



Evolution to CIMI

Charles (Cal) Loomis & Mohammed Airaj

LAL, Univ. Paris-Sud, CNRS/IN2P3

29 August 2013

Priorities for Evolution



Interfaces

- Adopt CIMI as the standard interface to services
- Provide complete browser interface for all services

Simplicity, Scalability, & Robustness

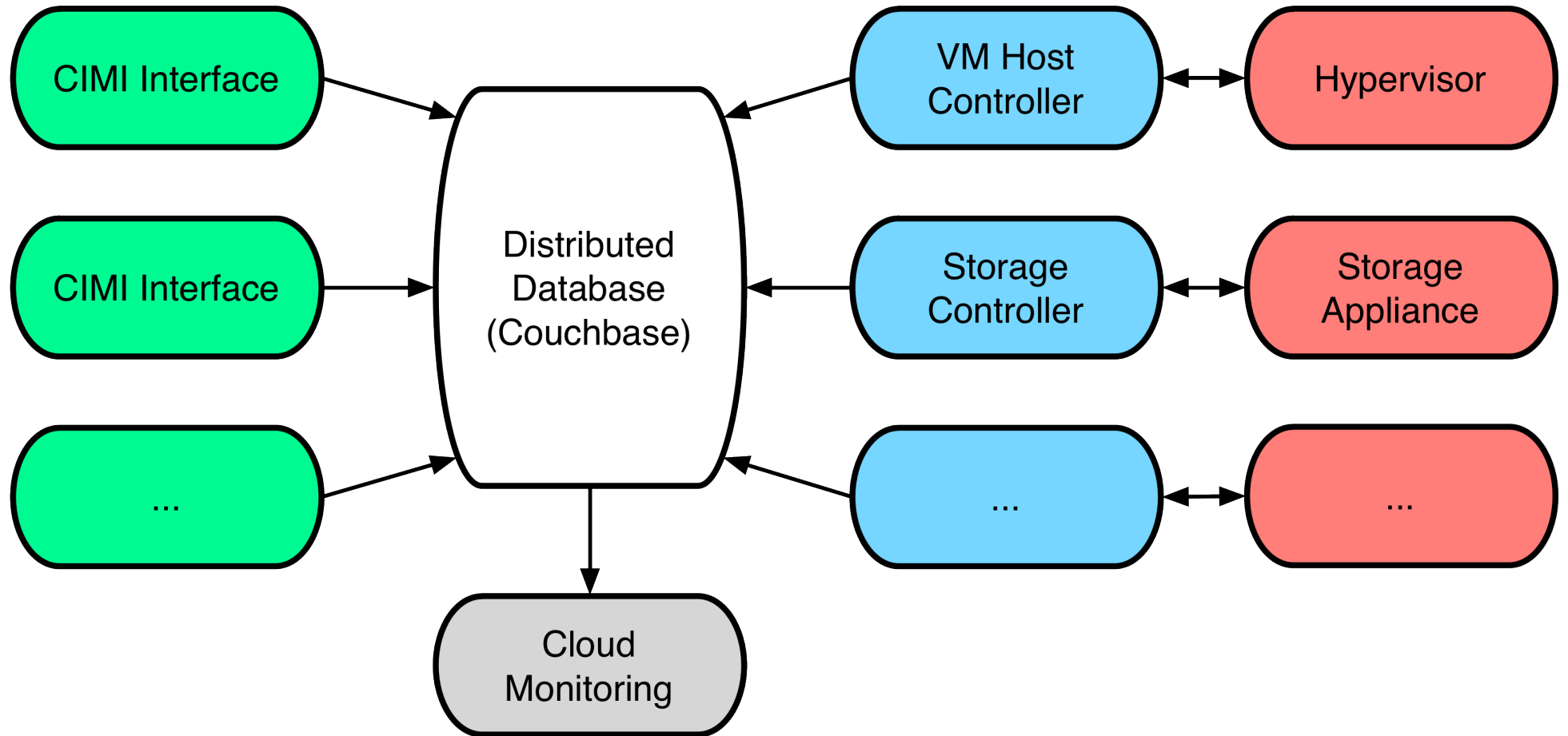
- Direct use of libvirt as VM manager
- Distributed database (Couchbase) as information 'bus'

