



Extending Cloud Foundry With Custom Integration

Cornelia Davis
Scott Frederick

Who and What...

OPERATOR

cf-iaas.yml

provision <my cloud>

add_capacity <my cloud>

DEVELOPER

target <my cloud>

push <my app>

create <my services>

bind <my services>

scale <my app> +100

Who and What...

OPERATOR

cf-iaas.yml

provision <my cloud>

add_capacity <my cloud>

Extending the
set of services

DEVELOPER

target <my cloud>

push <my app>

create <my services>

bind <my services>

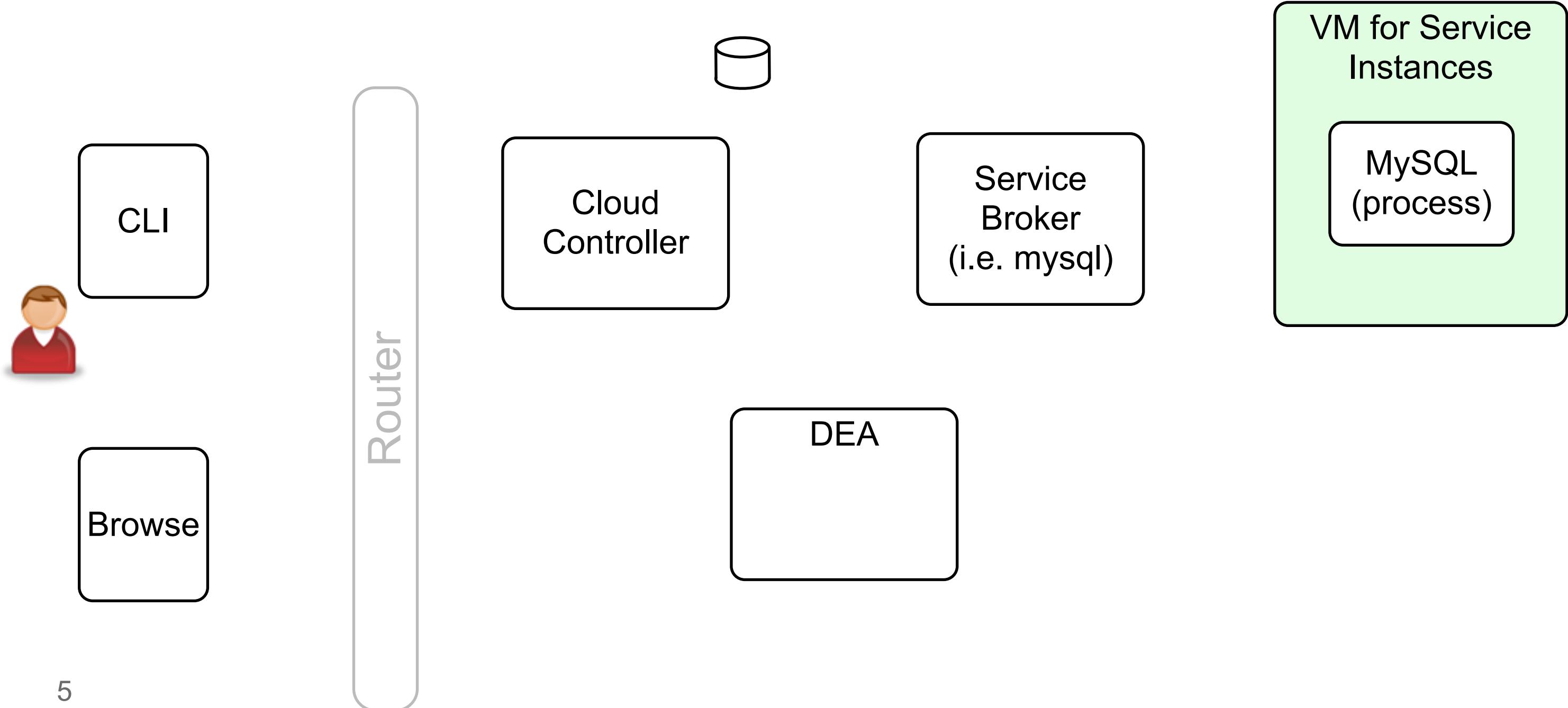
scale <my app> +100

Customizing
application containers

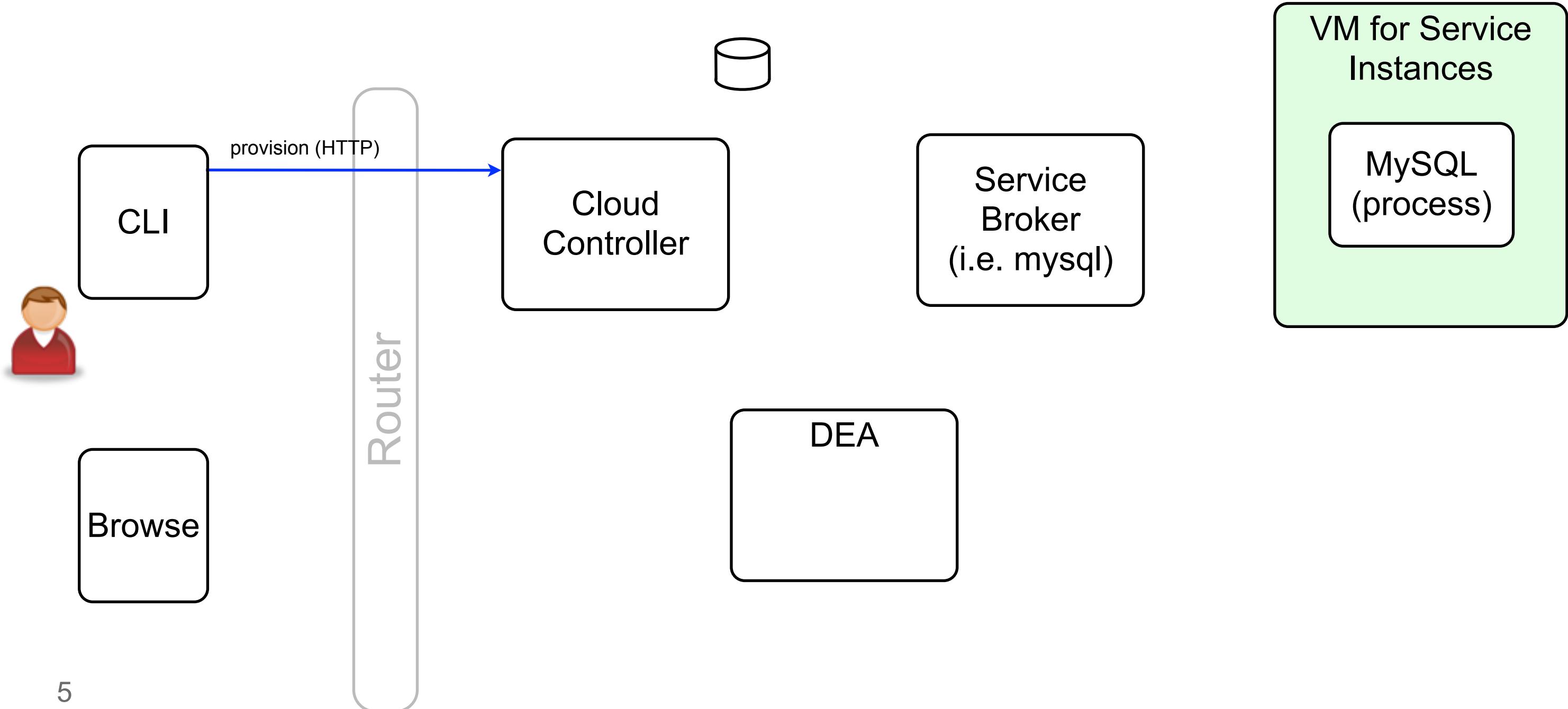
Services

Demo

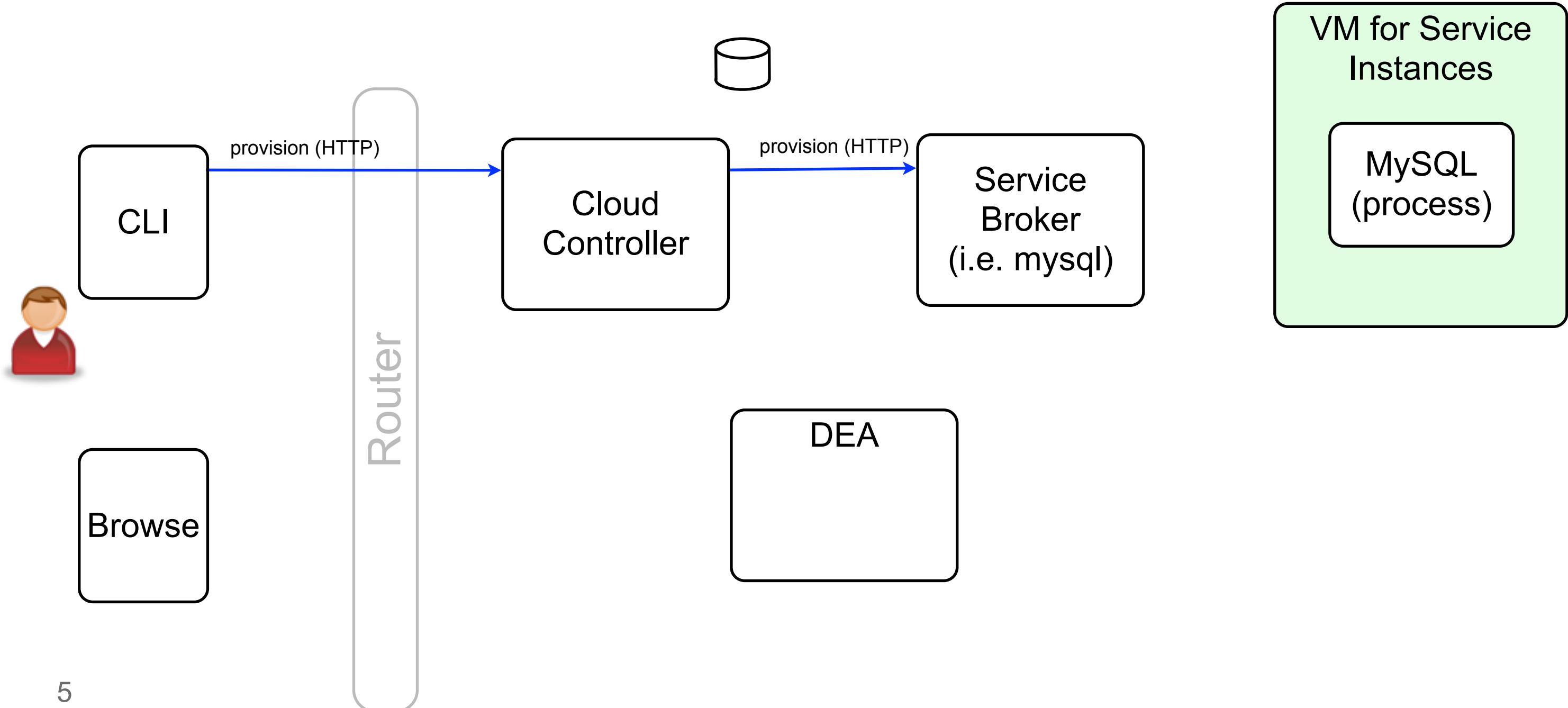
When you cf push



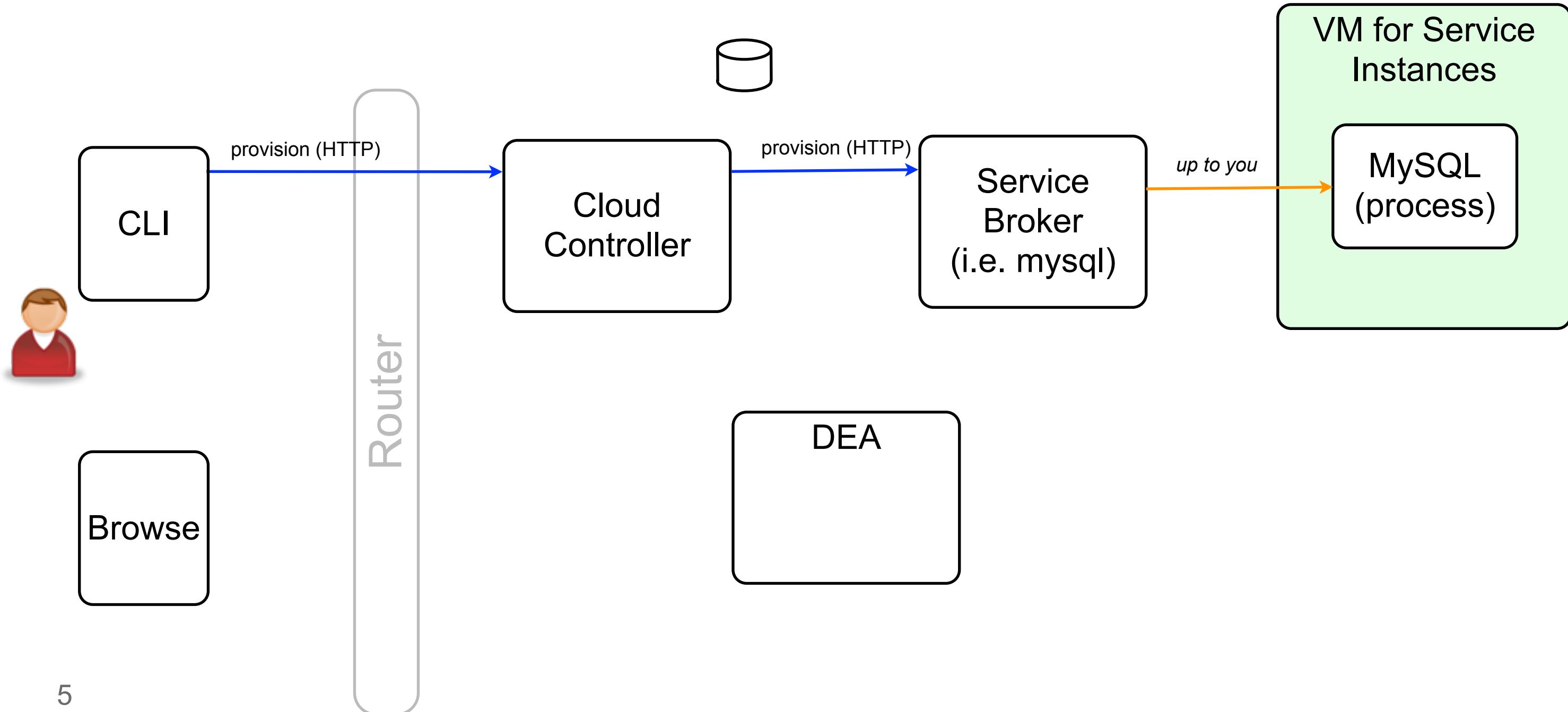
When you cf push



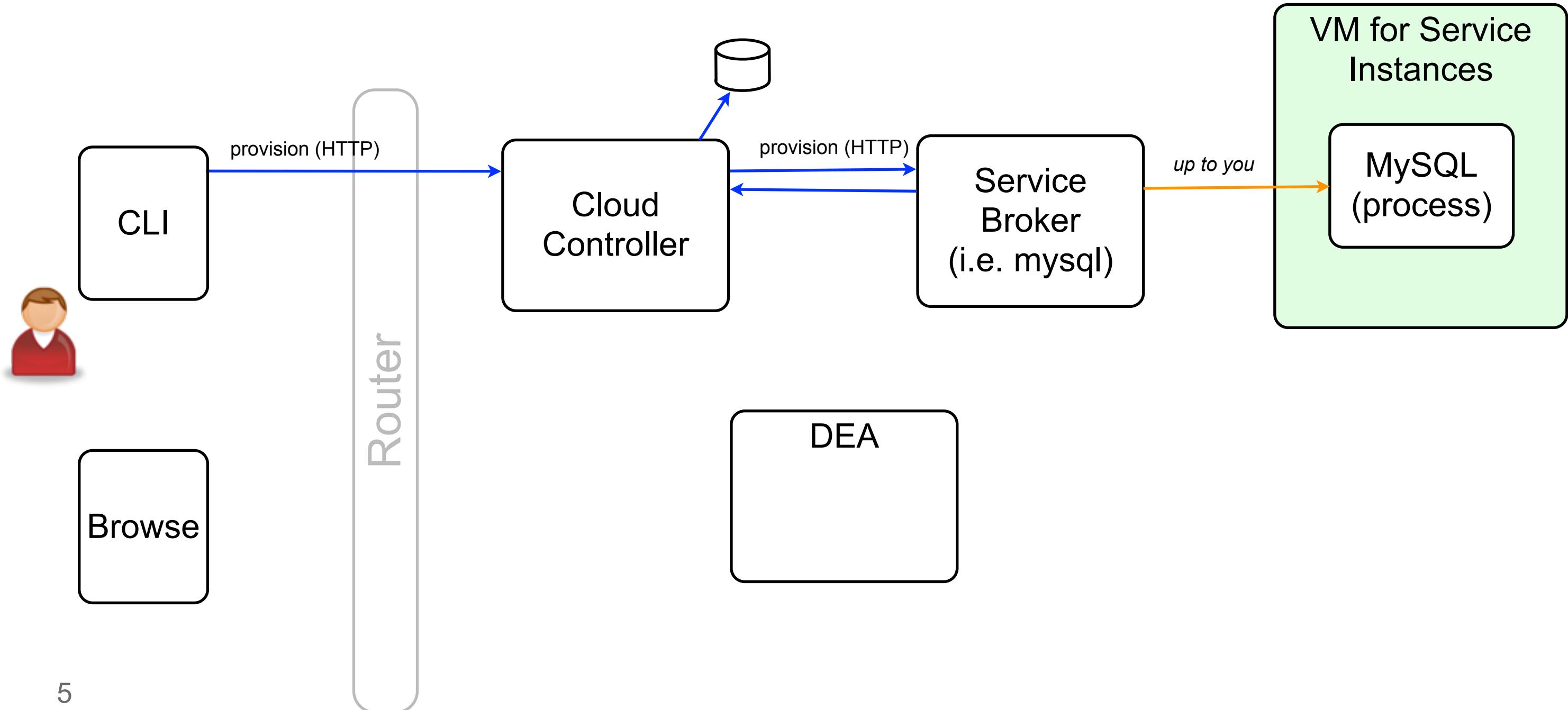
When you cf push



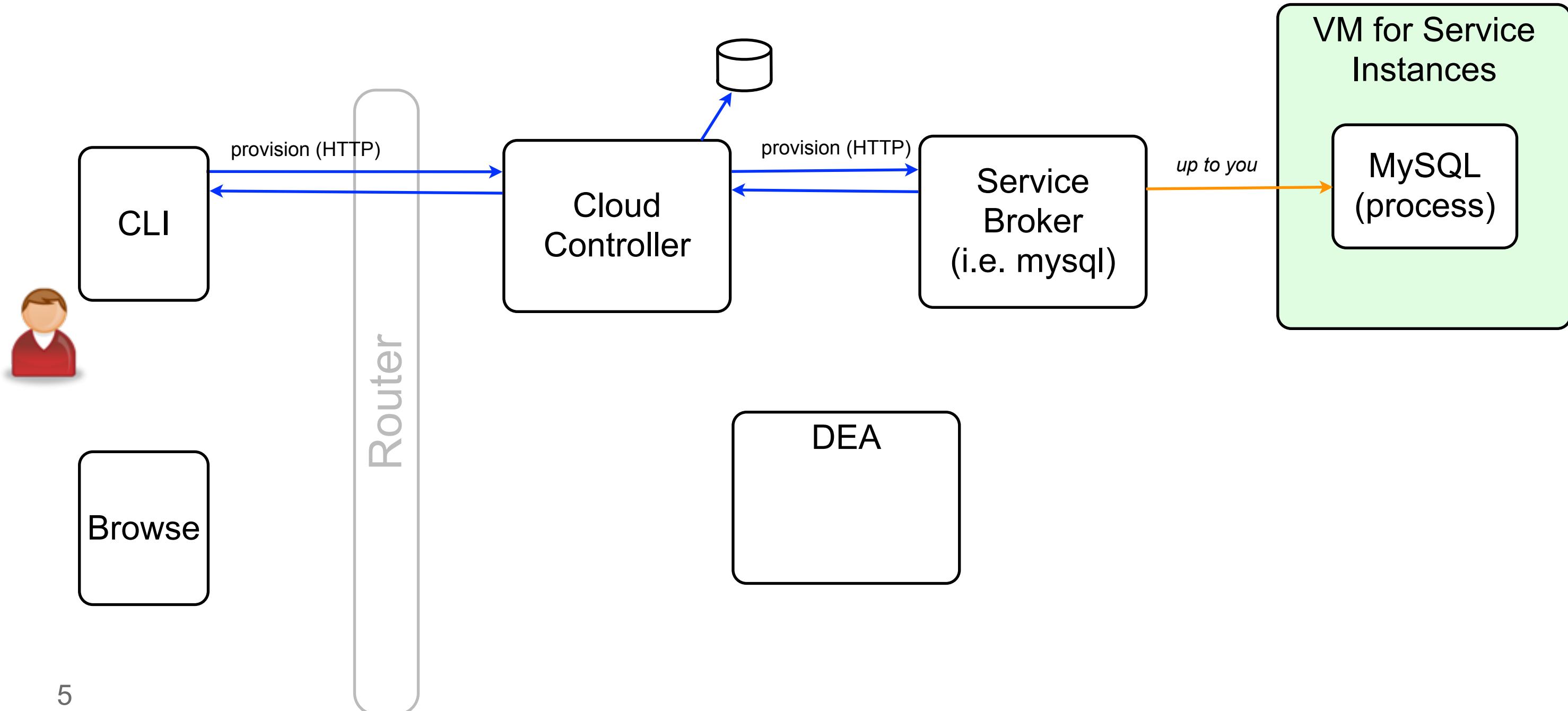
