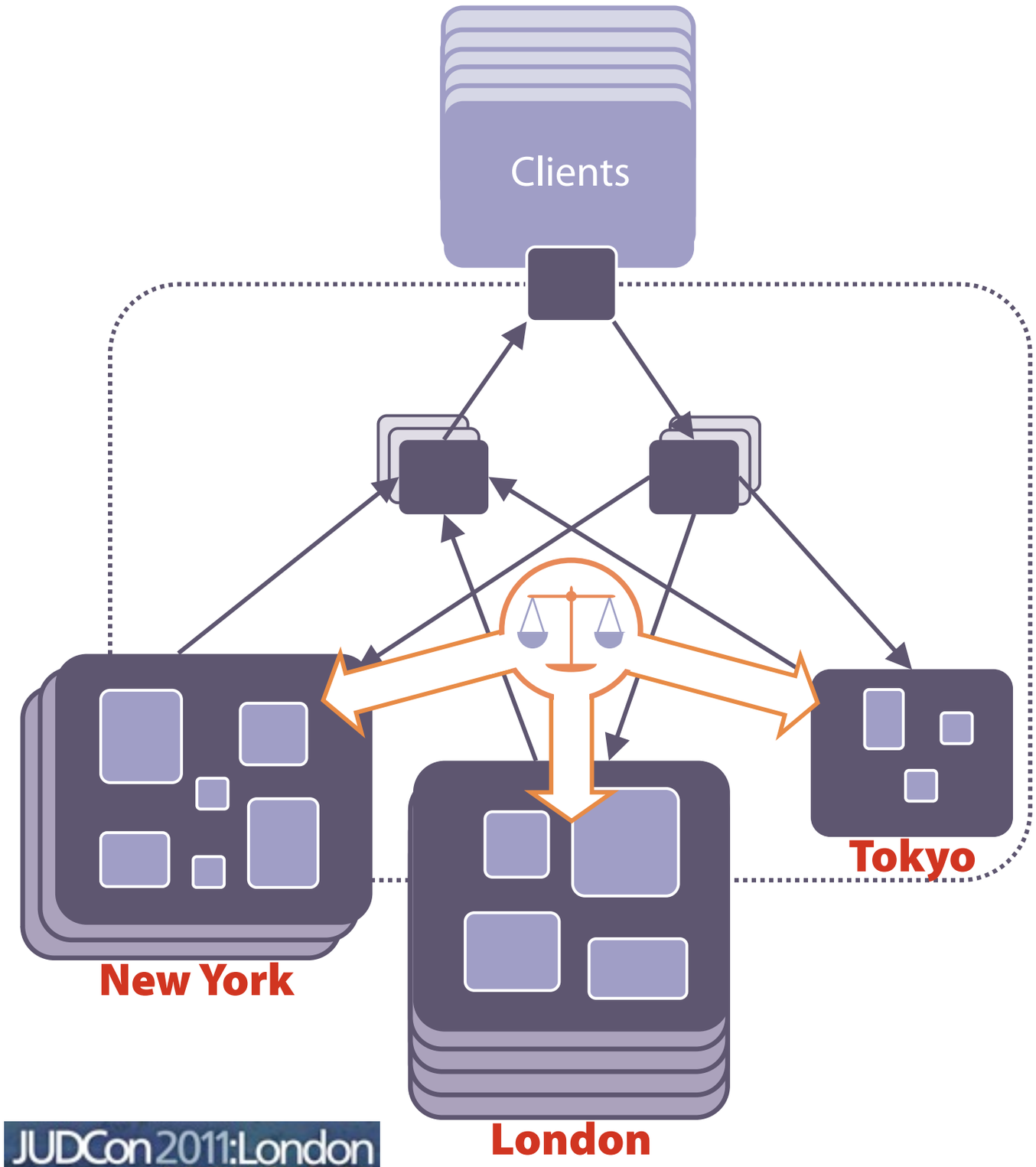
How Monterey Works



- Because components are very fine-grained, they can move quickly and in large numbers across wide-area networks (*e.g.* "follow-the sun" processing)
- Monterey is location-aware, and its policies ensure jurisdictional constraints are always enforced

cloudsoft
Bringing Business to the Cloud

How Monterey Works



- Clients, services, containers, segments, and routing nodes can all scale independently
- Monterey dynamically grows or shrinks resources as required, both within and across data-centres
- Any number of clients and applications can run simultaneously, with policies controlling the entire estate down to a per-segment & per-location level

Application Mobility

If your application can move, you have

elasticity

portability

wide-area

cloudsoft
Bringing Business to the Cloud

Application Mobility

Application mobility is the ability
to **dynamically change**
any part of the **infrastructure**
that an application is using
without any disruption of service.