Better services for system administrators

- Improved overview and monitoring of infrastructure
- Fine-grained accounting for all resources
- Migration control



New Architecture



Configuration Changes



Couchbase Configuration File

- Contact parameters for Couchbase database
- Defines document IDs for service configurations
- INI format with sections for individual services:
 - `/etc/stratuslab/couchbase.cfg`

All Service Configurations in Couchbase

- Documents with identifiers: `ServiceCfg/name-of-service`
- JSON format for all service configurations

Benefits



Scalability

- Couchbase will be more scalable than current centralized model
- Services are stateless and can be replicated for high loads

Simplicity

- Clean workflows: all interactions happen through “jobs” in CB
- Single user-facing service (CIMI)

Flexibility

- Support for multiple and varied backend services
 - Different types of storage backends (backed up, fast, etc.)
 - Virtual machines as well as linux containers
 - Dynamic network configurations



Service

- Written in Clojure (lisp on JVM)
- Uses “ring” web application framework
- Uses “friend” framework for authentication
- As for other services it runs within a Jetty container

External Dependencies

- All state is stored in Couchbase
- Allows HA deployments of both CIMI server and controllers

CIMI



CIMI	
daemon	cimi
purpose	CIMI interface to cloud services
ports	443 (redirect from port 80)
language	Clojure (depl. in Jetty container)
external requirements	Couchbase
config. files	/etc/stratuslab/couchbase.cfg others in Couchbase database
logs	/opt/stratuslab/cimi/logs/*