When you cf push



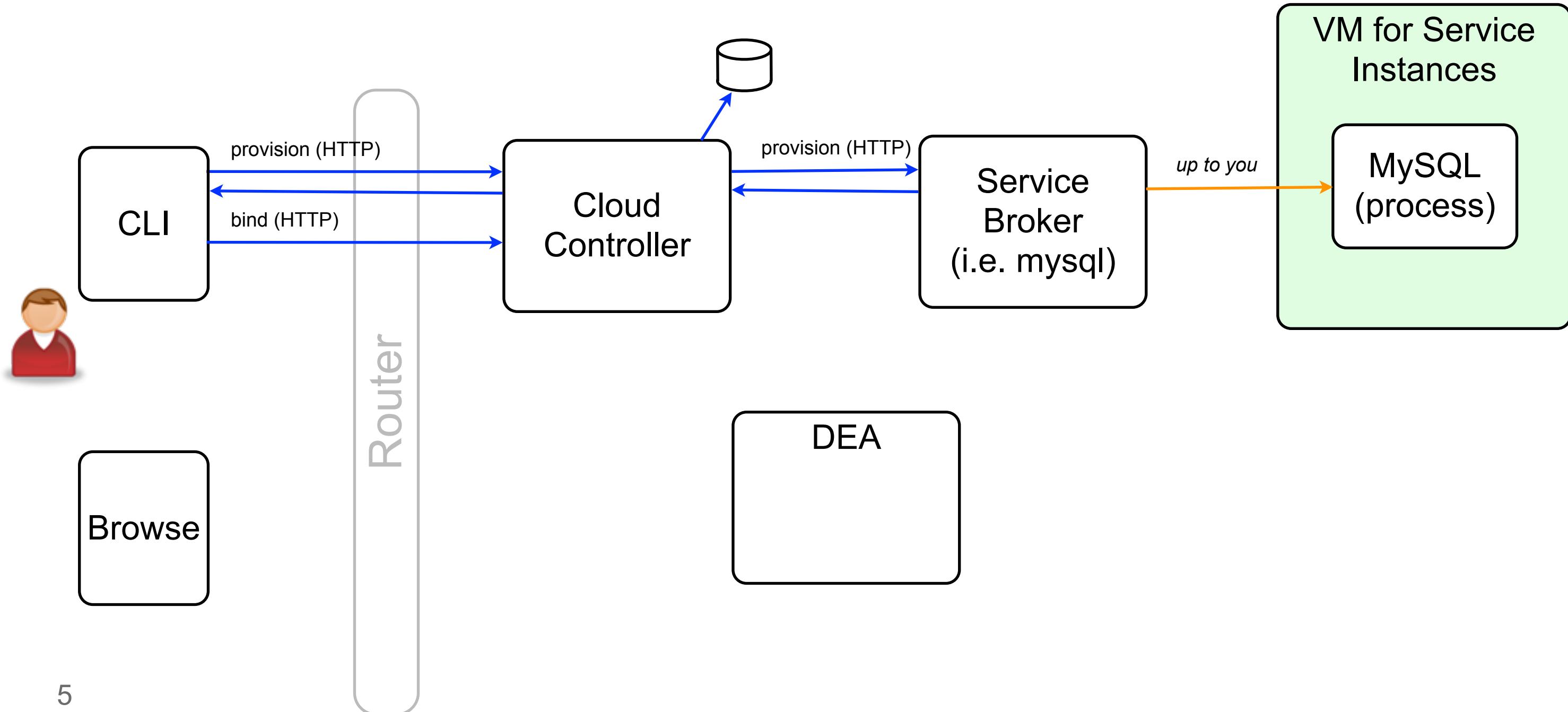
When you cf push



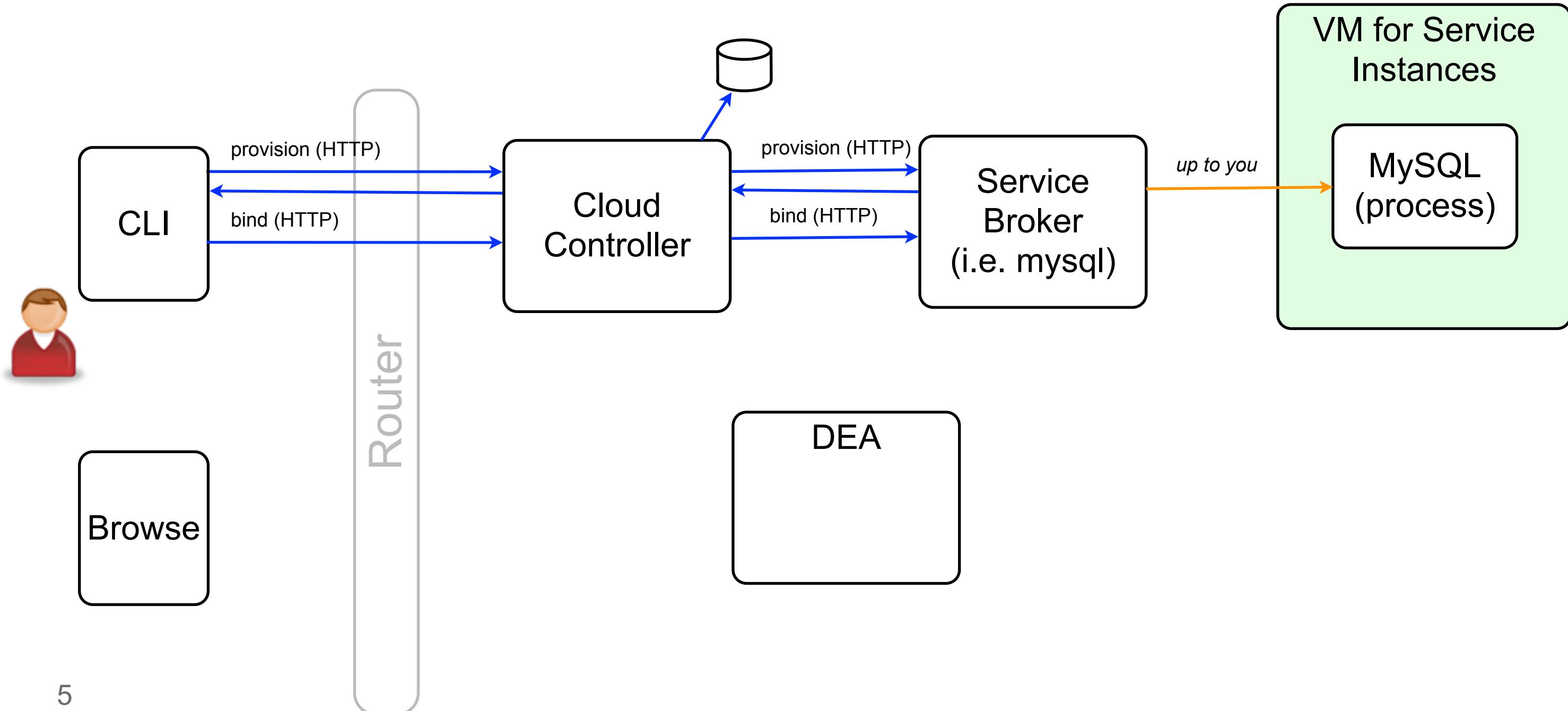
When you cf push



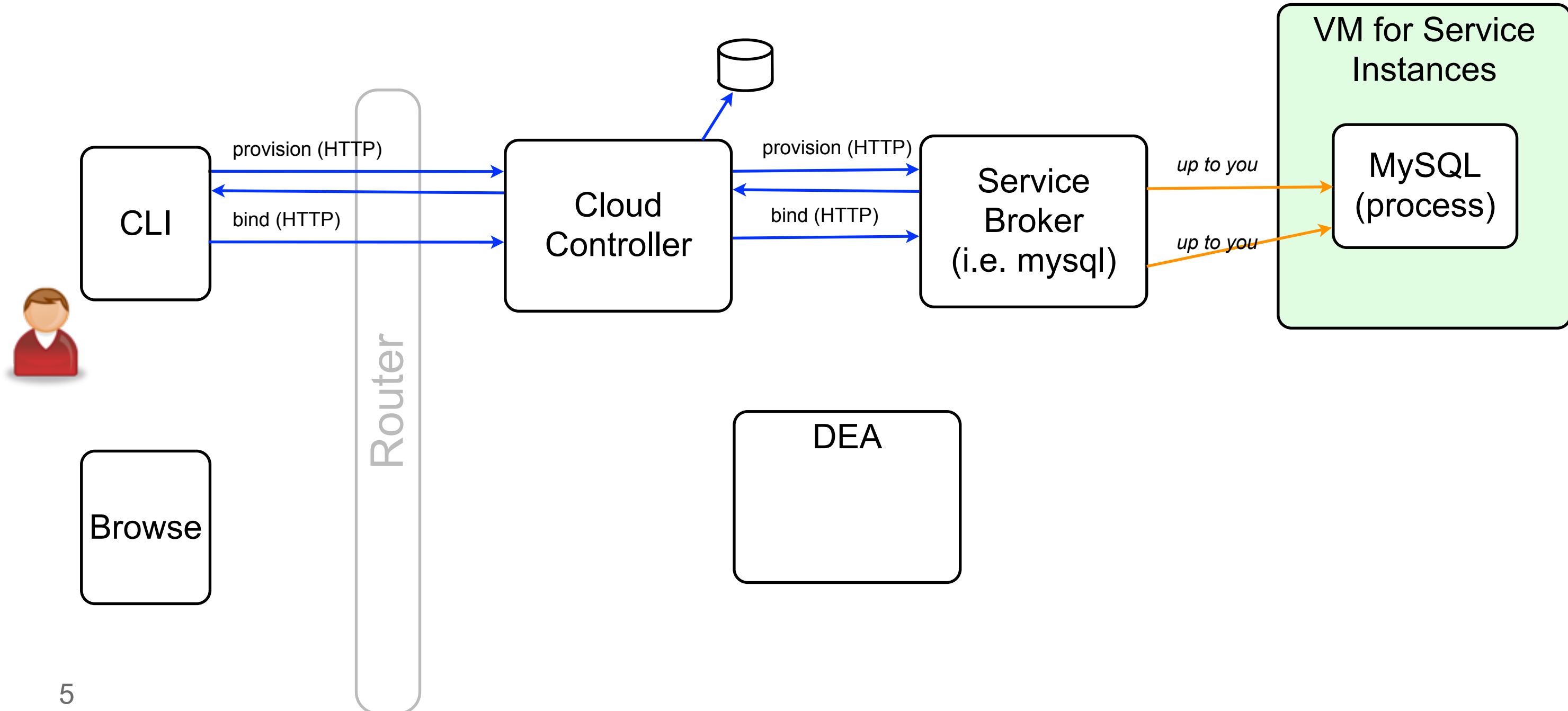
When you cf push



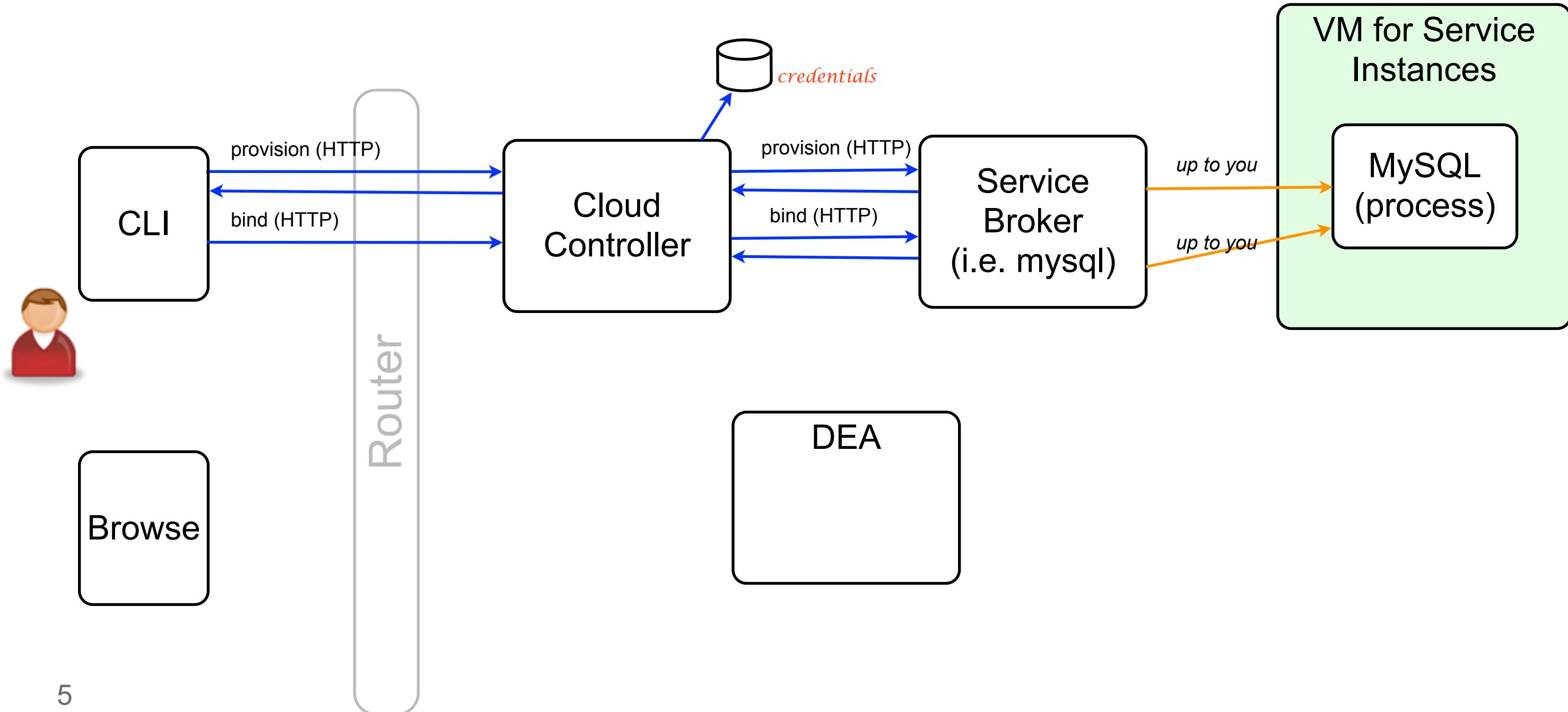
When you cf push



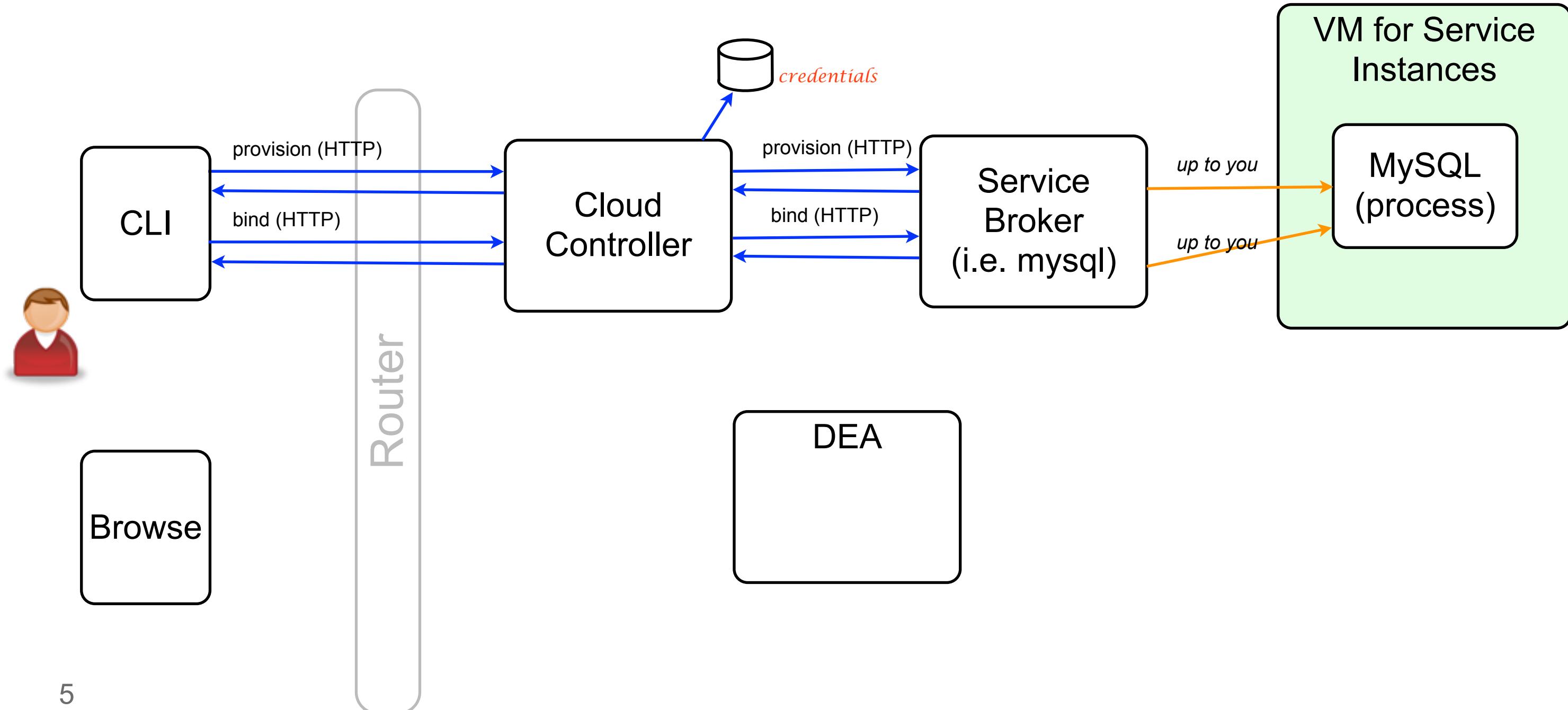
When you cf push



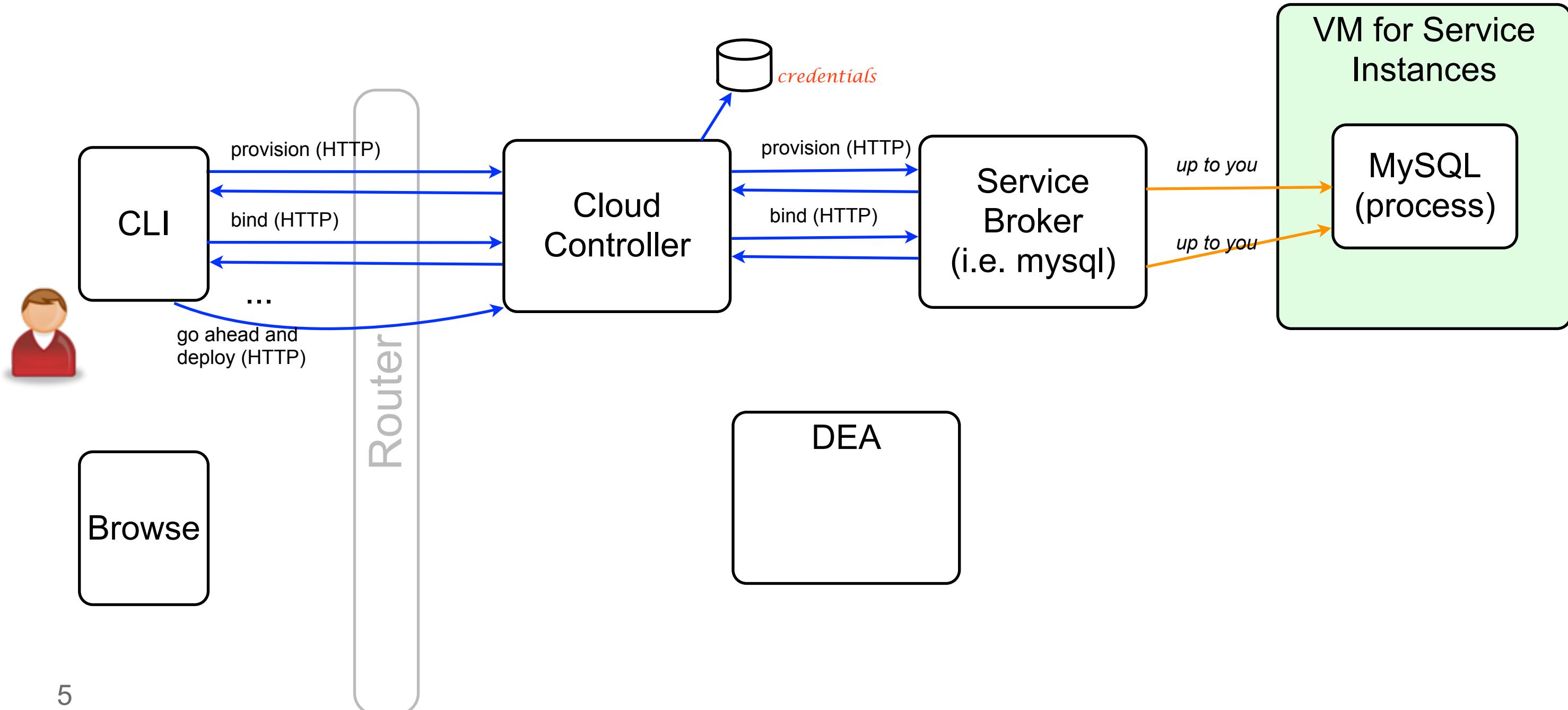
When you cf push



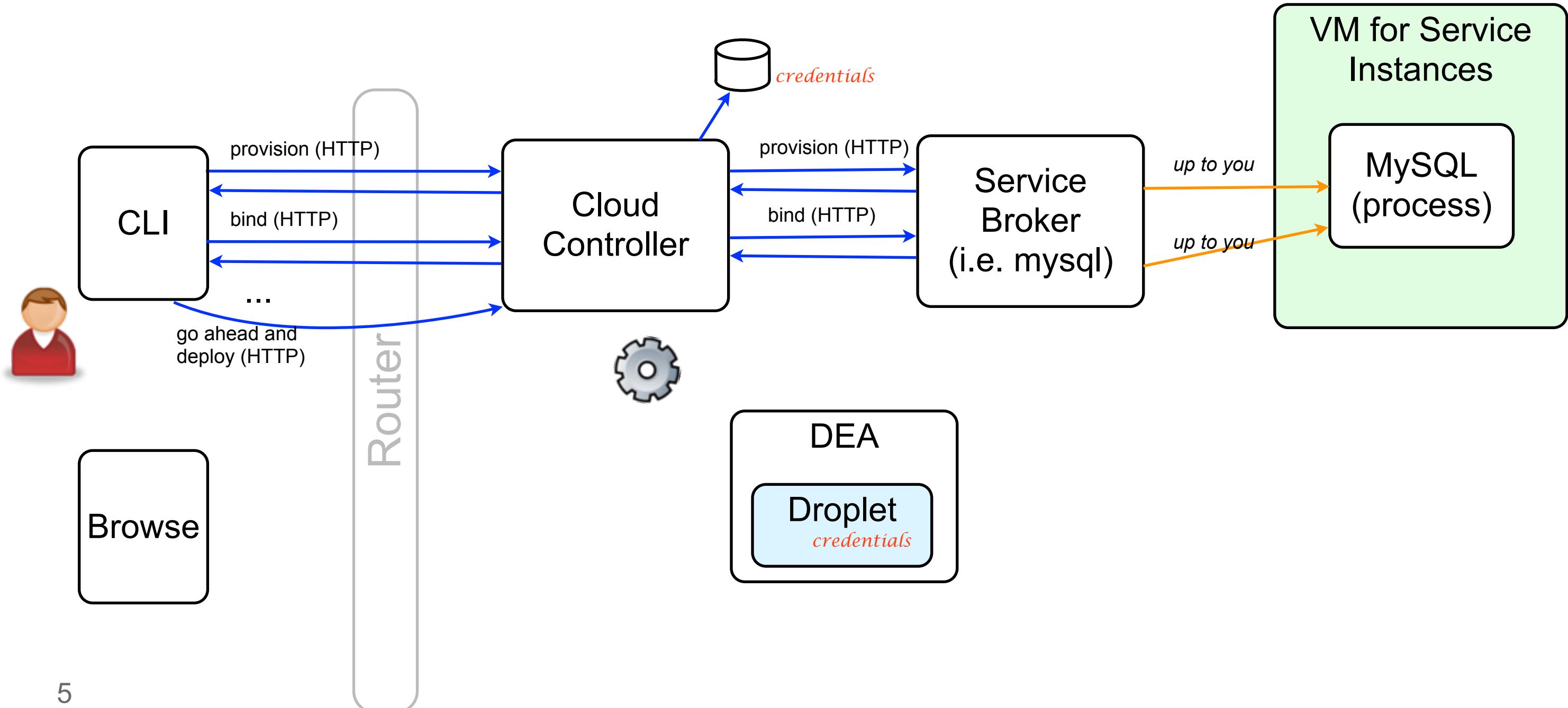
When you cf push



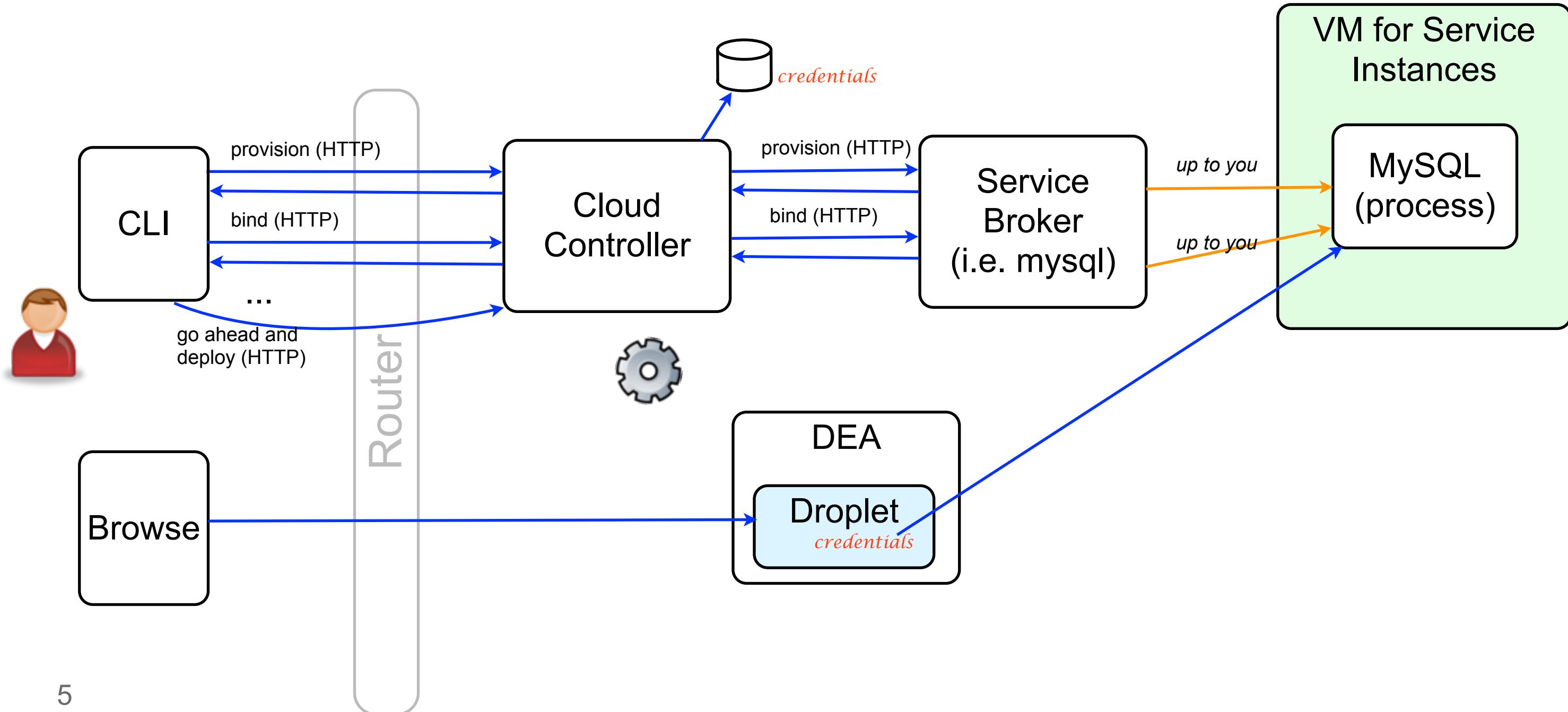
When you cf push



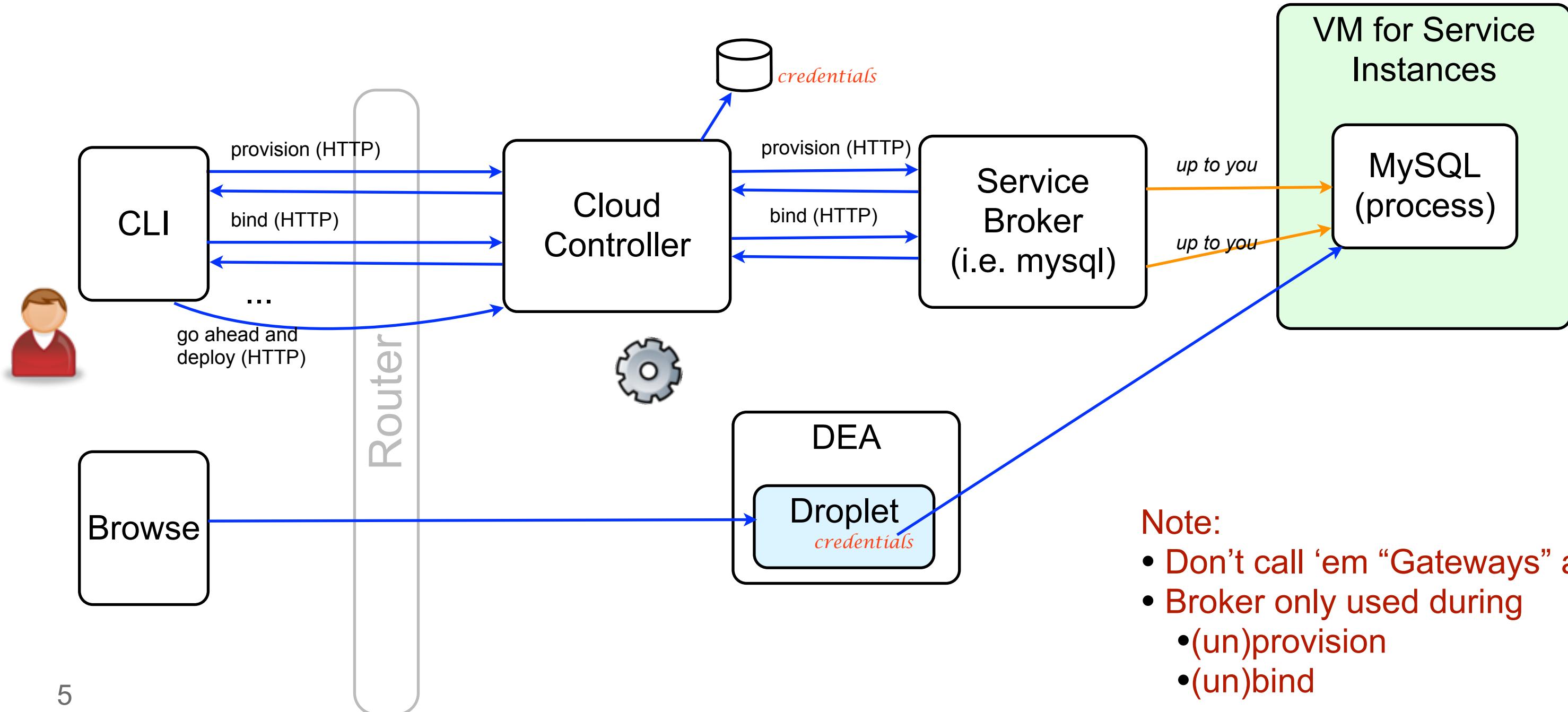
When you cf push



When you cf push



When you cf push



Demo

Agenda

- Extending services offerings within run.pivotal.io
 - Partnering
- Building services
 - Into your own Cloud Foundry
- User-provided service instances
 - For access to pre&externally-provisioned services