cloudsoft
Bringing Business to the Cloud

Intelligent Application Mobility

***Intelligent* application mobility** is
the ability to fully exploit application mobility
by **monitoring demand and performance**
and using **strategic, constraint-based policies**
to drive an **intelligent** set of **responses**

cloudsoft
Bringing Business to the Cloud

CDI + Monterey

- **Monterey hosts your POJO stateful business logic**
- **Monterey-Seam uses CDI injections to supply interfaces to these remote Monterey actors (application segments) in your CDI beans in AS7**
- **Callers don't care where in the world processing is happening; they just write their code as normal!**

cloudsoft
Bringing Business to the Cloud

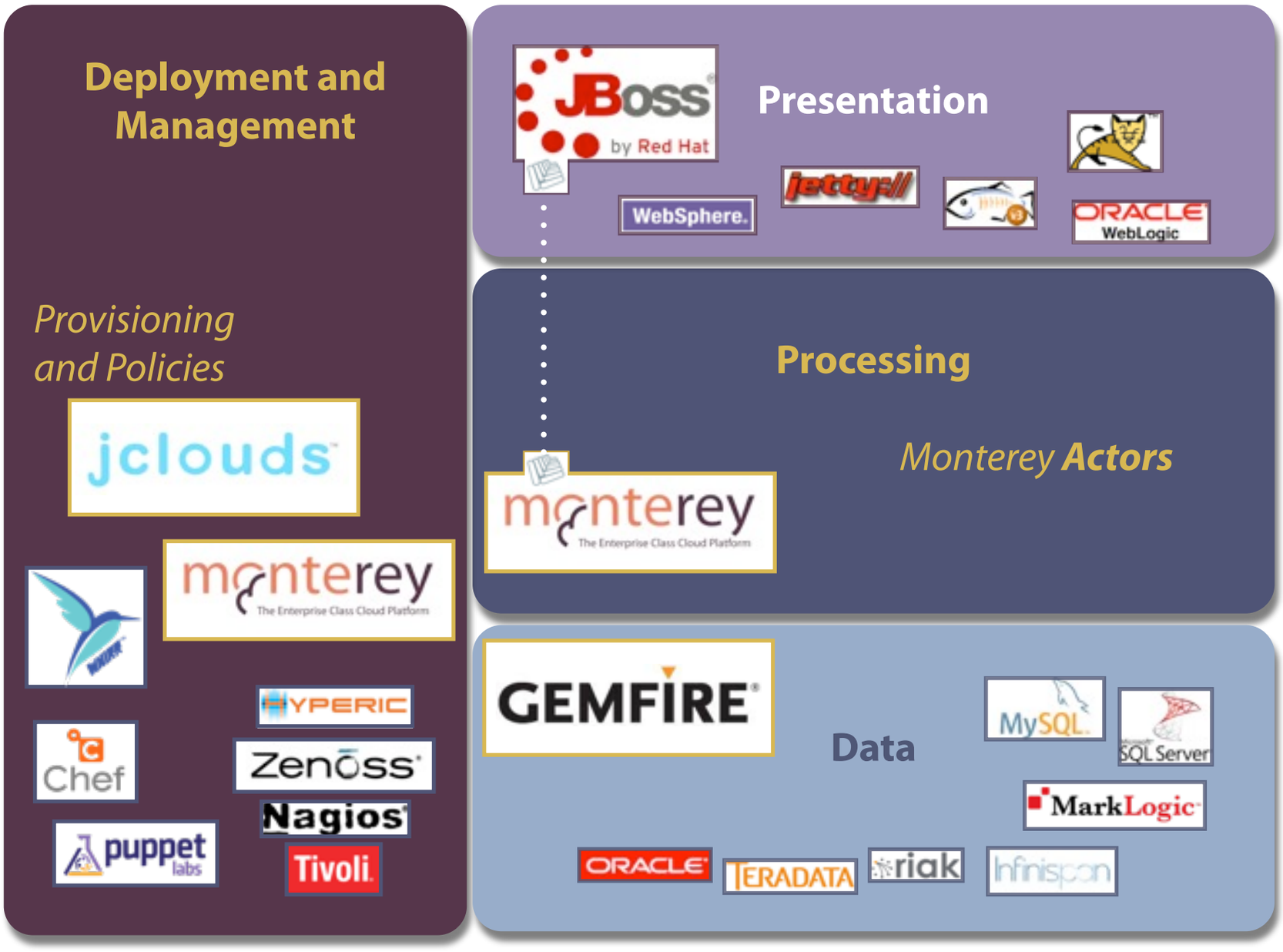
CDI + Monterey

- **Monterey hosts your POJO stateful business logic**
- **Monterey-Seam uses CDI injections to supply interfaces to these remote Monterey actors (application segments) in your CDI beans in AS7**
- **Callers don't care where in the world processing is happening; they just write their code as normal!**