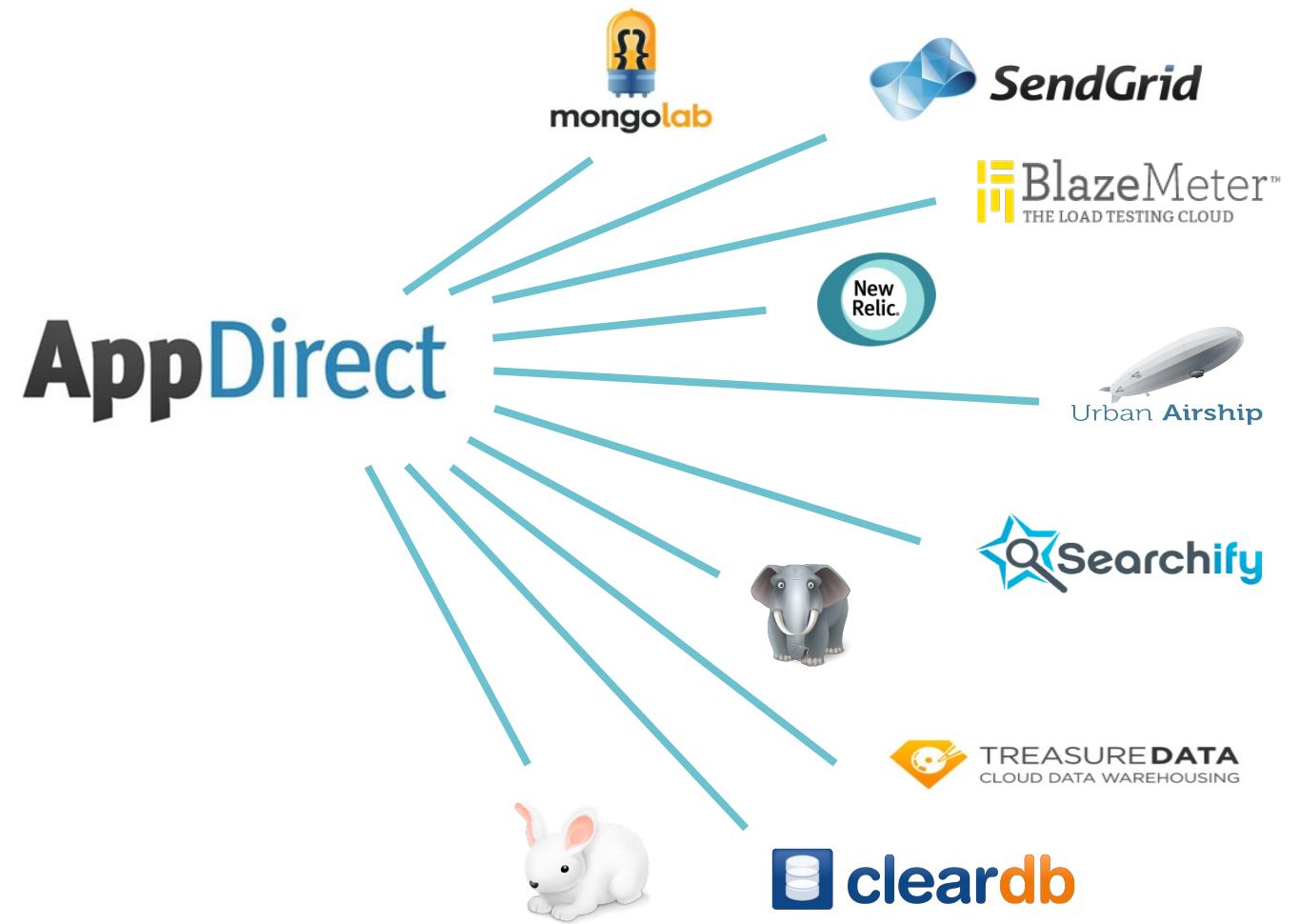
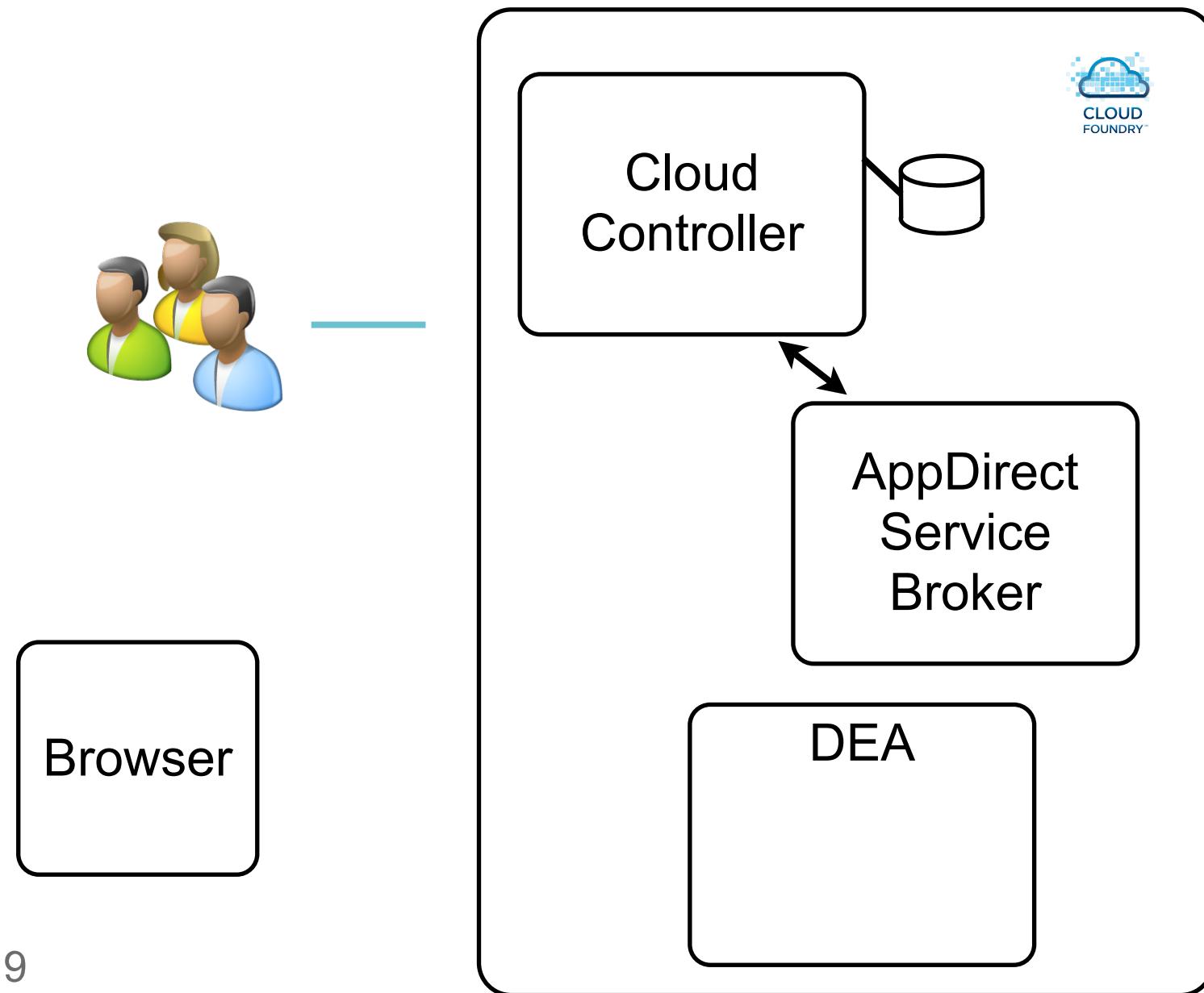
run.pivotal.io Marketplace Services



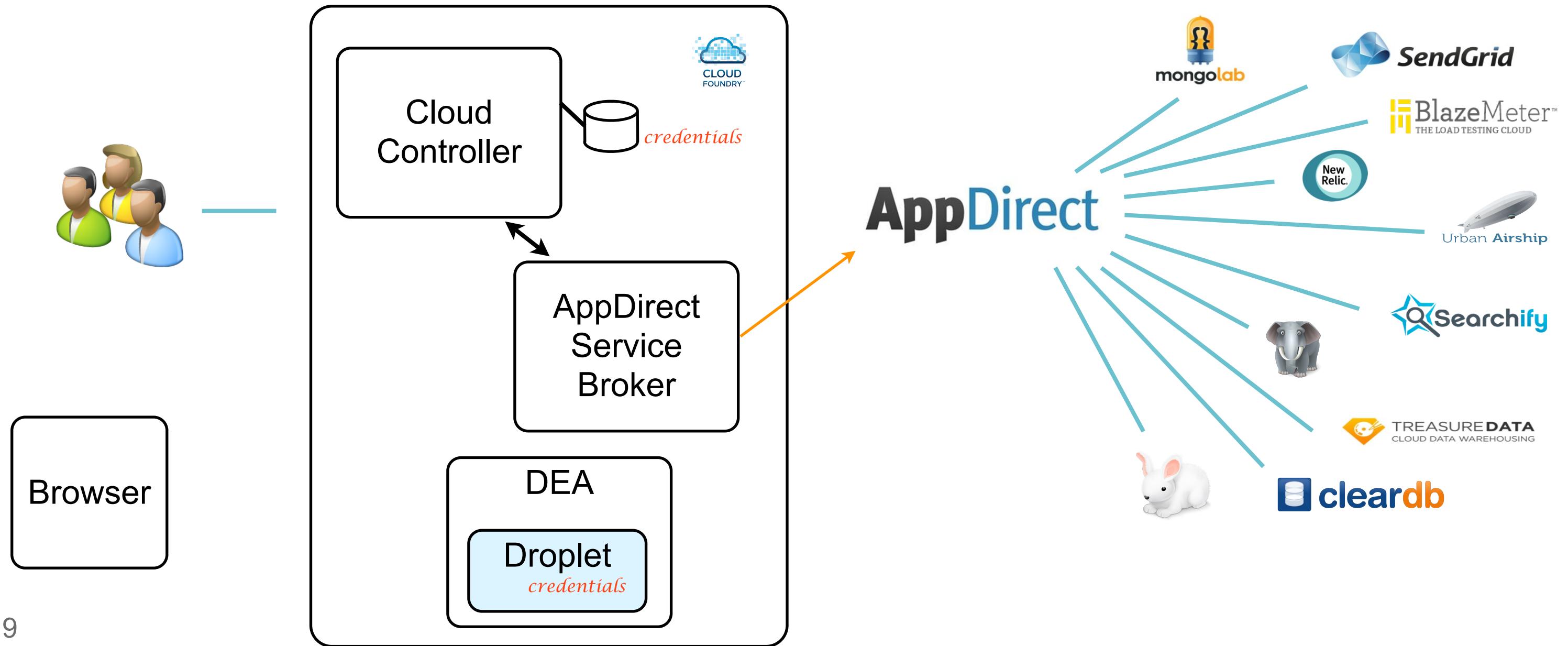
run.pivotal.io Marketplace Services



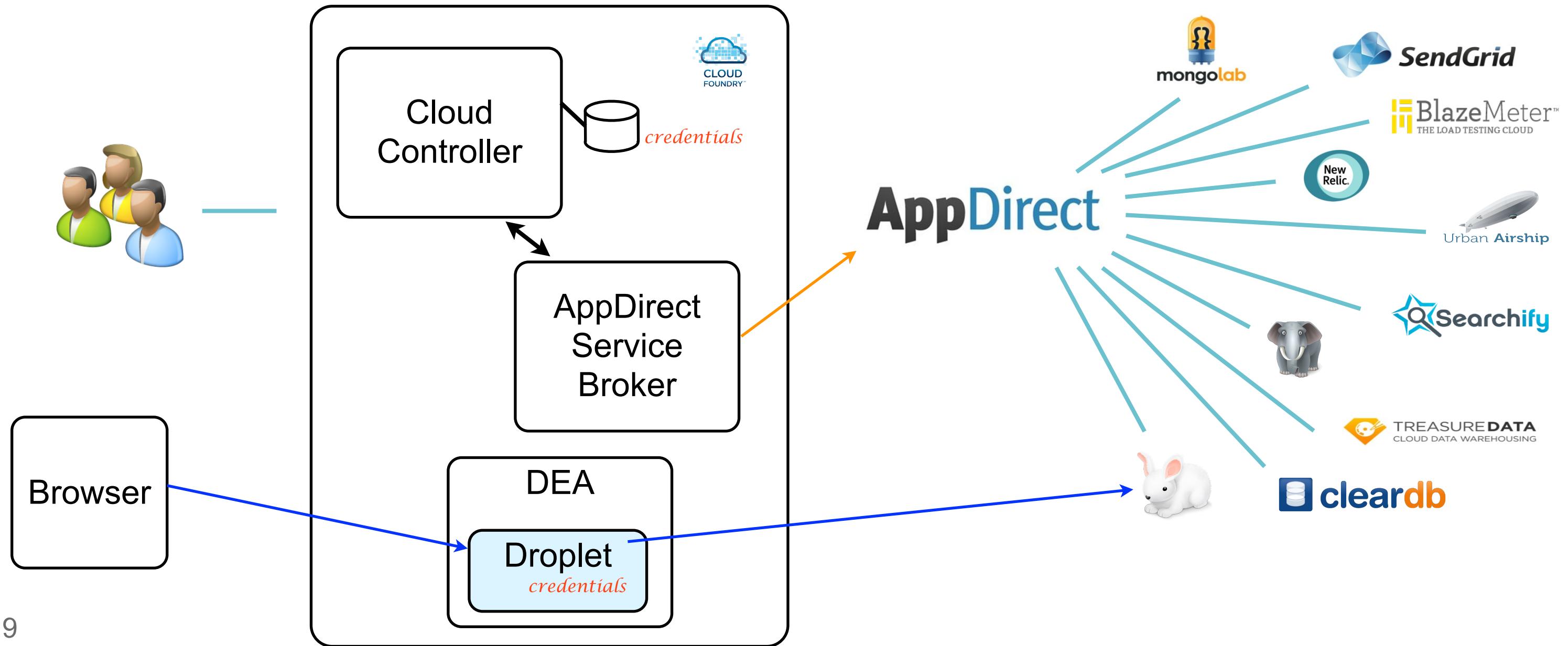
run.pivotal.io Marketplace Services



run.pivotal.io Marketplace Services



run.pivotal.io Marketplace Services



Demo

Cloud Foundry Services

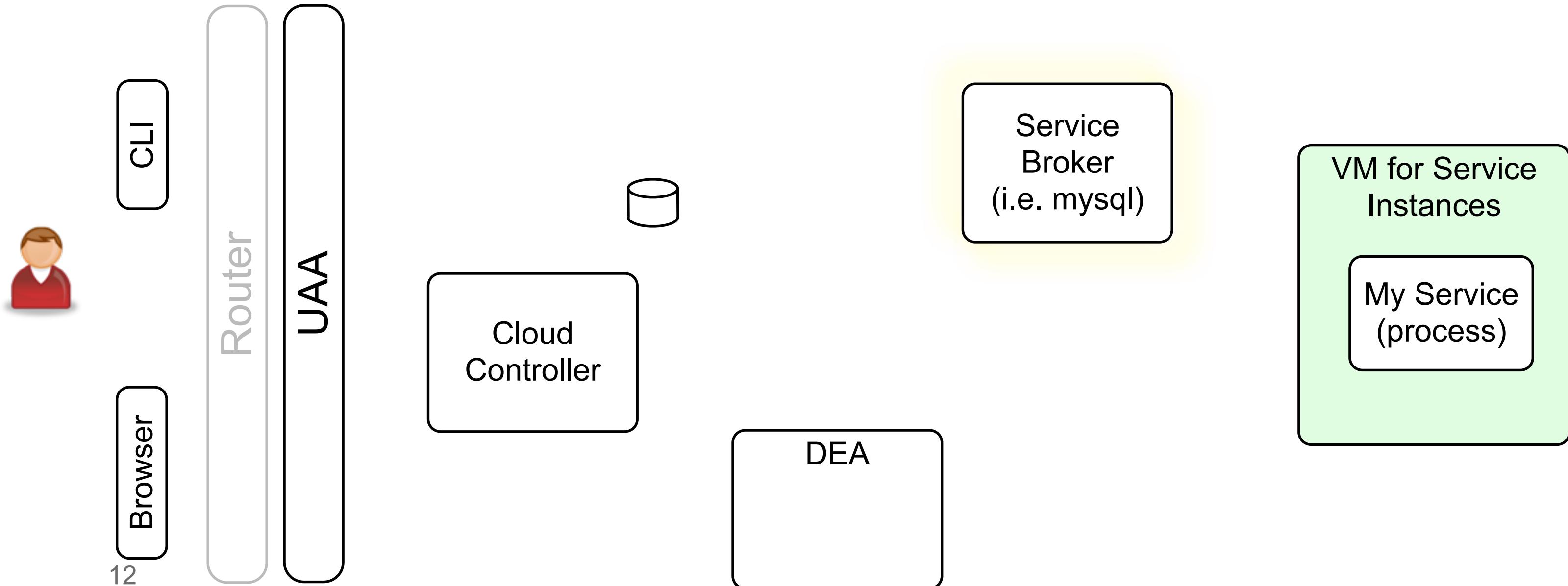
- There is only one built in - mysql  [cloudfoundry / cf-services-release](#)
 - or many more depending on how you count  [cloudfoundry / cf-services-contrib-release](#)
- How to build them - the evolution:
 - CF v1, subclass Ruby base classes
 - CF v2, a RESTful protocol (which is itself evolving)

Polyglot!

- There are three parts to the protocol...

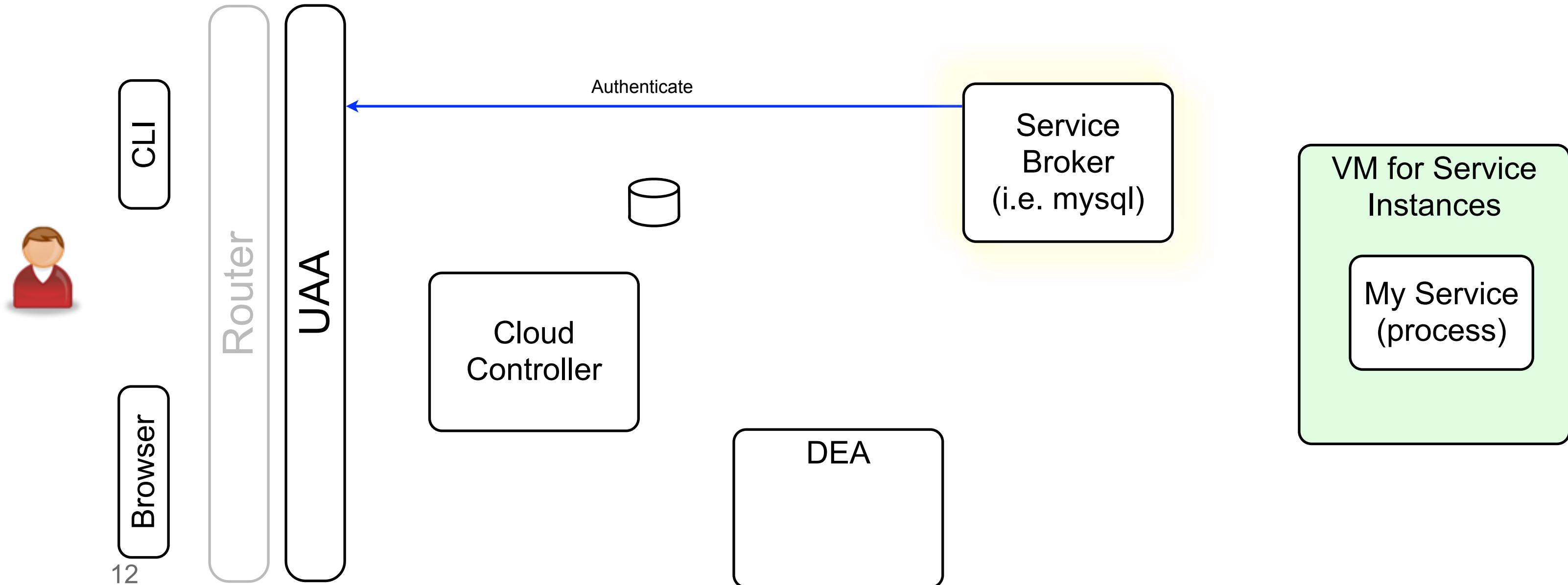
① Broker Registers Offerings

On Broker Startup...



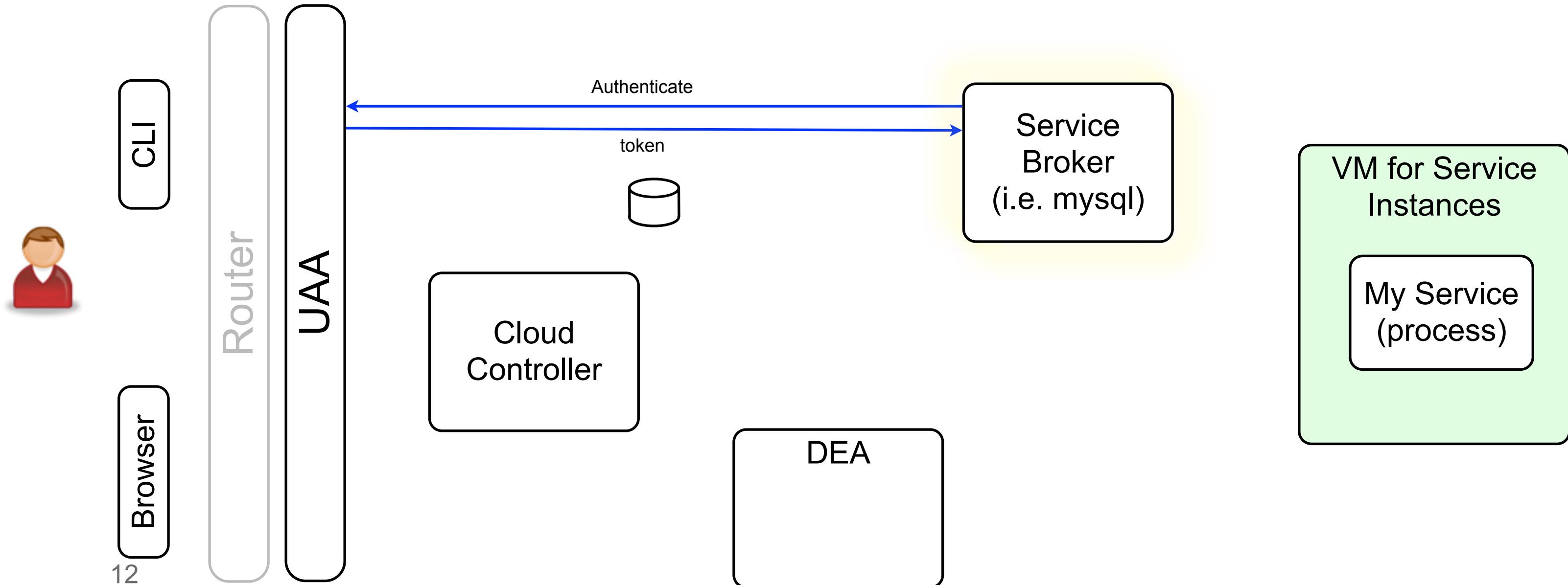
① Broker Registers Offerings

On Broker Startup...



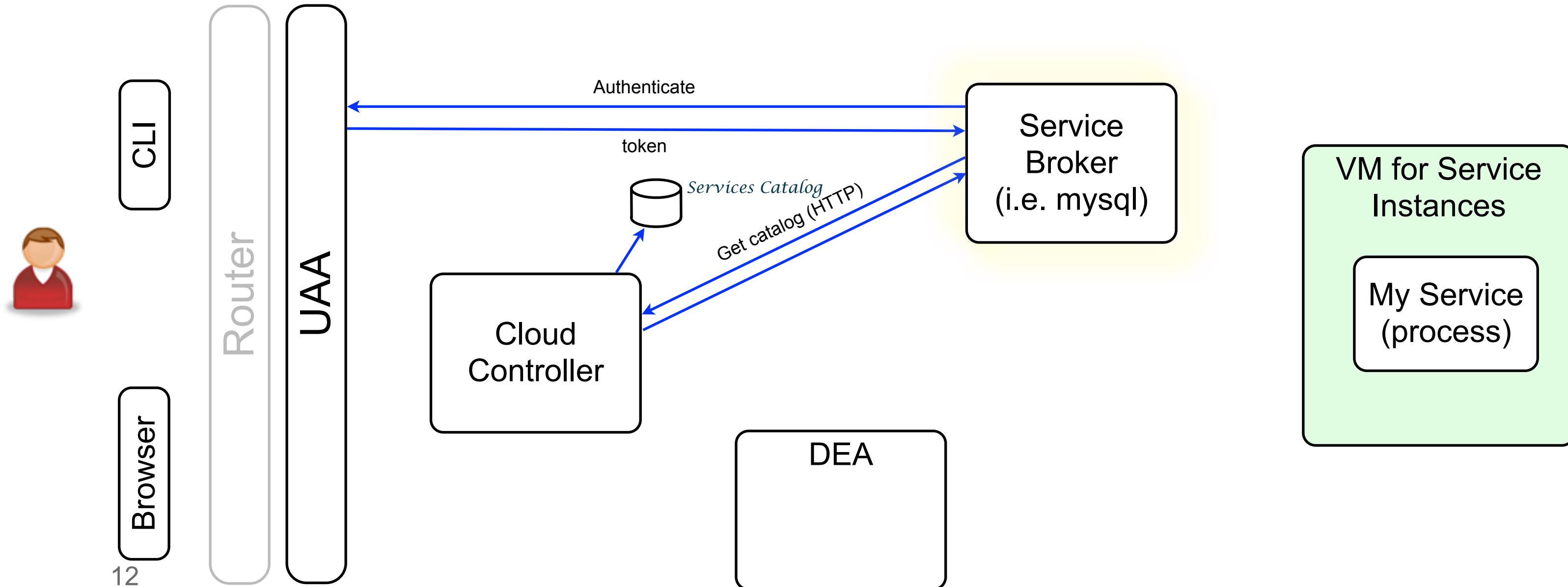
① Broker Registers Offerings

On Broker Startup...



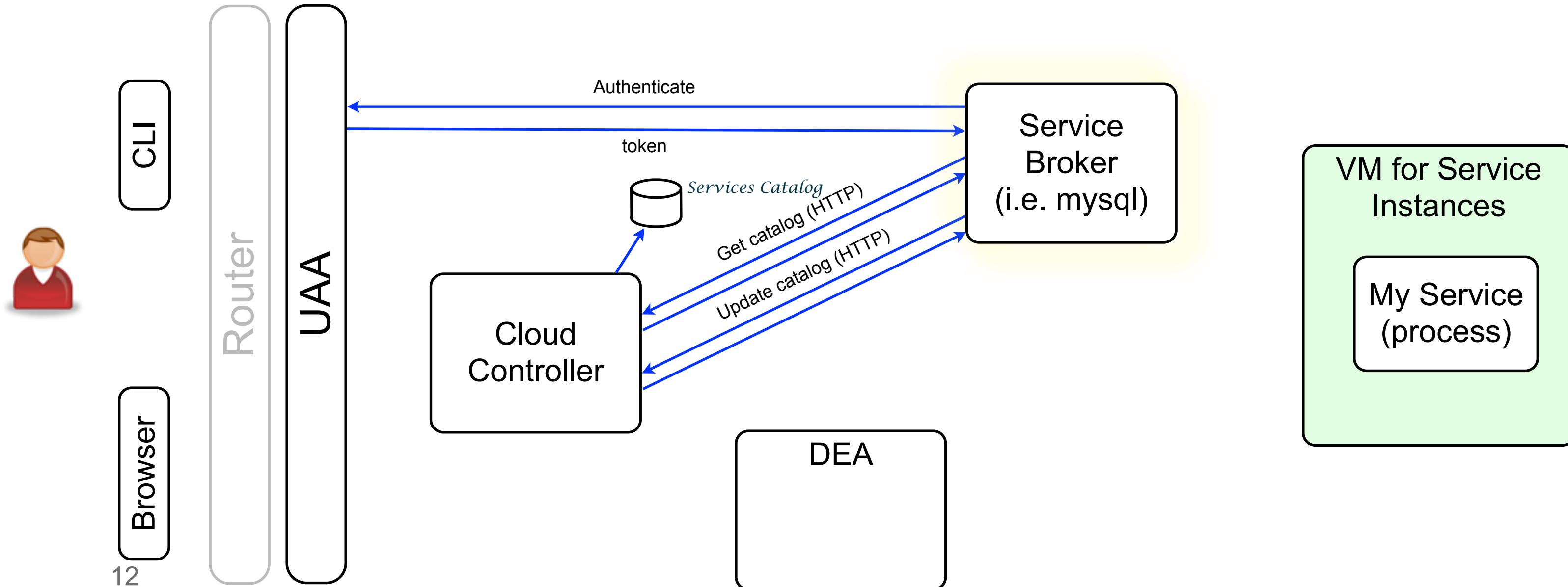
① Broker Registers Offerings

On Broker Startup...



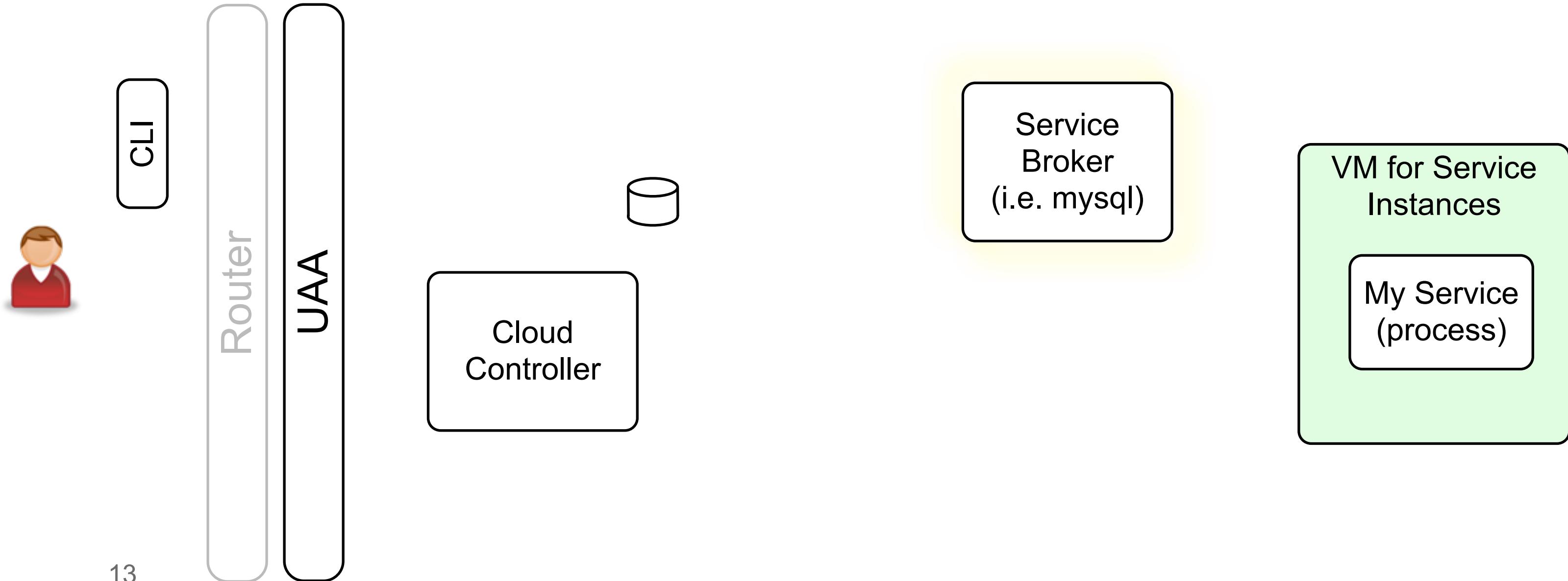
① Broker Registers Offerings

On Broker Startup...



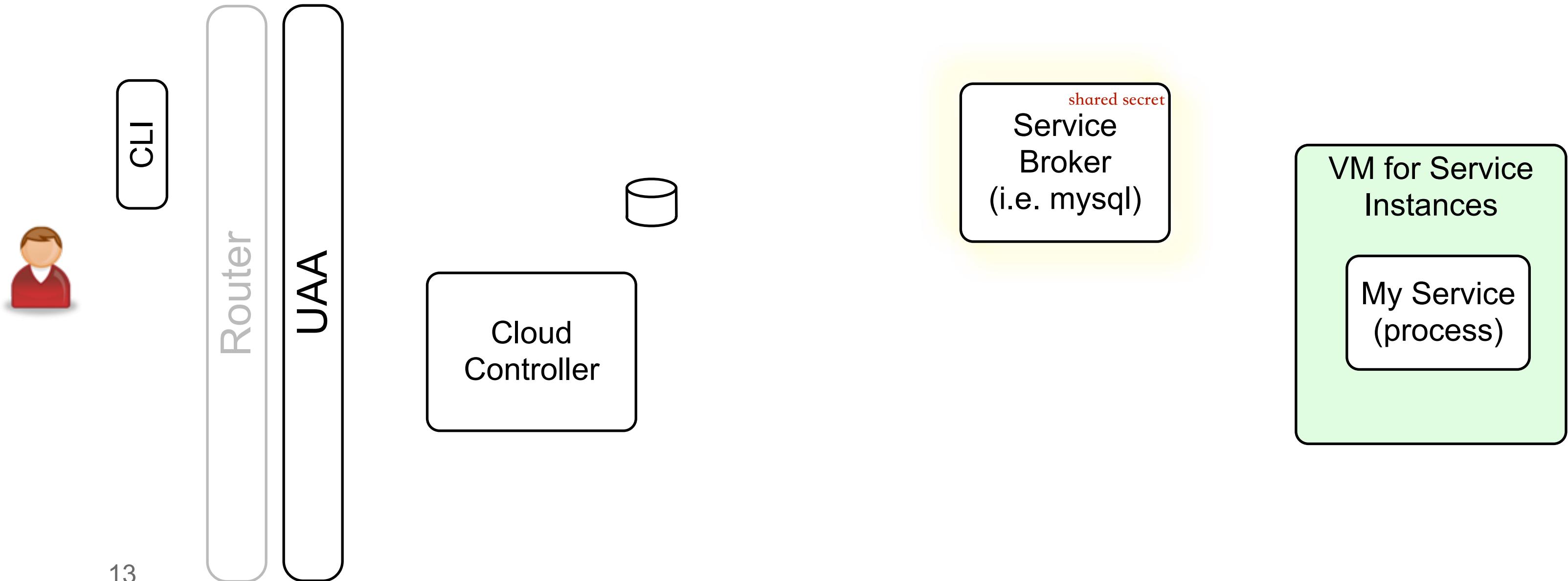
② Broker Services (un)provision and (un)bind

Prerequisite...



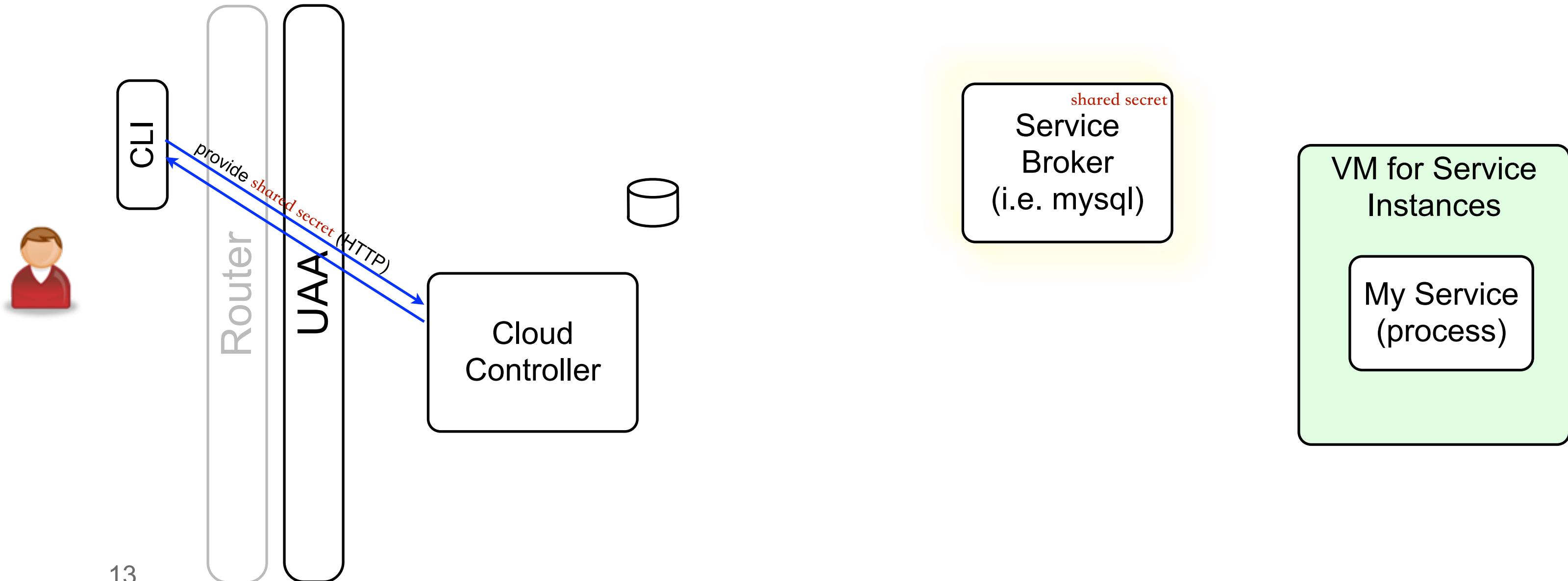
② Broker Services (un)provision and (un)bind

Prerequisite...

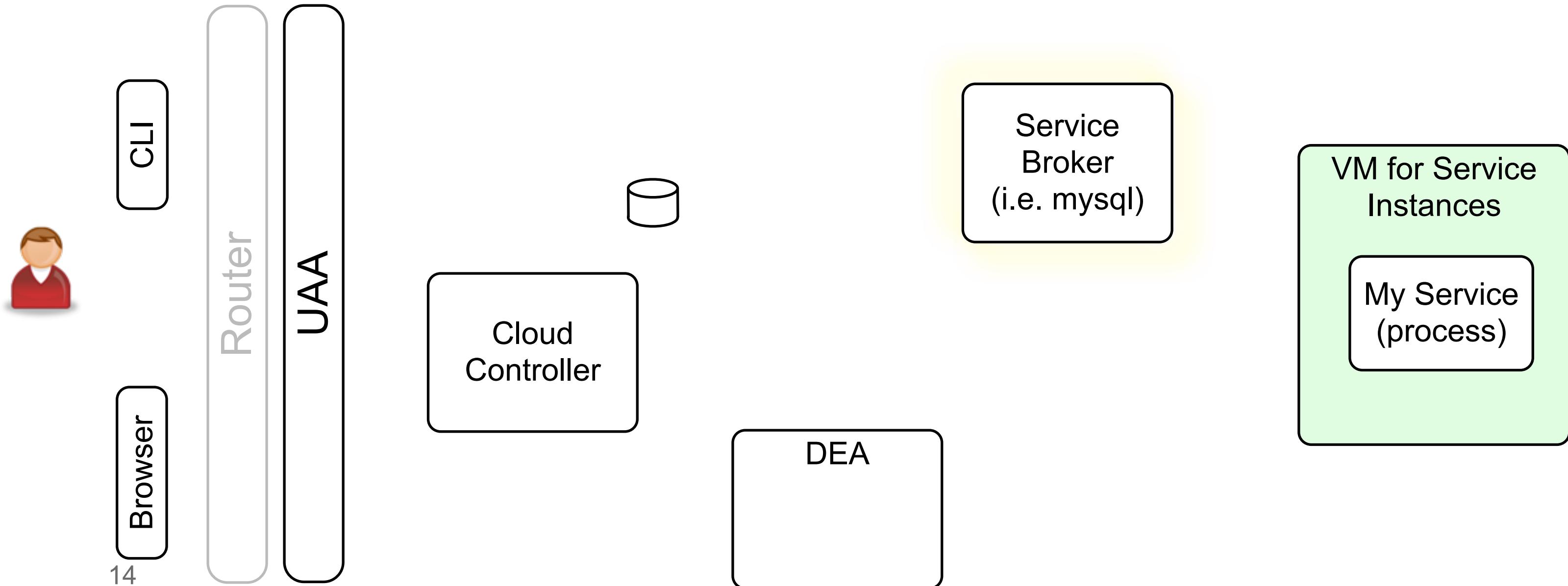


② Broker Services (un)provision and (un)bind

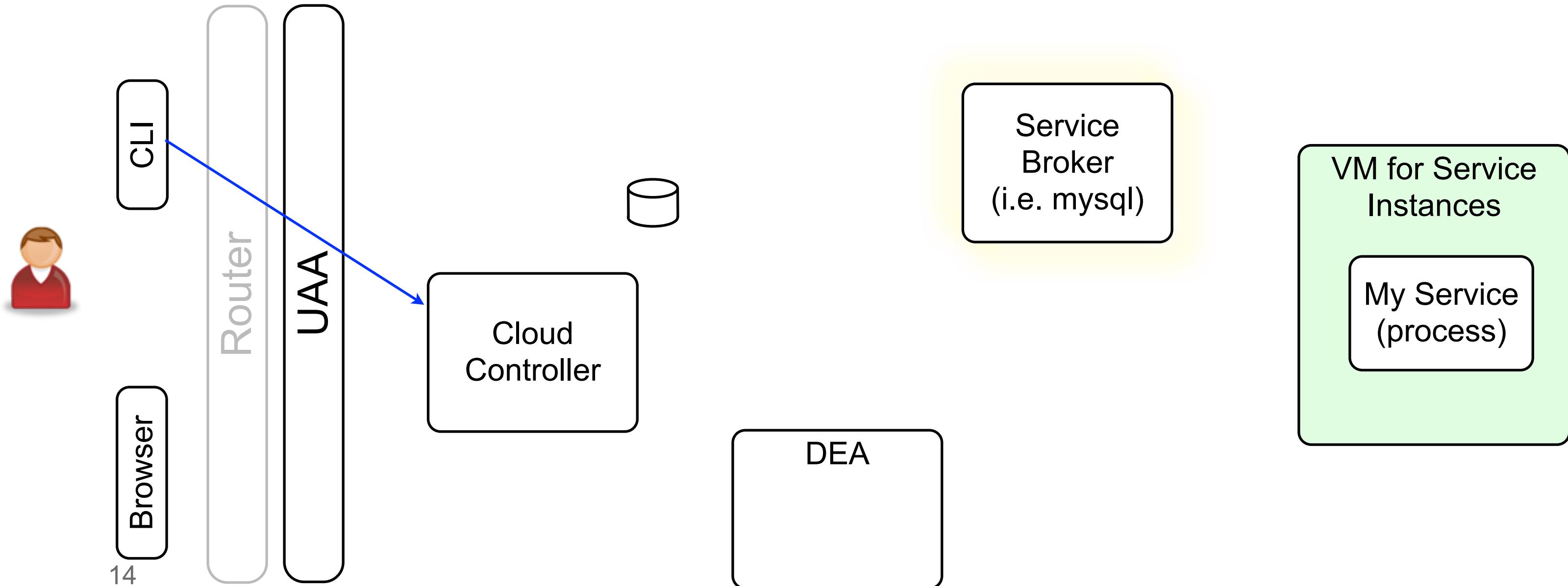
Prerequisite...



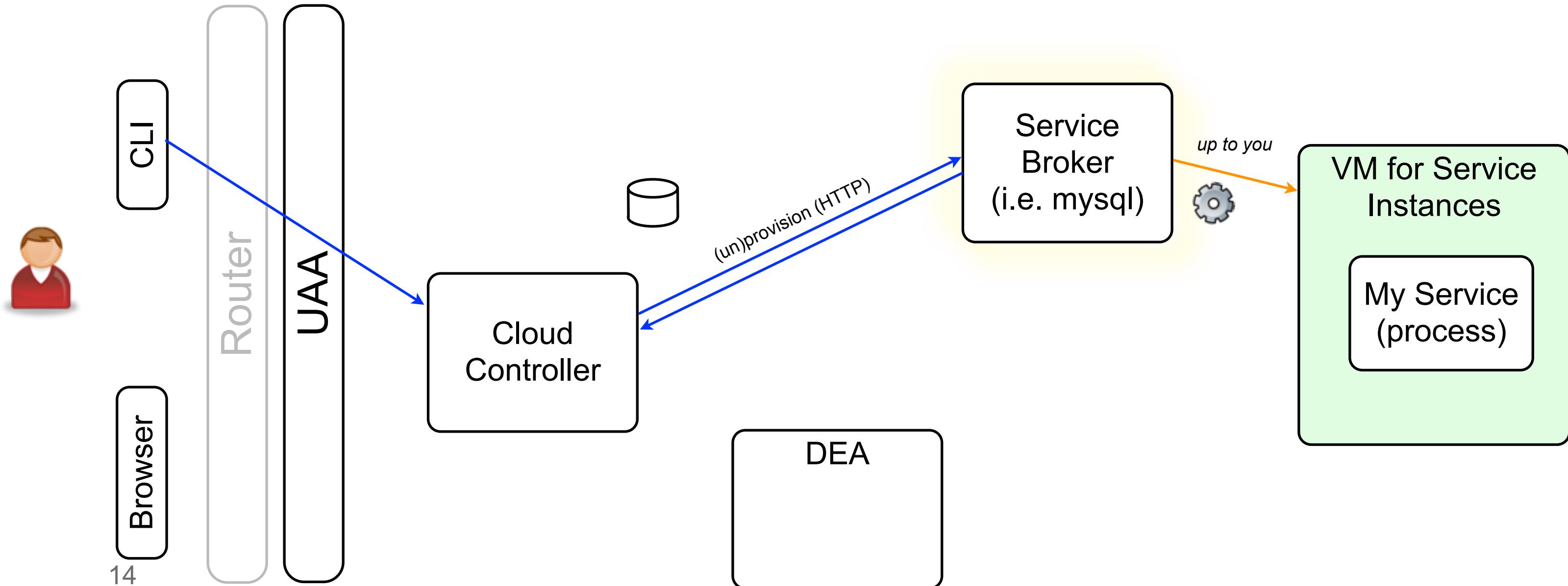
② Broker Services (un)provision and (un)bind



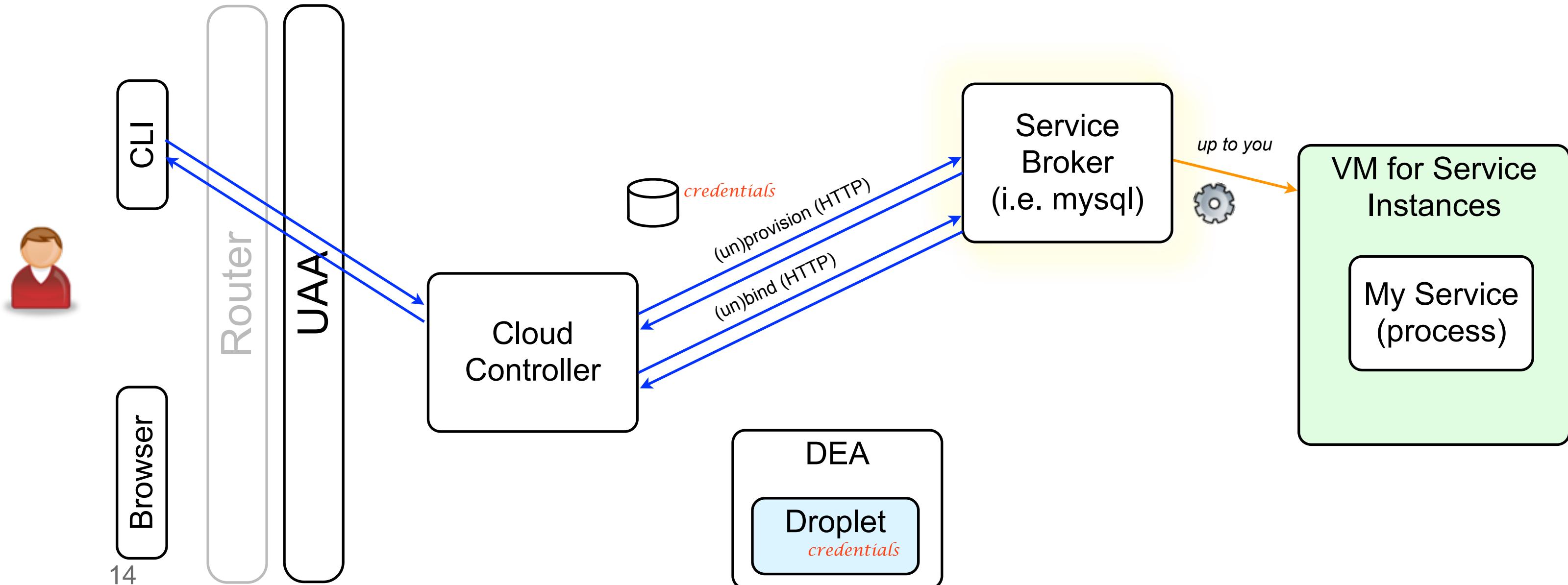
② Broker Services (un)provision and (un)bind