  - Monterey transparently routes the calls to correct locations
  - Even as these **stateful beans move**
  - With many options for **resilience and persistence**
  - Using **jclouds** to access many hosting providers
  - Optimizing for **latency** or **cost** or **jurisdiction**

cloudsoft
Bringing Business to the Cloud

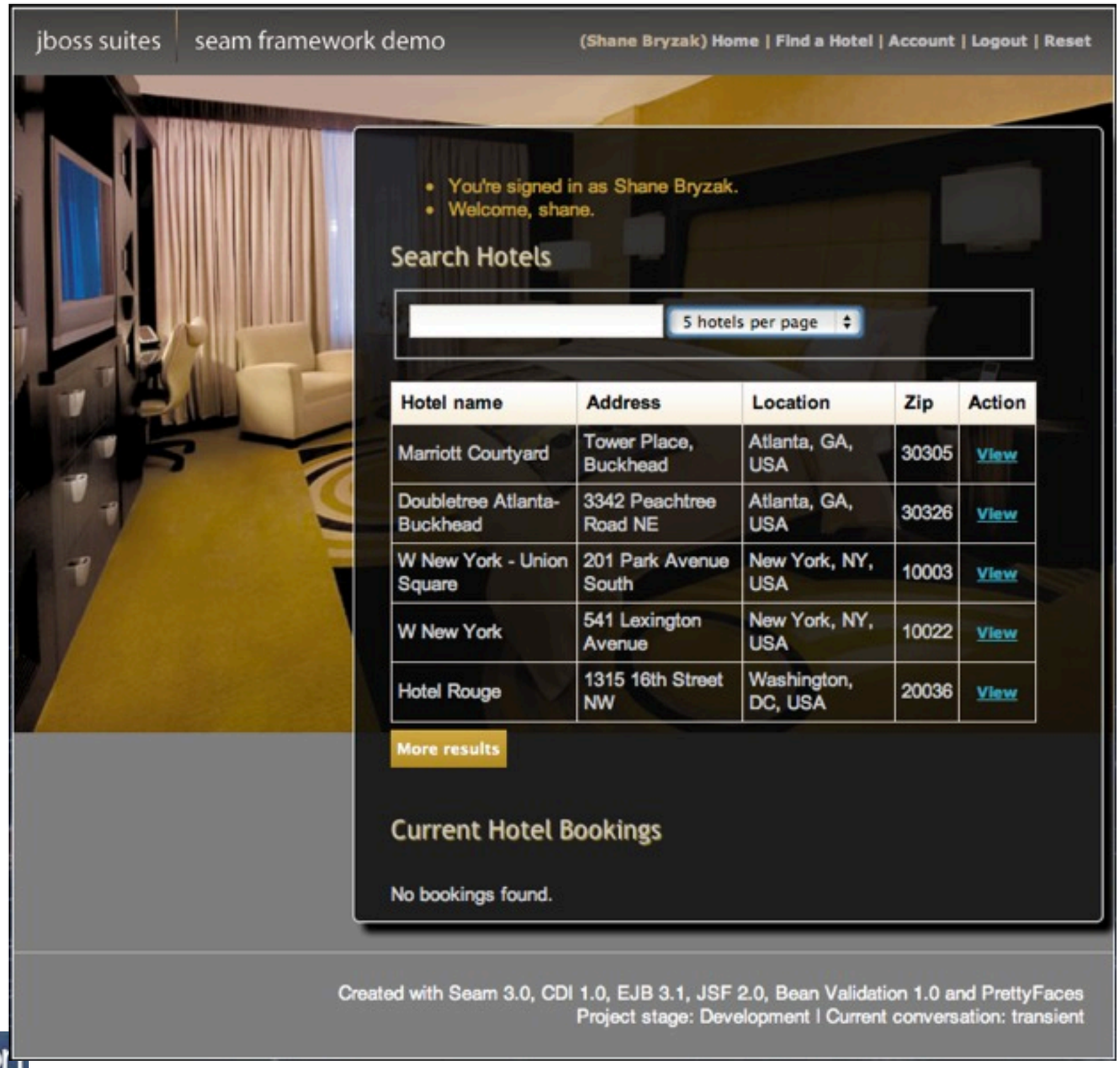
Software Overview



Download Now: http://www.cloudsoftcorp.com/developers

cloudsoft
Bringing Business to the Cloud

Seam Booking



Thank-you ,

monterey
The Enterprise Class Cloud Platform

jclouds™

- **Try them for free:**
  - **www.cloudsoftcorp.com/developers**
  - **www.jclouds.org/**
  - **seamframework.org/**
  - **www.jboss.org/as7**
- {**alex,aled**}@cloudsoftcorp.com

cloudsoft
Bringing Business to the Cloud

Wishlist for an Elastic App Tier

- **Design & Build**
  - Think in terms of **fine-grained components**
  - Provide a **domain specific API**
  - **Critical data** should live in-process
  - Test locally and at scale
  - Use **dependency injection**
- **Deploy & Run**
  - **Drag-and-drop** my app to a **simulator**, or a **cloud**, or an **in-house environment**
  - **Location awareness** to guarantee compliance
  - The ability to specify the **operational policies** that matter to me—**resilience**, **performance**, **cost**
- **Monitor & Manage**
  - **Insight** into **what each segment is doing**
  - **Visibility** as to **where each segment is running**
  - **Real-time policies** that **optimize** execution