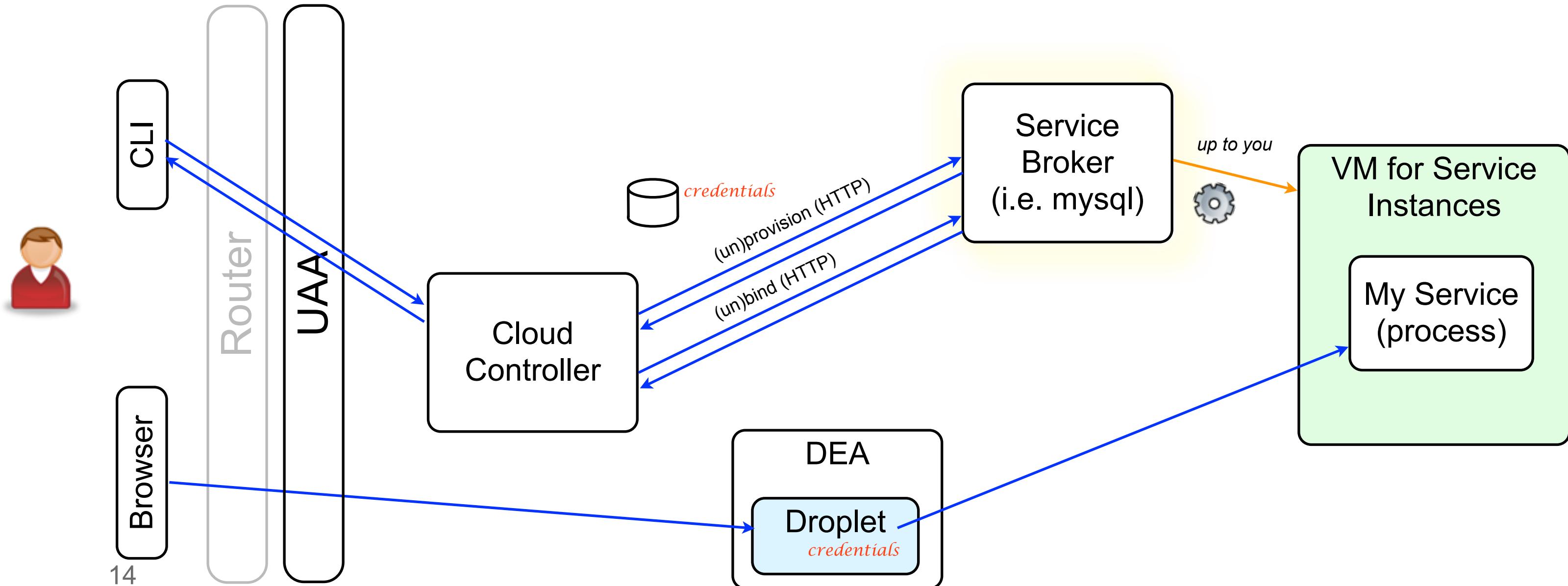
② Broker Services (un)provision and (un)bind



② Broker Services (un)provision and (un)bind

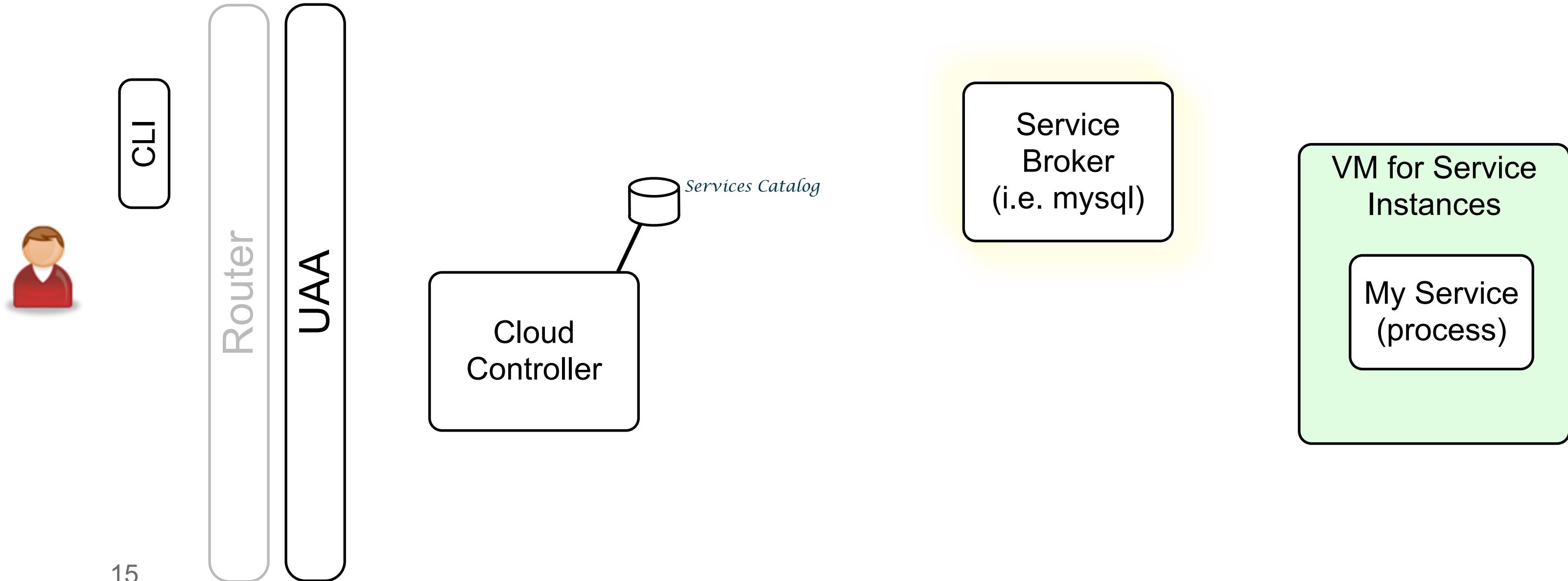


② Broker Services (un)provision and (un)bind



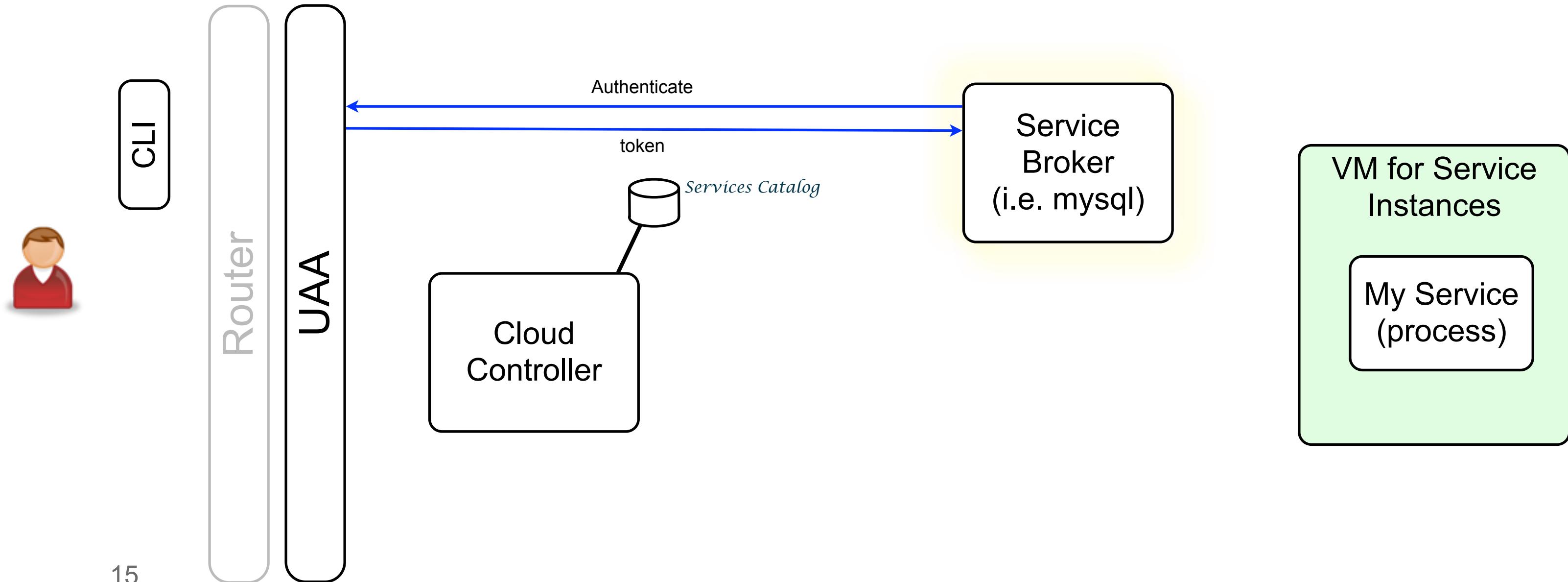
③ Orphan Management

Broker is responsible for eventual consistency...



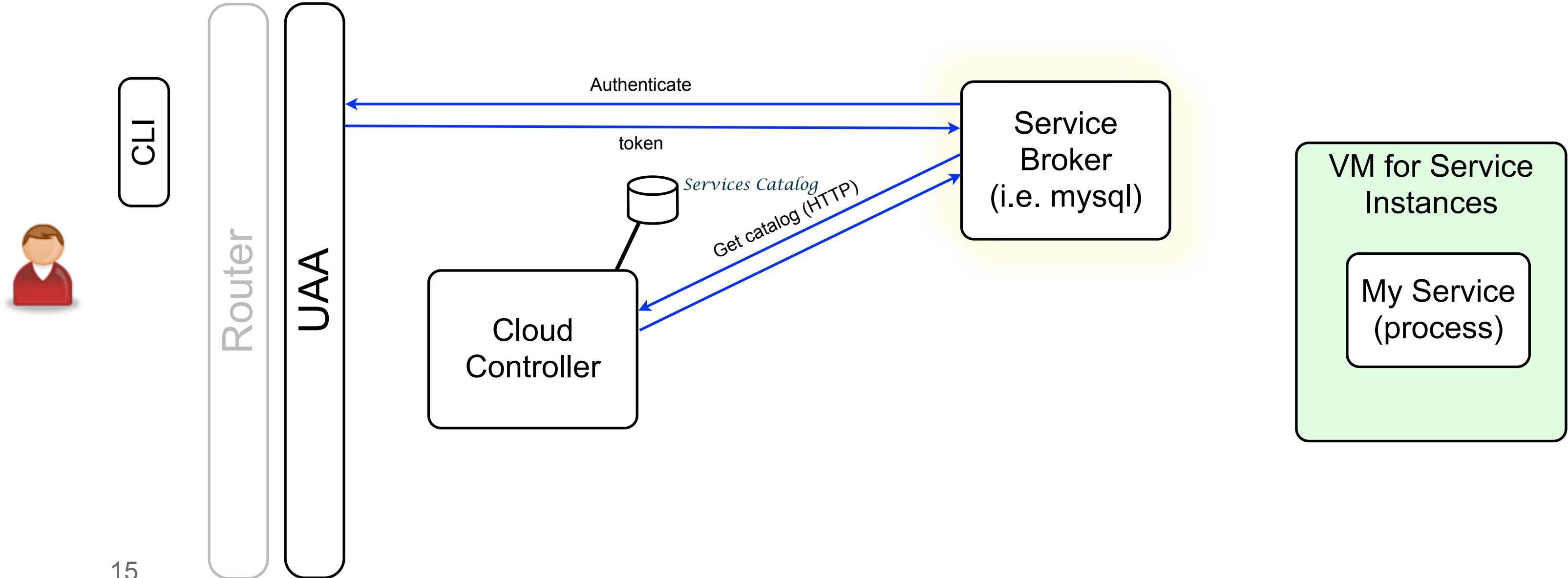
③ Orphan Management

Broker is responsible for eventual consistency...



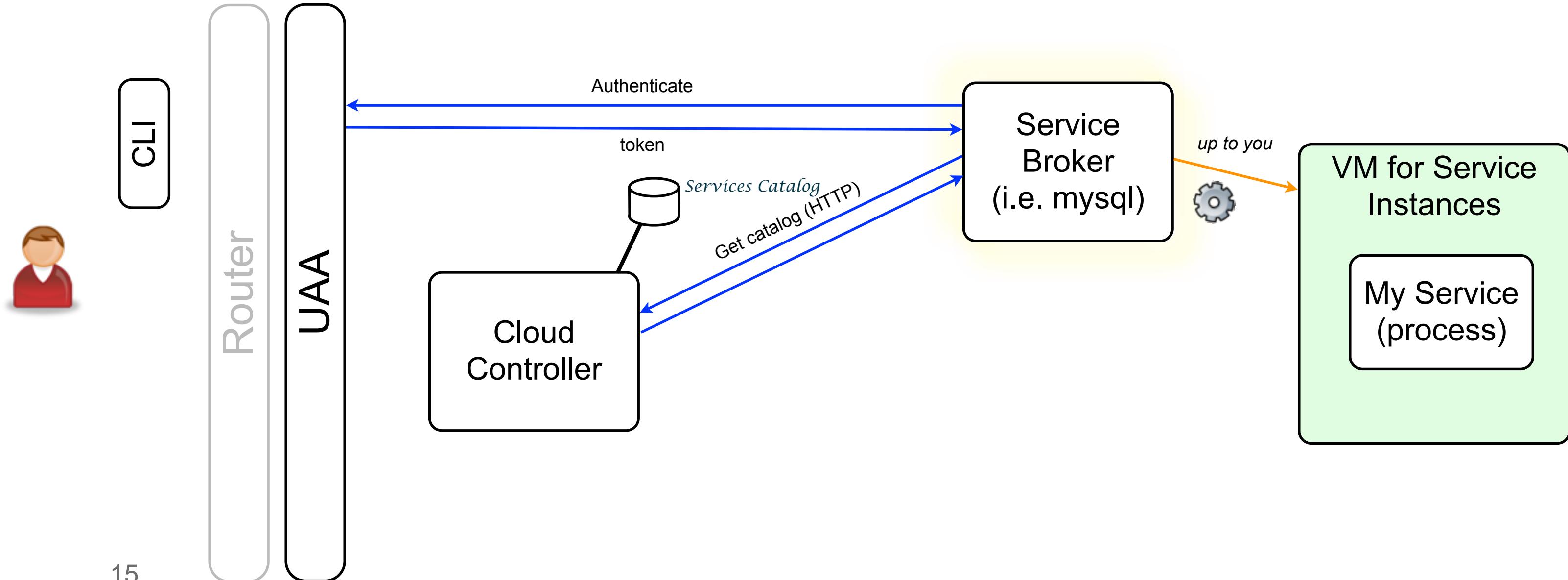
③ Orphan Management

Broker is responsible for eventual consistency...



③ Orphan Management

Broker is responsible for eventual consistency...

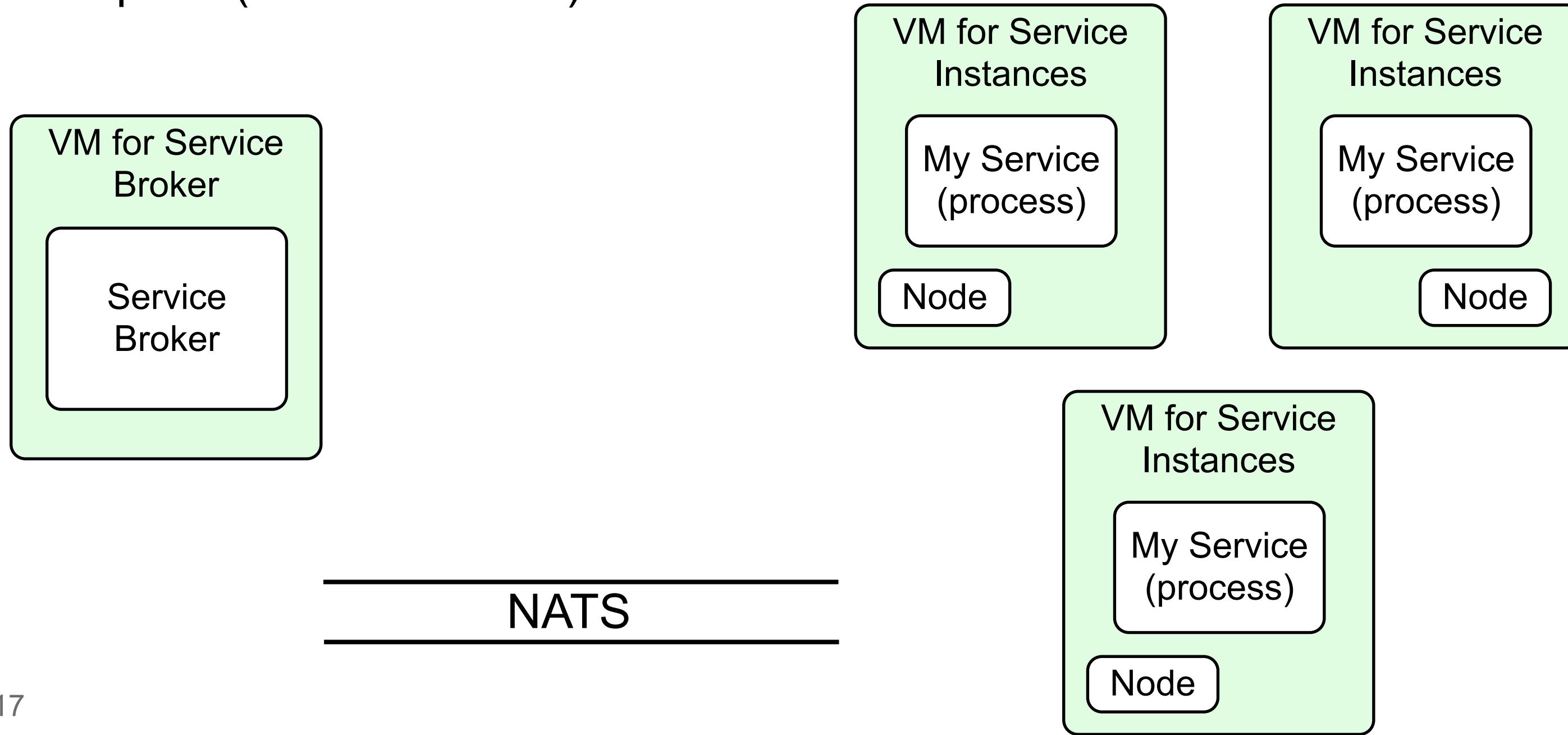


New (V2) Services API

- Simplifies things a bunch!
- Unidirectional - everything initiated from Cloud Controller
- Asynchronous provisioning
- From V1 RESTful Services API:
 - Replace offerings registration with endpoint that allows the CC to GET the offerings
 - Remove orphan management

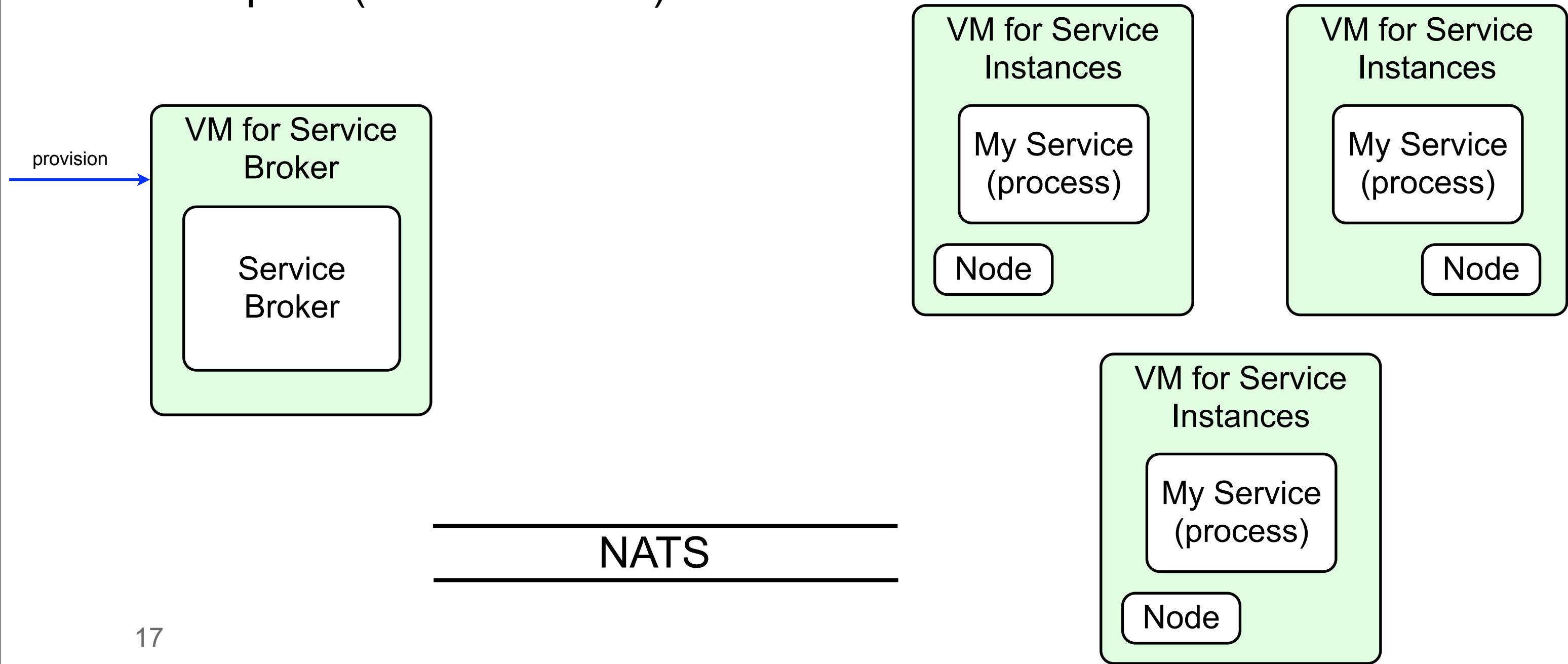
It's Up To YOU Patterns

One option (CF v1 services)...



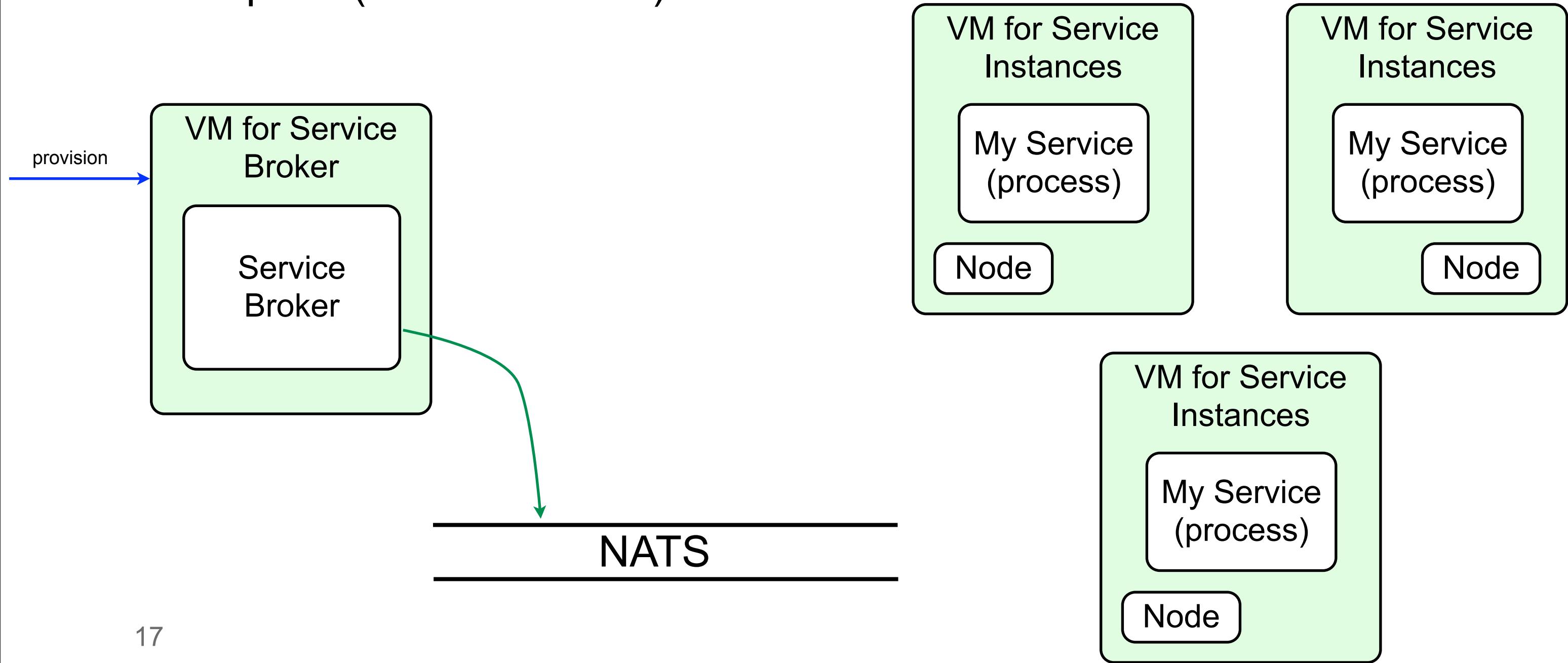
It's Up To YOU Patterns

One option (CF v1 services)...



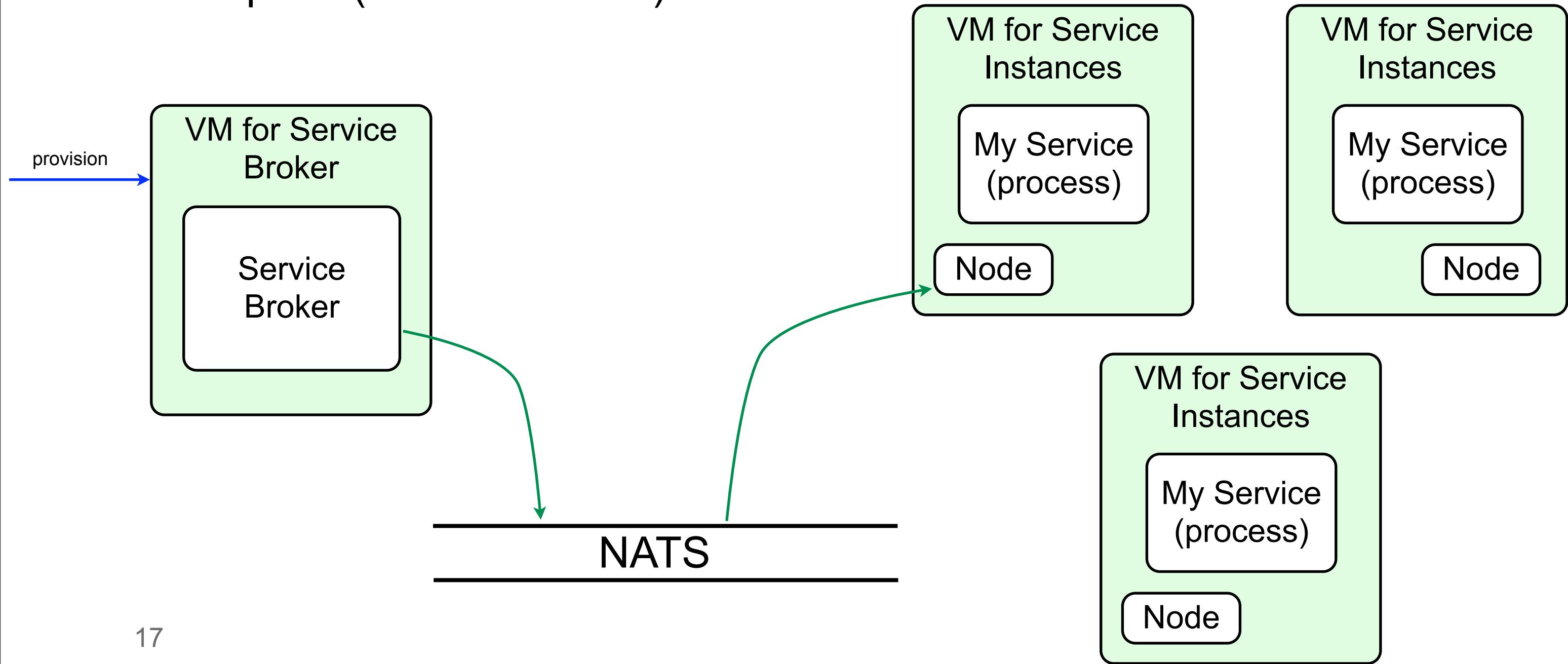
It's Up To YOU Patterns

One option (CF v1 services)...



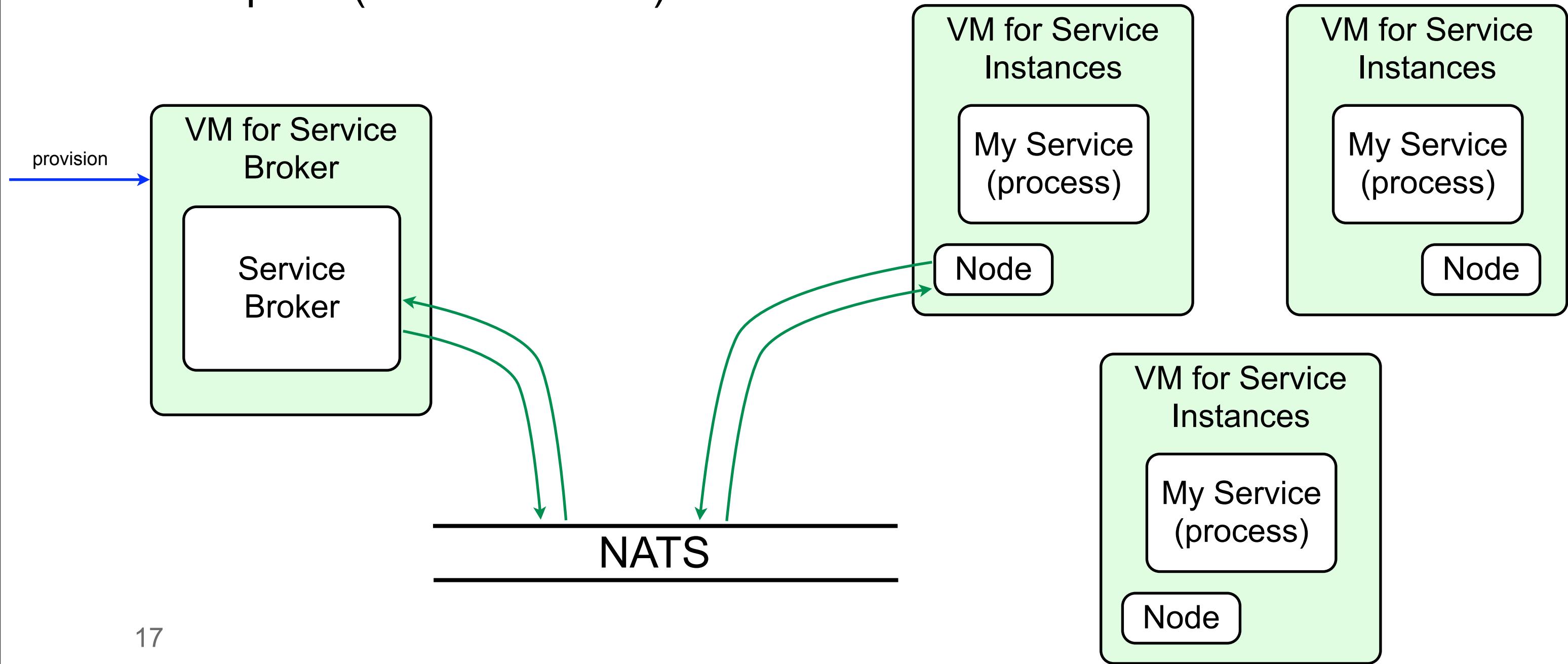
It's Up To YOU Patterns

One option (CF v1 services)...



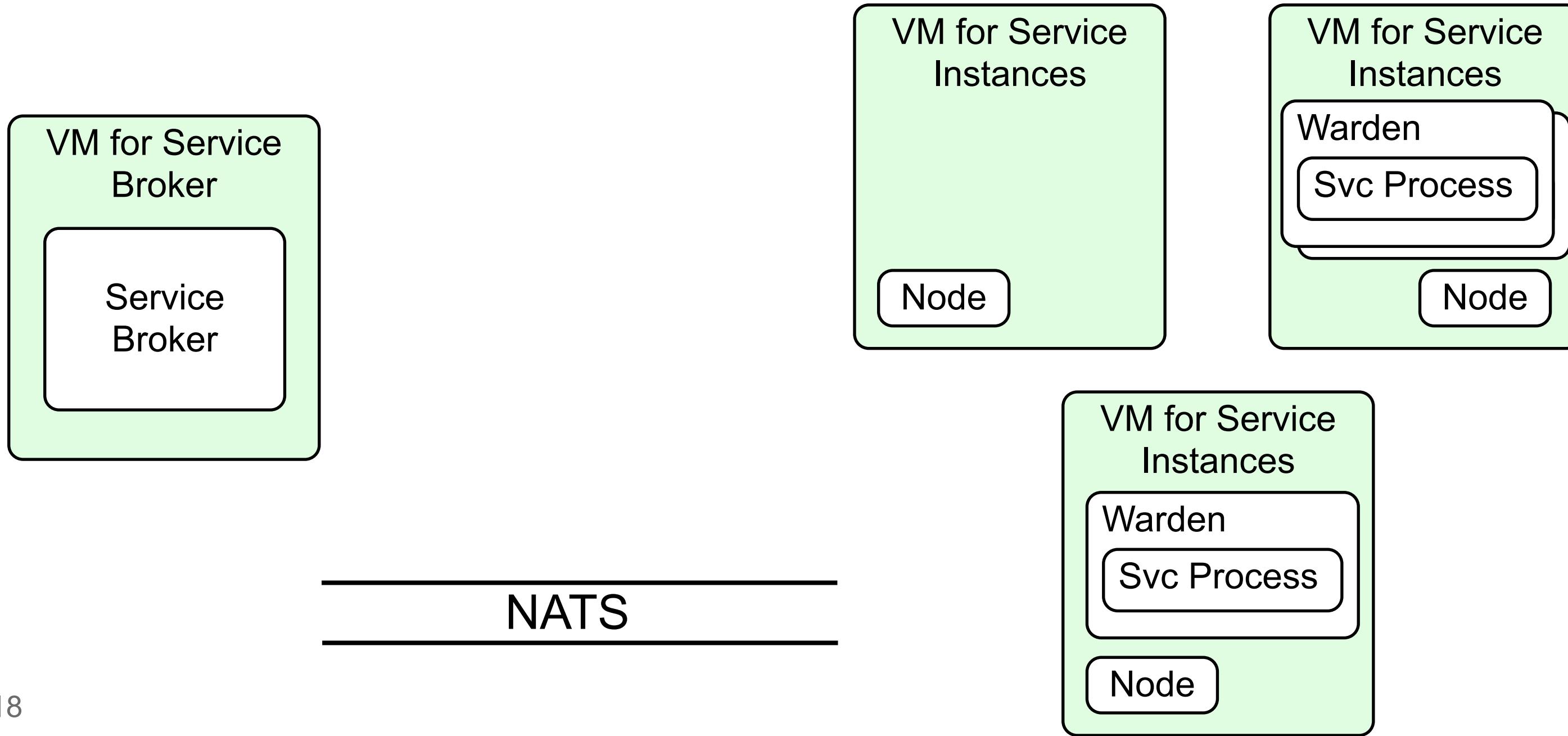
It's Up To YOU Patterns

One option (CF v1 services)...



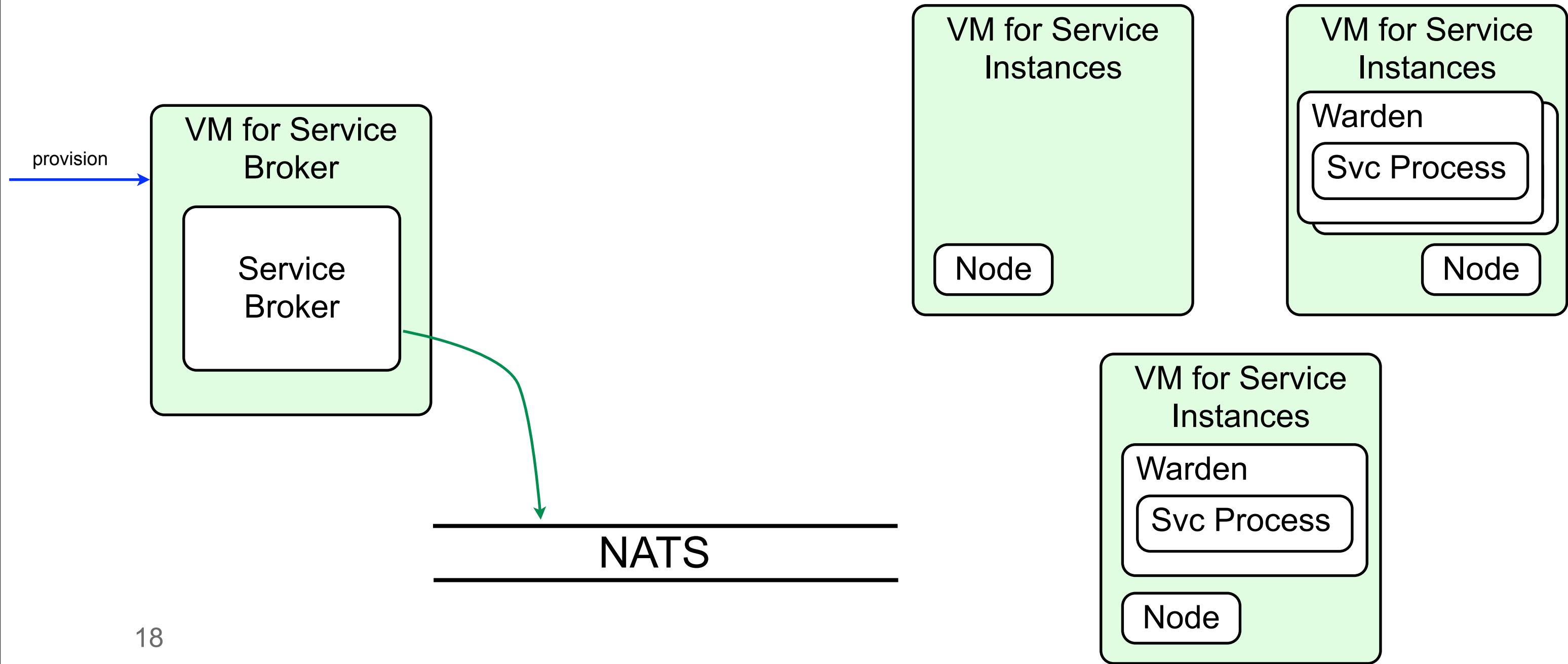
It's Up To YOU Patterns

Another...



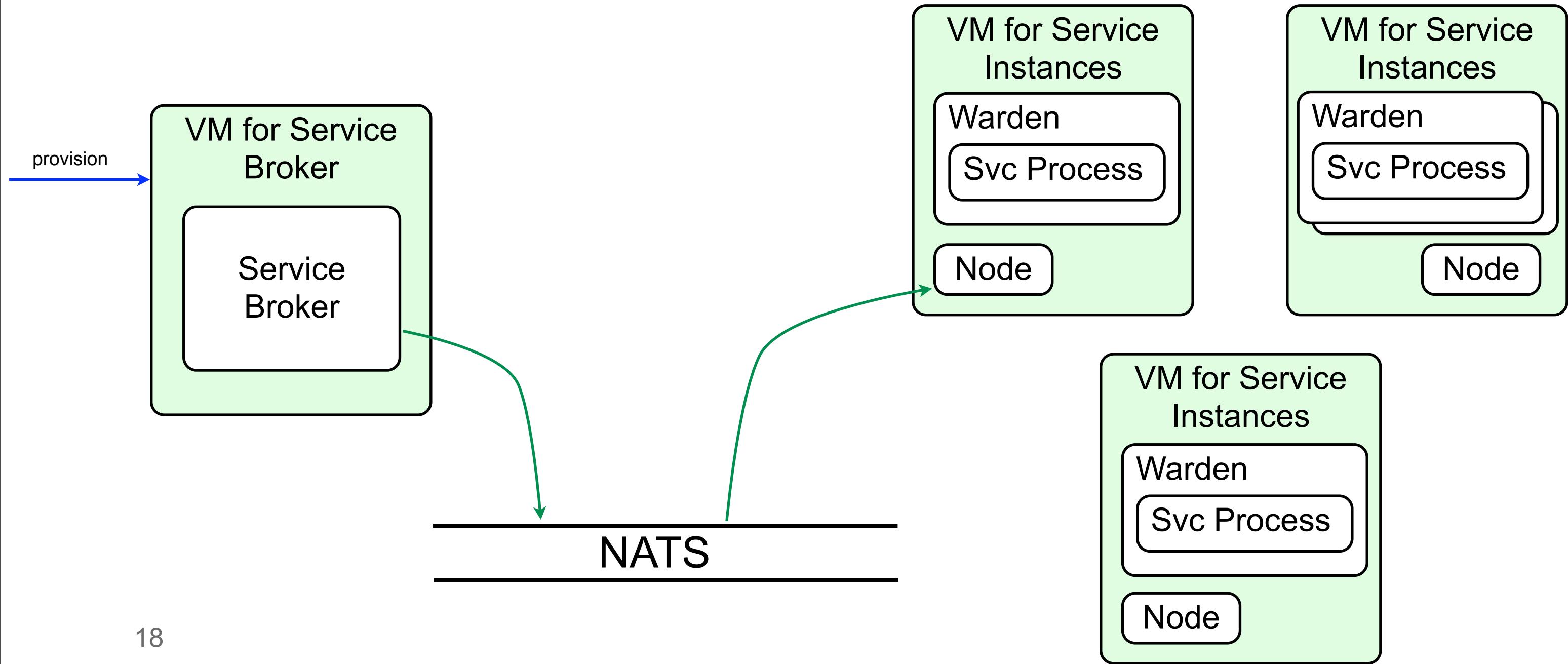
It's Up To YOU Patterns

Another...



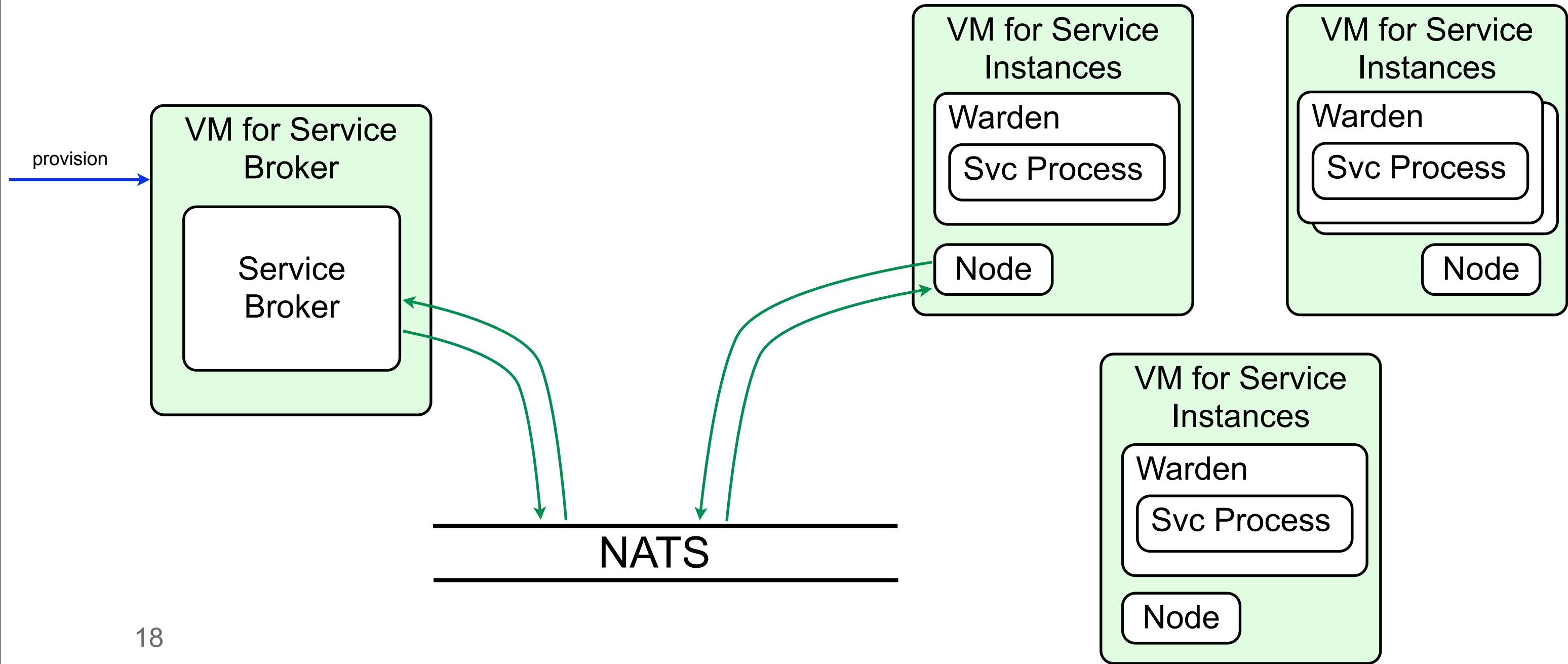
It's Up To YOU Patterns

Another...



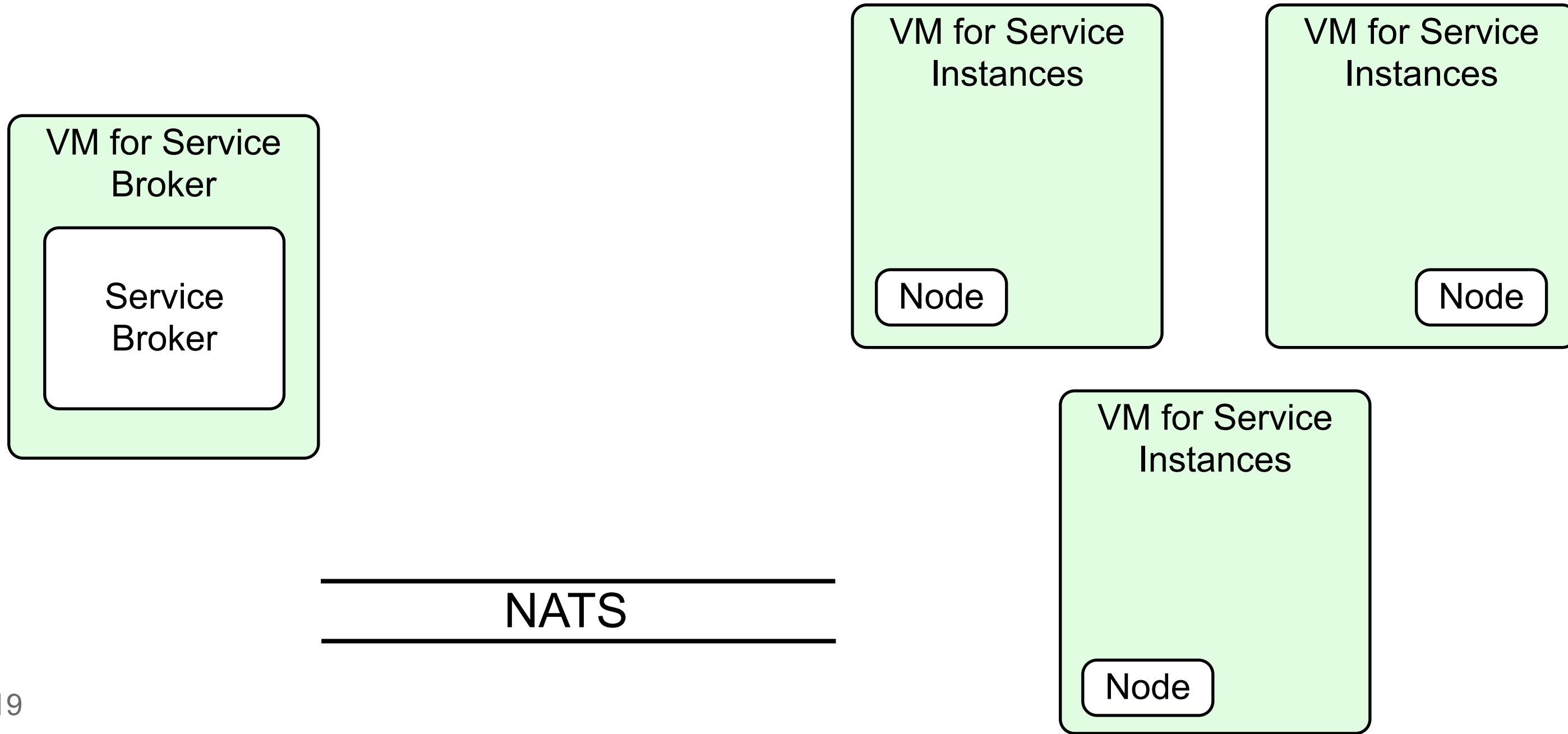
It's Up To YOU Patterns

Another...



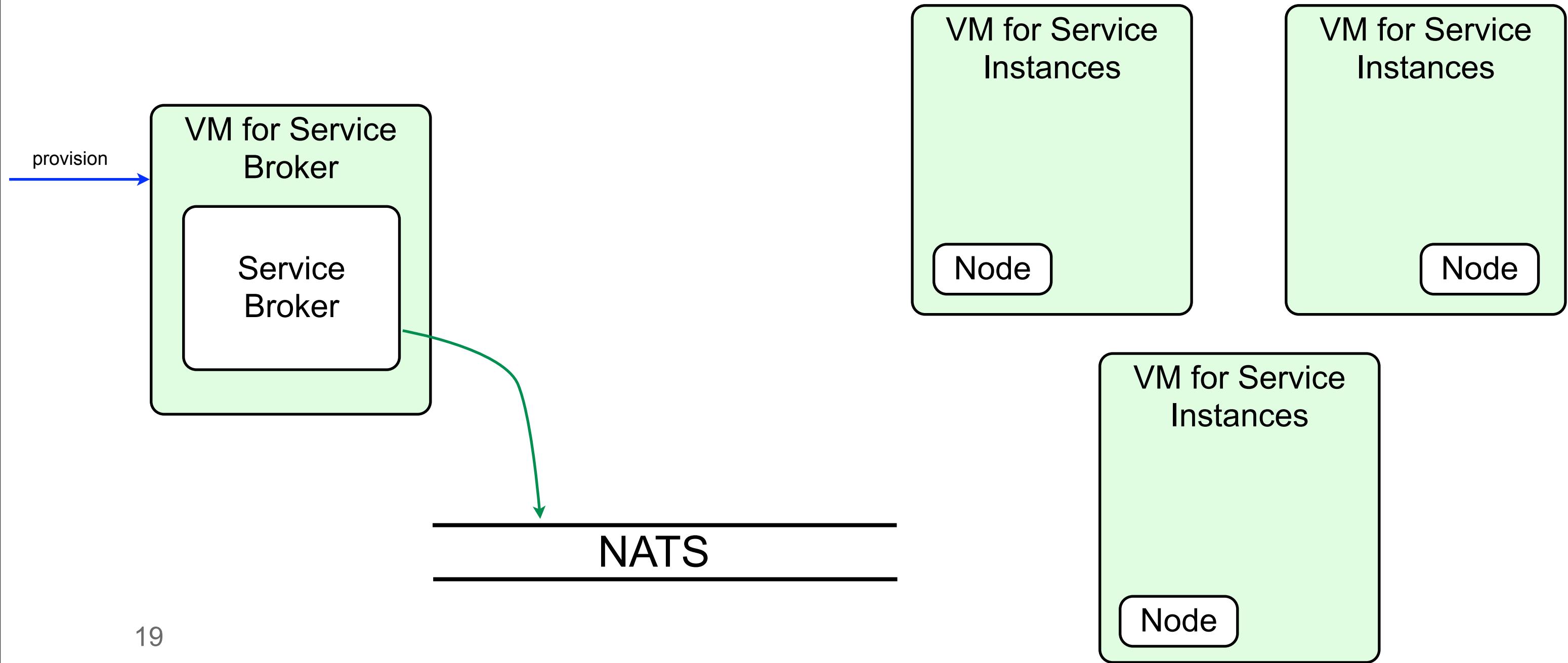
It's Up To YOU Patterns

And another...



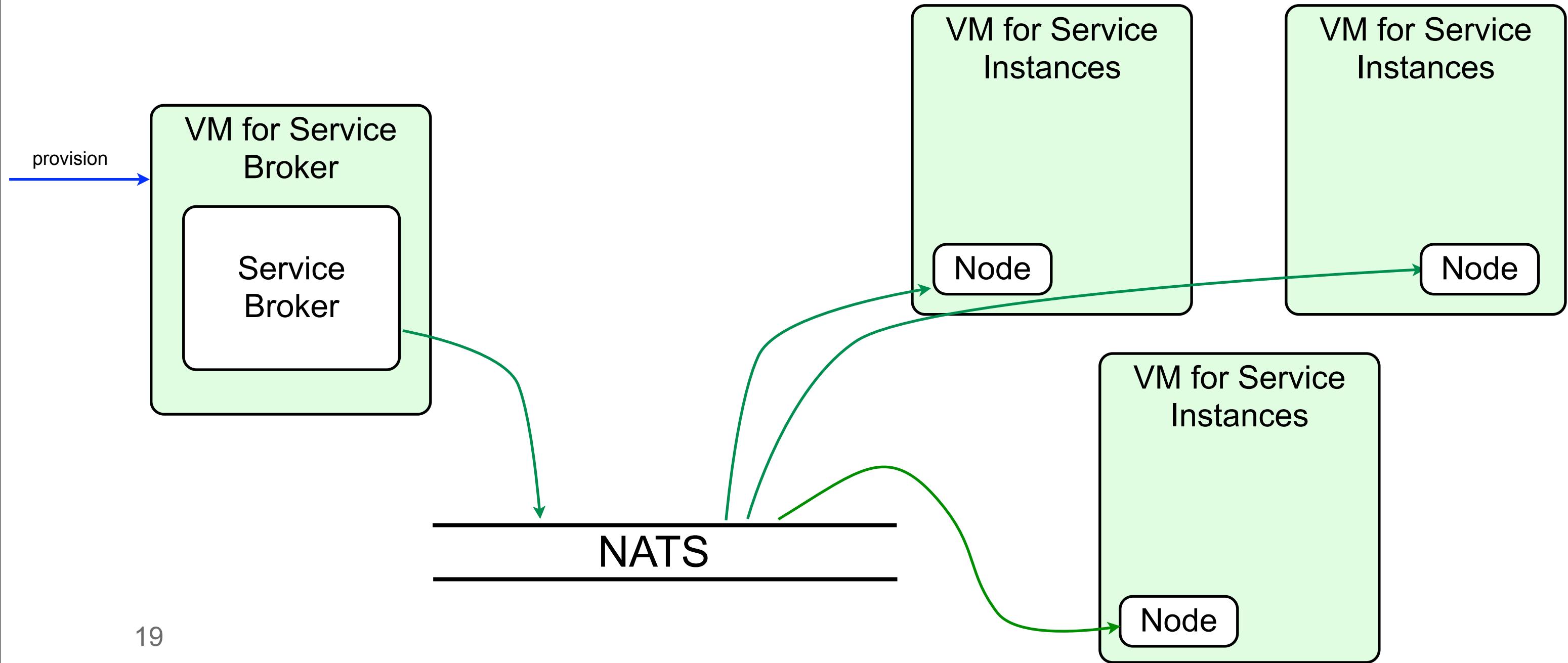
It's Up To YOU Patterns

And another...



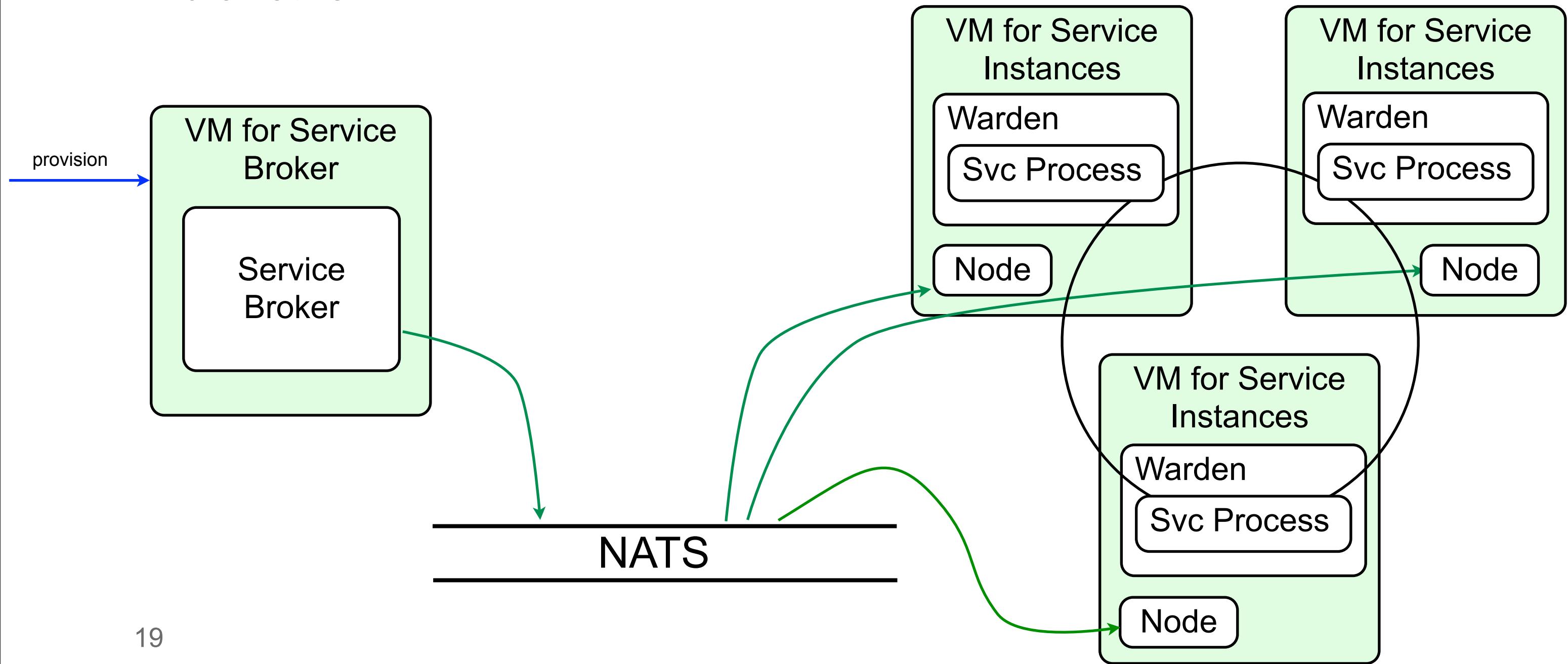
It's Up To YOU Patterns

And another...



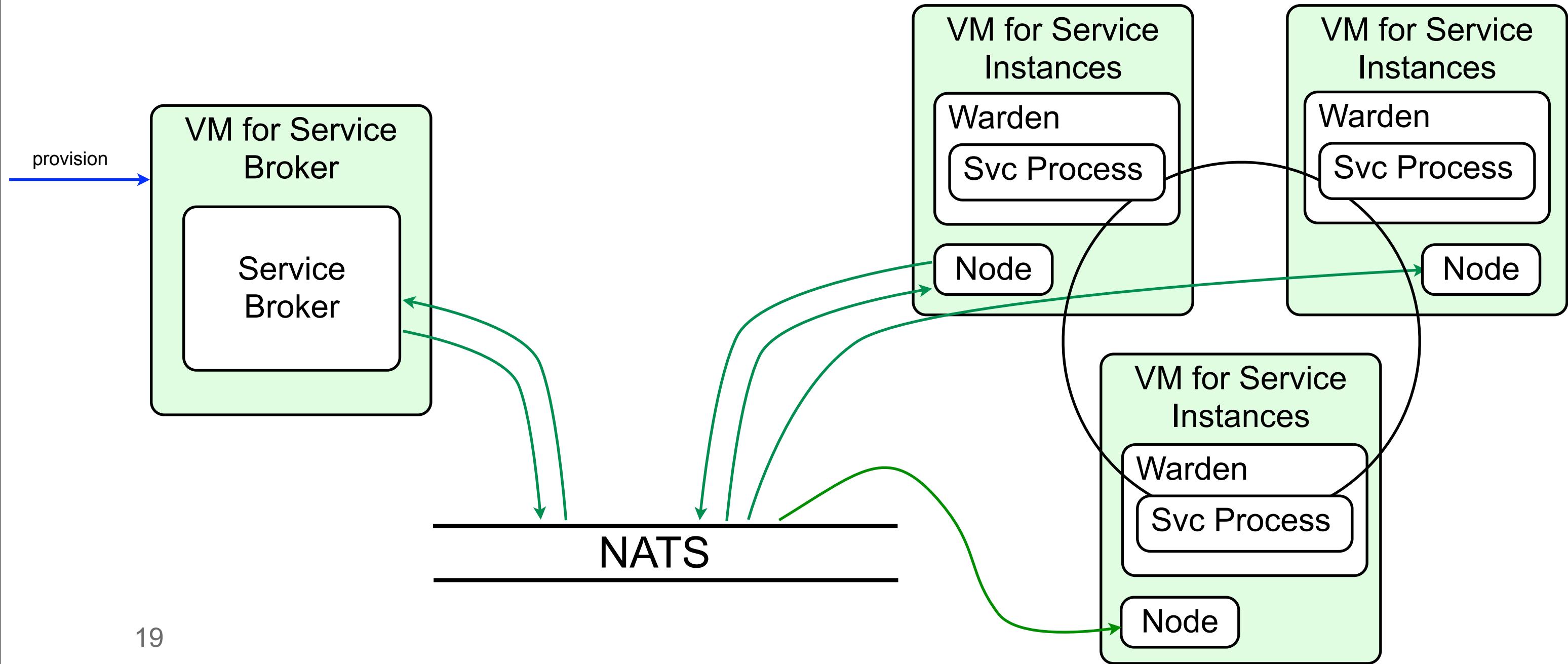
It's Up To YOU Patterns

And another...



It's Up To YOU Patterns

And another...



BOSH deployment

packages

- **service broker**
 - **your impl**
- **service node**
 - **services binaries**
 - **services warden**
 - **your node impl**

jobs

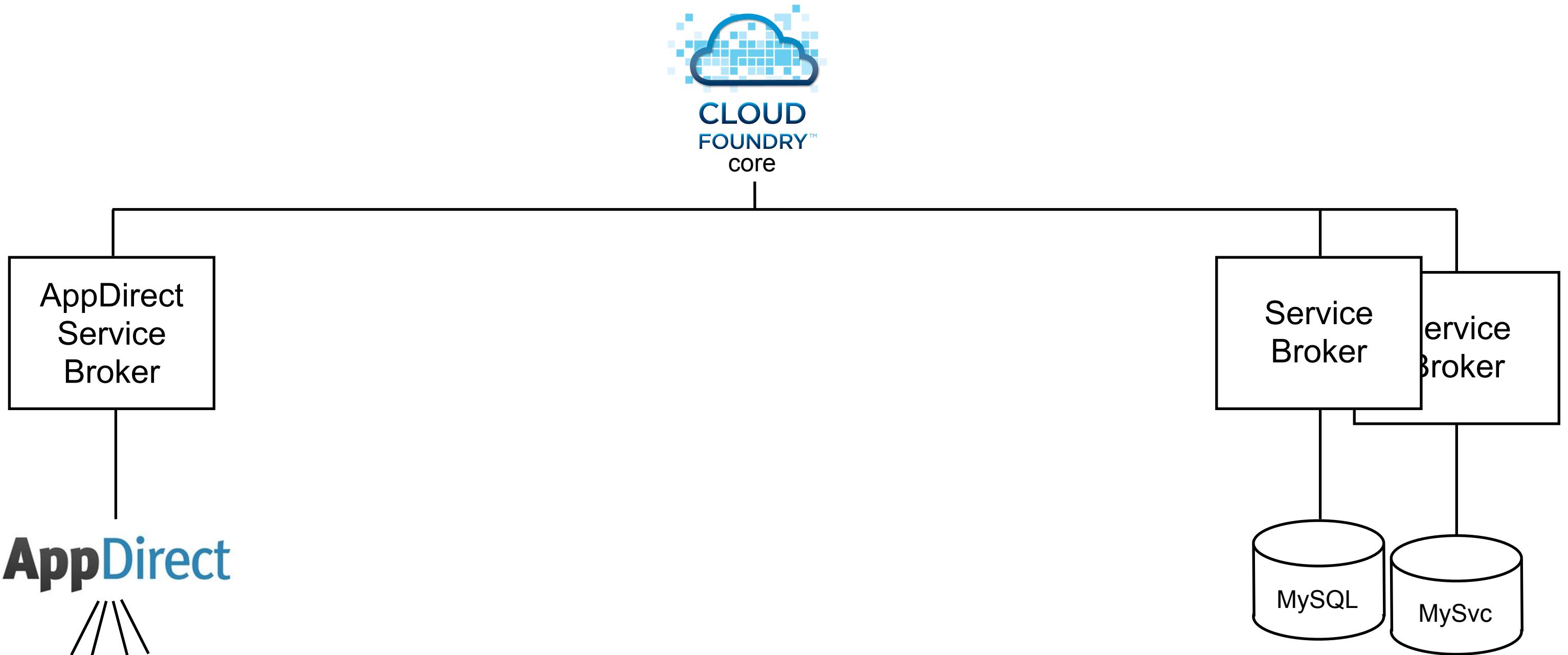
- **service broker**
 - **ref service broker pkg**
 - **start web service**
- **service node**
 - **ref service node pkg**
 - **start node**

Provisions
broker VMs

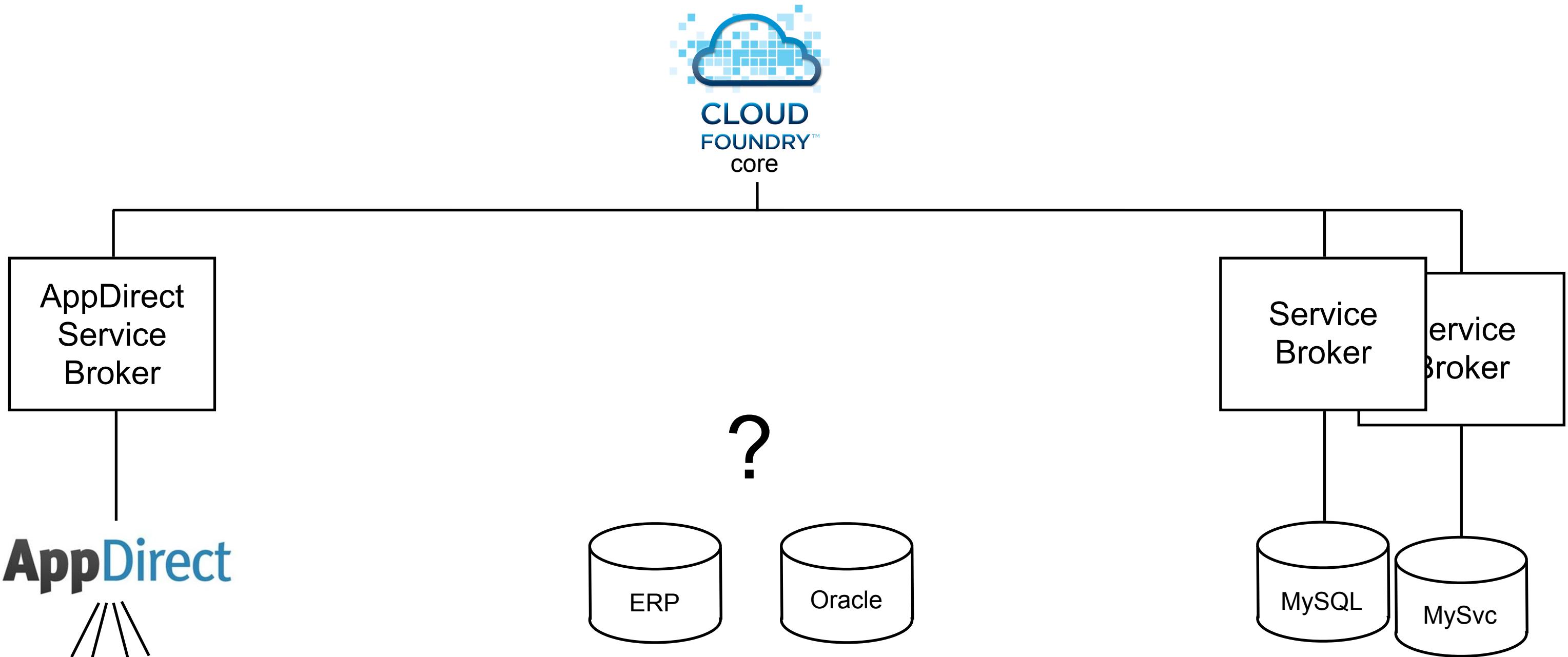
Provisions
service VMs

Configure shared secret (for example)

But what about...

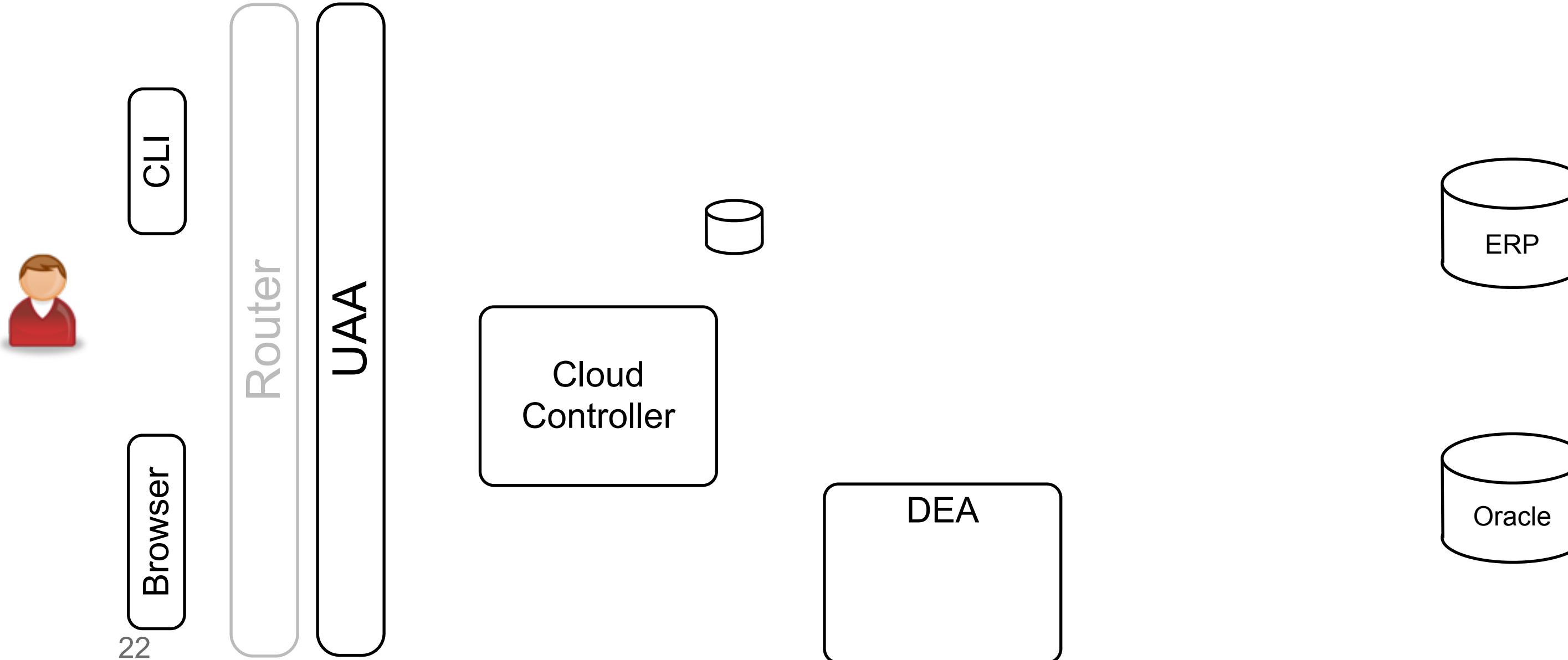


But what about...



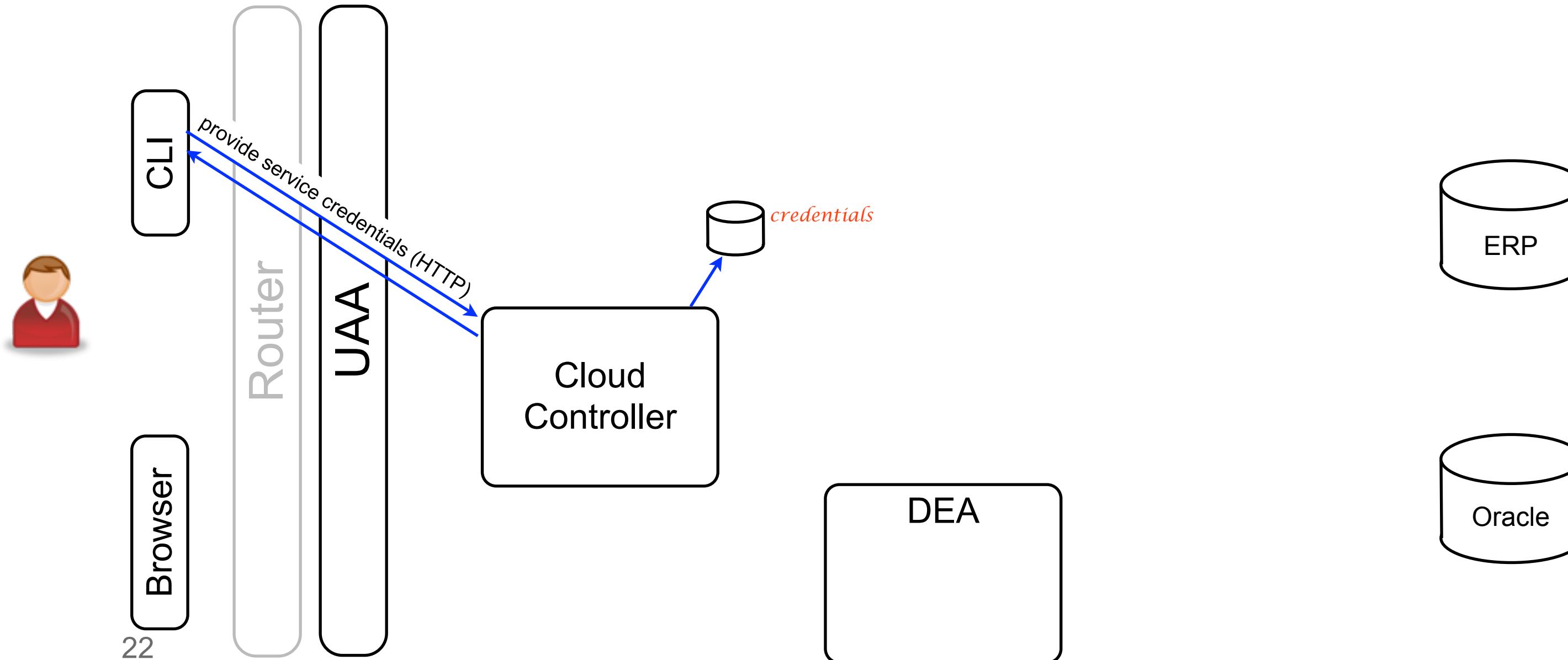
User-provided Service Instances

Prerequisite...



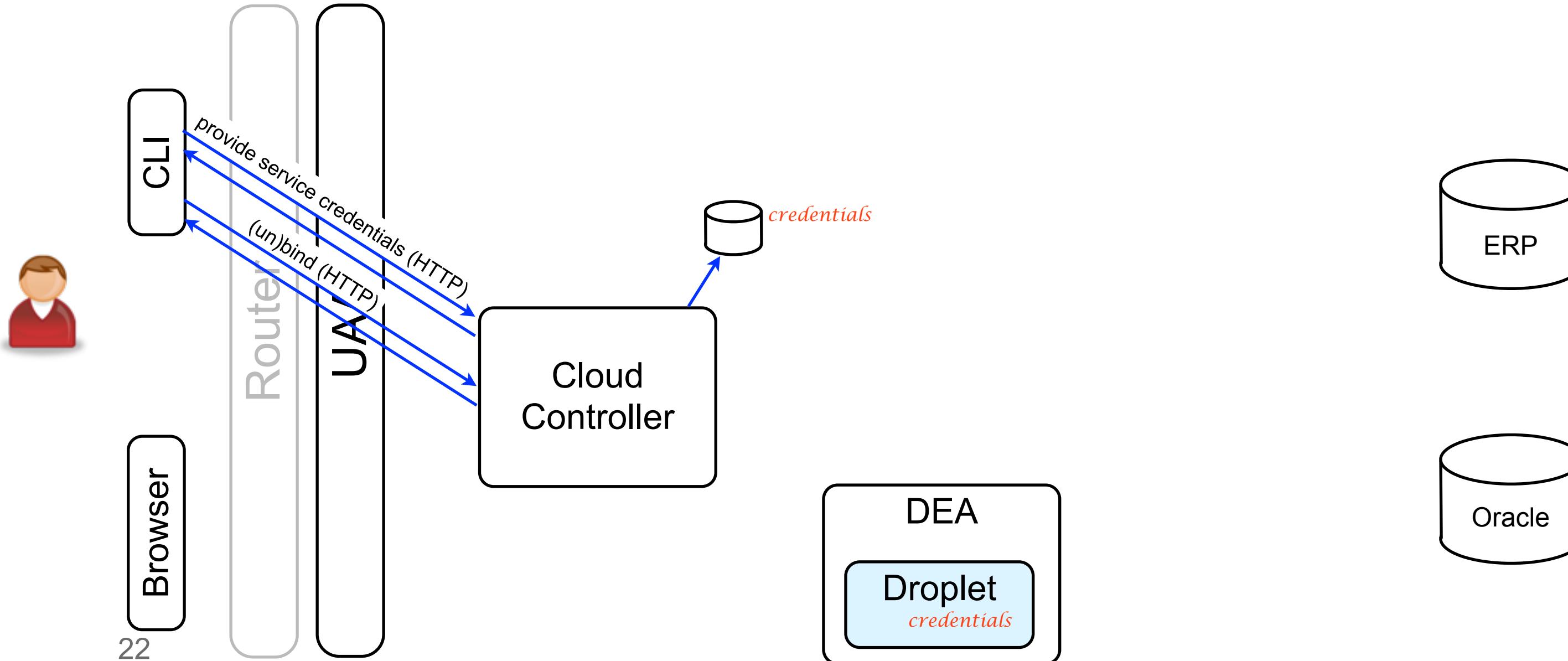
User-provided Service Instances

Prerequisite...



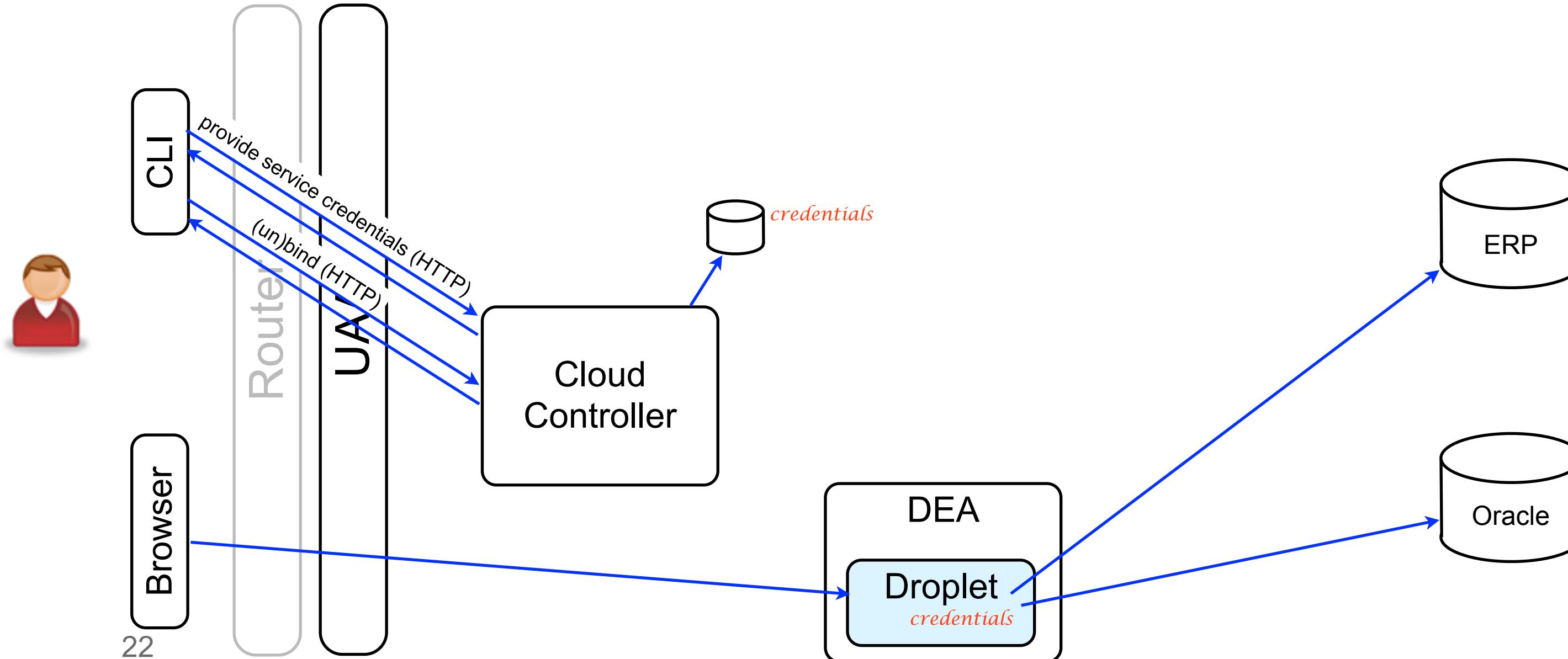
User-provided Service Instances

Prerequisite...



User-provided Service Instances

Prerequisite...



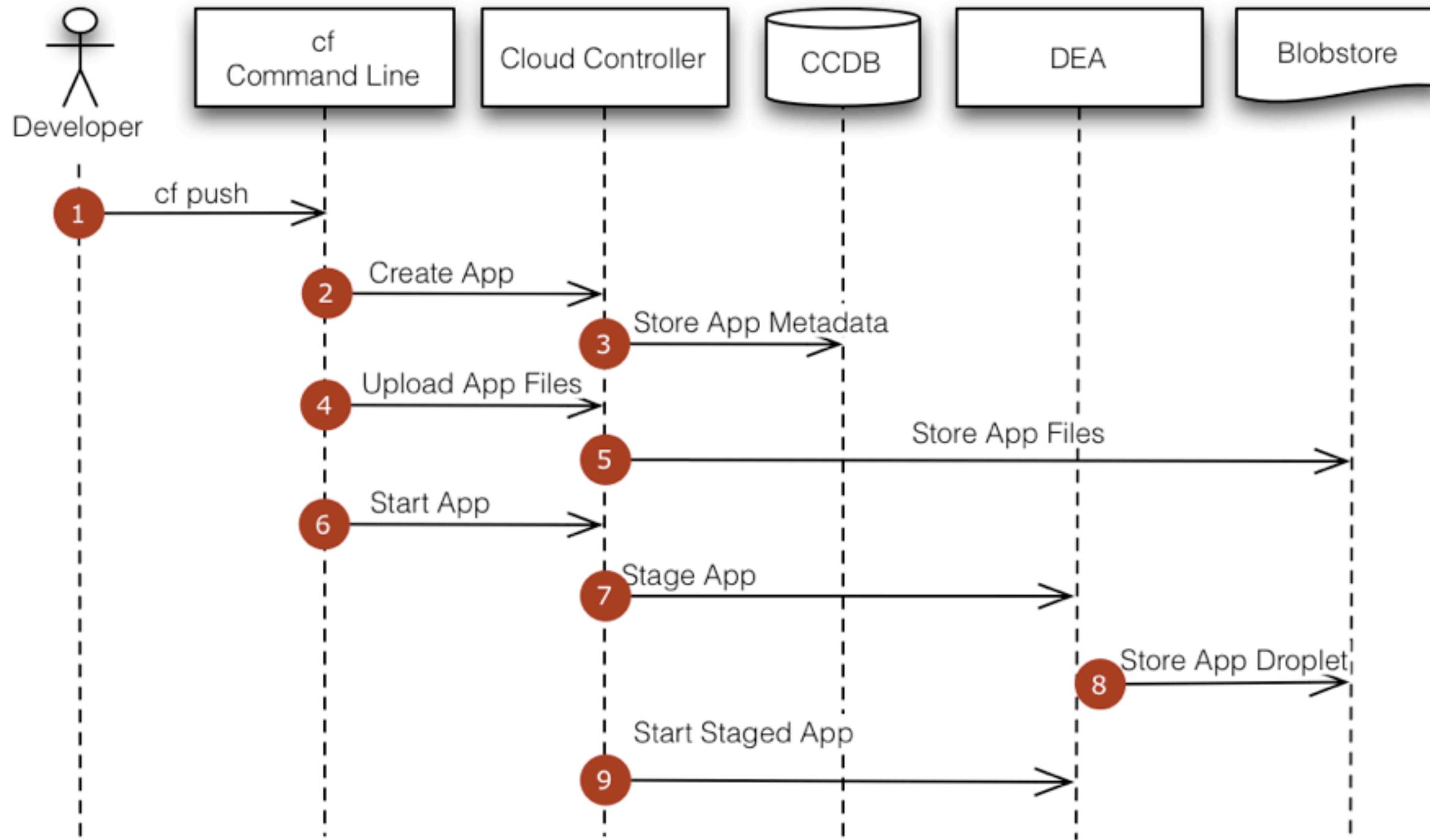
Demo

Tell 'em what you told 'em...

- Extending services offerings within run.pivotal.io
 - Partnering
- Building services
 - Into your own Cloud Foundry
- User-provided service instances
 - For access to pre&externally-provisioned services

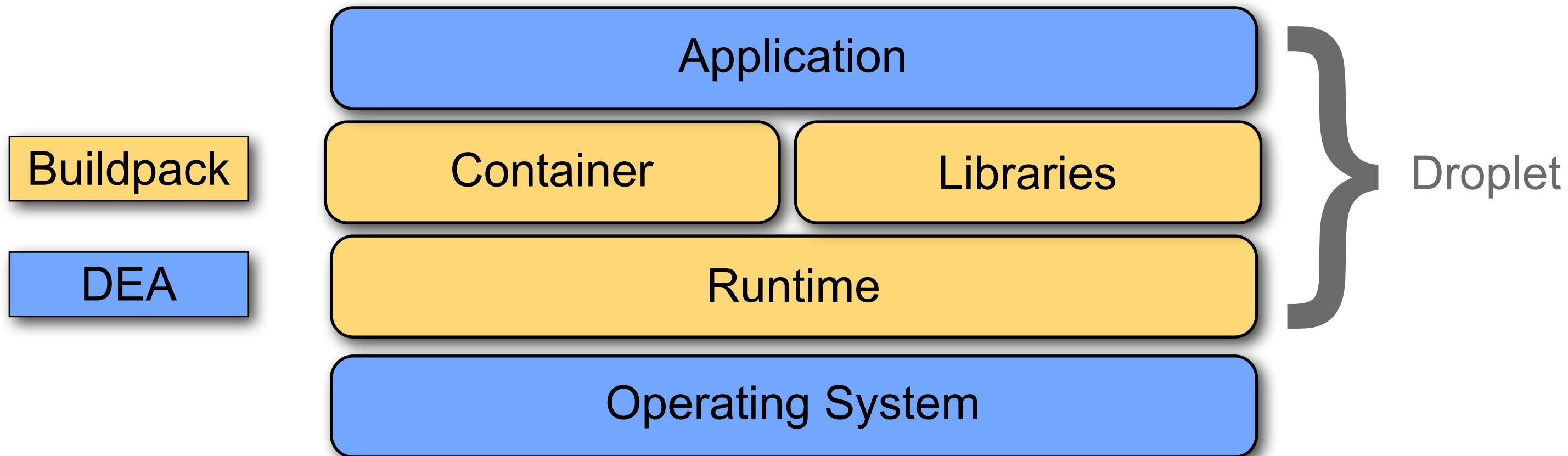
Buildpacks

Deploying to Cloud Foundry



Staging and Buildpacks

Buildpacks are responsible for preparing the machine image for an application

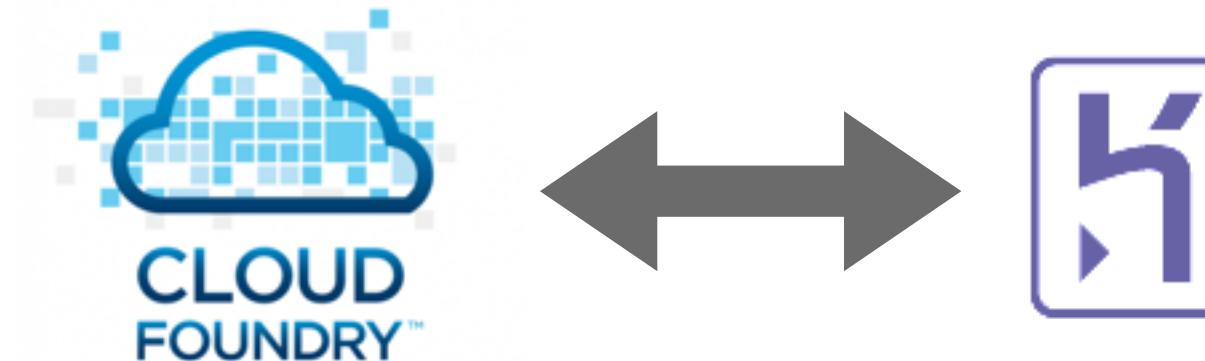


Compatibility

Cloud Foundry buildpacks follow the Heroku buildpack design

Cloud Foundry and Heroku buildpacks are compatible
(if you take care to make them compatible)

Other PaaS providers are adopting the buildpack design



Built-in or BYO

cf push

The application is tested
against a set of curated
buildpacks



cf push --buildpack <url>

The buildpack is
referenced by a Git URL



Built-in Buildpacks

cloudfoundry / **dea_ng**

branch: master ➔ **dea_ng** / buildpacks / **vendor** / [+](#)

Merge pull request #53 from glyn/master ...
mariash authored 9 days ago

..

java @ d1fcf9b Upgrade Java buildpack → cloudfoundry / **java-buildpack**

nodejs @ 6ab2969 Merging local and Heroku Node manifests [finishes #46649889] → cloudfoundry / **heroku-buildpack-nodejs**

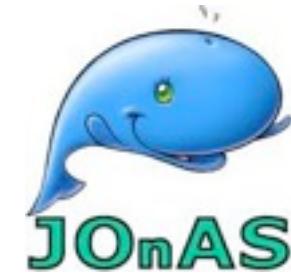
ruby @ 2fc4ad8 bump ruby buildpack to support internetless-mode → cloudfoundry / **heroku-buildpack-ruby**



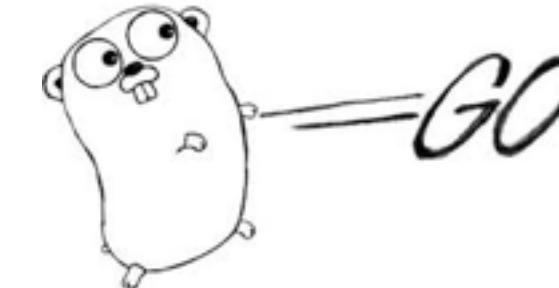
Tested Buildpacks

<https://github.com/cloudfoundry-community/cf-docs-contrib/wiki/Buildpacks>

Containers



Languages



Buildpack API

/bin/detect app_directory

Inspect app bits to determine buildpack applicability

/bin/compile app_directory cache_directory

Download and install runtime, container, packages, libraries;
install app bits as necessary

/bin/release app_directory

Build app start command



/bin/detect

Inspect the app bits to determine if the buildpack knows how to handle the application

 Ruby <small>A Programmer's Best Friend</small>	Gemfile exists
	package.json exists
	setup.py exists

On match, return exit code 0 and write to STDOUT a string identifying the buildpack (often just the name of the language supported)

/bin/detect

```
$ cf push
```

DEA iterates over built-in
buildpacks calling
/bin/detect scripts
until one of them returns
exit code 0

```
$ cf push --buildpack <url>
```

/bin/detect is
not called

/bin/compile

Download and install any necessary
runtime (Java VM, Ruby interpreter, JavaScript interpreter)
container (web server)
support libraries, packages, modules (Ruby gems, NPM packages)

...and then installing the app bits into the runtime or container

/bin/compile Caching

Runtime, container, and support packages are downloaded from sources external to Cloud Foundry

DEA provides a location for storing downloaded artifacts to speed subsequent staging operations

/bin/release

Build a YAML-formatted hash with three possible keys

```
addons: []
config_vars: {}
default_process_types:
web: <start command>
```

On Cloud Foundry, currently only the `web:` value
is used to get the start command for the app

Java Buildpack

Supports a variety of JVM languages, containers, and frameworks with a modular, configurable, and extensible design



Java Buildpack Concepts



Containers

How an application is run

Frameworks

Additional application transformations

JREs Java Runtimes

Java Buildpack Concepts



Containers

Java main()

Tomcat

Groovy

Spring Boot CLI

Play

Frameworks

Spring config

Play config

Play JPA config

New Relic

JREs

OpenJDK



Container Detection Criteria

Java main ()	META-INF/MANIFEST.MF exists with Main-class attribute set
Tomcat	WEB-INF directory exists
Groovy	.groovy file with a main () method, or .groovy file with no classes, or .groovy file with a shebang (# !) declaration
Spring Boot CLI	one or more POGO .groovy files with no main () method, and no WEB-INF directory
Play	start and lib/play.play_*.jar exist

Choose zero or one



Framework Detection Criteria

Spring	spring-core*.jar exists
Play config	Play application detected
Play JPA config	play-java-jpa plugin exists in app
New Relic	New Relic service bound to app

Choose all that apply



/bin/compile Output Example

```
-----> Downloaded app package (18M)
-----> Downloading OpenJDK 1.7.0_21 JRE (17.5s)
      Expanding JRE to .java (1.4s)
-----> Downloading Auto Reconfiguration 0.7.1 (1.4s)
      Modifying /WEB-INF/web.xml for Auto Reconfig
-----> Downloading Tomcat 7.0.42 (3.5s)
      Expanding Tomcat to .tomcat (0.2s)
      Downloading Buildpack Tomcat Support 1.1.1 (0.0s)
-----> Uploading droplet (55M)
```

[DEA]

[Buildpack]

[DEA]

See What's Going On

```
$ cf files <app-name> app
```

```
.buildpack-diagnostics/  
  .java/          └── [ Buildpack-installed runtime  
  .lib/           └── [ Buildpack-installed support libraries  
  .tomcat/        └── [ Buildpack-installed container  
  META-INF/  
  WEB-INF/        └── [ DEA-downloaded application files  
  assets/
```

See What's Going On

```
$ cf files <app-name> staging_info.yml
```

```
detected_buildpack: openjdk-1.7.0_21, tomcat-7.0.42,  
spring-auto-reconfiguration-0.7.1
```

```
start_command: JAVA_HOME=.java
```

```
JAVA_OPTS="-Dhttp.port=$PORT
```

```
-Djava.io.tmpdir=$TMPDIR -XX:MaxPermSize=52428K
```

```
-XX:OnOutOfMemoryError=./.buildpack-diagnostics/killjava
```

```
-Xmx384M -Xss1M" .tomcat/bin/catalina.sh run
```

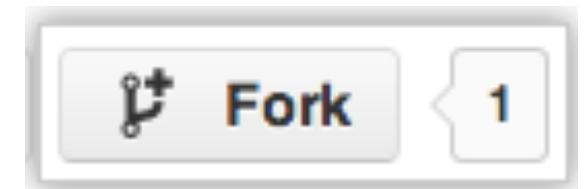


Customization

Two ways to customize the Java buildpack

Configure artifacts used by standard JREs, Containers, and Frameworks

Extend the buildpack with your own JREs, Containers, and Frameworks



Customization is done by forking the buildpack



Customization by Configuration

Configuration files in `java-buildpack/config` determine the behavior of a JRE, Container, or Framework

```
# cloudfoundry/java-buildpack/config/openjdk.yml
---
version: 1.7.0_+
repository_root: "http://download.pivotal.io.s3.amazonaws.com/openjdk/{platform}/{architecture}"
memory_sizes:
memory_heuristics:
  heap: 0.75
  permgen: 0.1
  stack: 0.05
  native: 0.1
```

```
# http://download.pivotal.io.s3.amazonaws.com/openjdk/lucid/x86_64/index.yml
---
1.6.0_27: http://download.pivotal.io.s3.amazonaws.com/openjdk/lucid/x86_64/openjdk-1.6.0_27.tar.gz
1.7.0_21: http://download.pivotal.io.s3.amazonaws.com/openjdk/lucid/x86_64/openjdk-1.7.0_21.tar.gz
1.7.0_25: http://download.pivotal.io.s3.amazonaws.com/openjdk/lucid/x86_64/openjdk-1.7.0_25.tar.gz
1.8.0_M6: http://download.pivotal.io.s3.amazonaws.com/openjdk/lucid/x86_64/openjdk-1.8.0_M6.tar.gz
1.8.0_M7: http://download.pivotal.io.s3.amazonaws.com/openjdk/lucid/x86_64/openjdk-1.8.0_M7.tar.gz
```



Customization by Configuration

Example: customizing the Tomcat artifact for download

```
# cloudfoundry/java-buildpack/config/tomcat.yml
---
version: 7.0.42
repository_root: "http://files.example.com/tomcat-custom"
support:
  version: 1.1.+
  repository_root: "http://files.example.com/tomcat-buildpack-support"
```

```
# http://files.example.com/tomcat-custom/index.yml
---
7.0.40: http://files.example.com/tomcat-custom/tomcat-7.0.40.tar.gz
7.0.41: http://files.example.com/tomcat-custom/tomcat-7.0.41.tar.gz
7.0.42: http://files.example.com/tomcat-custom/tomcat-7.0.42.tar.gz
```



Customization by Configuration

Tomcat container supports simple customization of context.xml and server.xml.

```
cloudfoundry/java-buildpack/resources/tomcat/conf  
└── context.xml  
└── server.xml
```



Customization by Extension

Implement a JRE, Container, or Framework support class as one Ruby file in the appropriate directory

```
cloudfoundry/java-buildpack/lib/java_buildpack
  └── jre
  └── container
  └── framework
```

(with additional support classes as necessary)



Customization by Extension

Support class types have similar interfaces, following the buildpack scripts naming conventions

```
# initialize the support class with platform information provided in context
# context includes app_dir, lib_dir, environment, java_home, java_opts, vcap_application, vcap_services
def initialize(context)

# return a String or an Array<String> that uniquely identifies the container/framework/jre, or nil
def detect

# download and unpack the container/framework/jre, and transform the application as necessary
def compile

# create and return the command to run the application with (containers) or add options to
context[:java_opts] (frameworks)
def release
```



Customization by Extension

Add new support class to config/components.yml

```
---
```

```
containers:
  - "JavaBuildpack::Container::Groovy"
  - "JavaBuildpack::Container::Main"
  - "JavaBuildpack::Container::SpringBootCli"
  - "JavaBuildpack::Container::Tomcat"
  - "JavaBuildpack::Container::Play"
```

```
jres:
  - "JavaBuildpack::Jre::OpenJdk"
```

```
frameworks:
  - "JavaBuildpack::Framework::JavaOpts"
  - "JavaBuildpack::Framework::NewRelic"
  - "JavaBuildpack::Framework::PlayAutoReconfiguration"
  - "JavaBuildpack::Framework::PlayJpaPlugin"
  - "JavaBuildpack::Framework::SpringAutoReconfiguration"
```



Customization

Much more information and documentation included in the GitHub repository

<https://github.com/cloudfoundry/java-buildpack>



Cornelia Davis
cdavis@gopivotal.com
[@cdavisafc](https://twitter.com/cdavisafc)

Scott Frederick
sfrederick@gopivotal.com
[@scotttfred](https://twitter.com/scotttfred)

Community Engineer, Cloud Foundry at Pivotal