Couchbase on Frontend



Install

```
$ stratus-install --couchbase
```

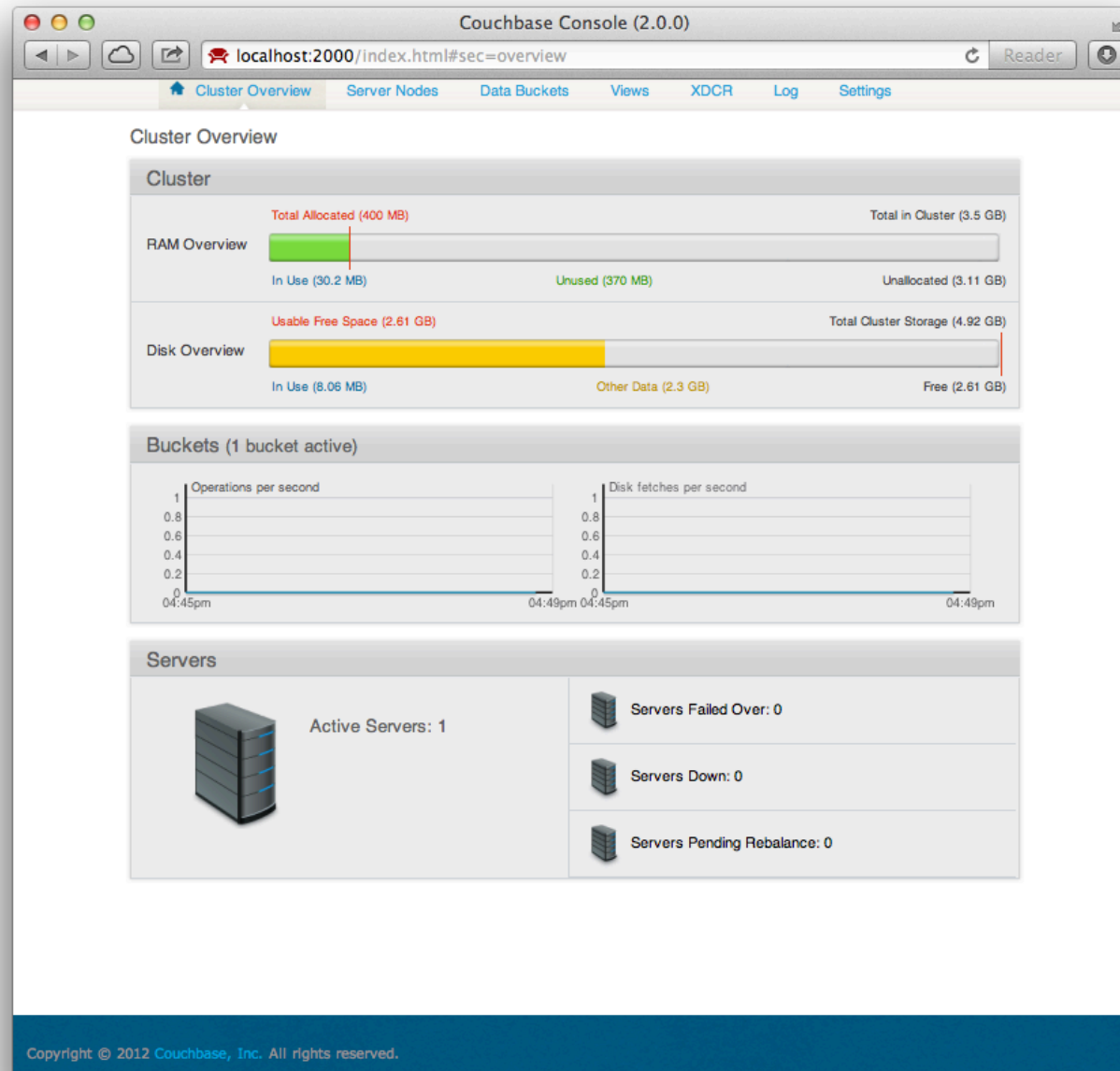
Access Couchbase console

- Administrator username: “admin”
- Generated password: /opt/couchbase/cluster-password.txt
- From separate window tunnel to the machine:

```
$ ssh -L2000:your-machine:8091 root@your-machine -N
```

- To see database and contents view <http://localhost:2000/>

Couchbase console



Configure Authentication



Add JSON document to database

- Create document named “ServiceCfg/authn/basic”
- This is the equivalent of the `login-pswd.properties` file

```
{
  "root": {
    "username": "root",
    "password": "bcrypted value",
    "roles": [
      "::ADMIN"
    ]
  }
}
```

Crypting Password



Add JSON document to database

- Add package “py-bcrypt”
- Replace ‘hello’ with your password

```
python -c "  
> import bcrypt  
> h=bcrypt.hashpw('hello', bcrypt.gensalt())  
> print h  
> "  
$2a$12$zvS7axGrws6/YH2AuIyXpufc174KV5bjBTp.vo400sGZsehP7CpFS
```

- Put return value ‘\$2a\$12...’ in your Couchbase file.

CIMI on Frontend



Install

```
$ stratus-install --cimi
```

Access service with browser or 'curl'

- Browser: <https://your-machine/>
- Curl:

```
$ curl -s --insecure https://your-machine | \
  python -mjson.tool

{
  "acl": {
    "owner": {
      "principal": "::ADMIN",
      "type": "ROLE"
    }
  }
  ...
}
```

Service Messages



Service Messages → user announcements

List existing messages (should be none):

```
$ curl -s --insecure \  
  https://onevm-73.lal.in2p3.fr/ServiceMessage | \  
  python -mjson.tool  
  
{  
  "count": 0,  
  "id": "/ServiceMessage",  
  "resourceURI": "http://stratuslab.eu/cimi/1/  
ServiceMessageCollection",  
  "serviceMessages": []  
}
```

Service Messages



Add new message:

```
$ curl -s --insecure \  
  -H "Content-Type: application/json" \  
  --user root:hello \  
  https://onevm-73.lal.in2p3.fr/ServiceMessage -d '  
{ "name": "message title",  
  "description": "some longer text message" }'
```

Check that it shows up in the listing.

Can also get individual message by visiting direct URL.

Do you see differences if you are anonymous or root?

Service Messages



Add new message:

```
$ curl -s --insecure \  
  -H "Content-Type: application/json" \  
  --user root:hello \  
  https://onevm-73.lal.in2p3.fr/ServiceMessage -d '  
{ "name": "message title",  
  "description": "some longer text message" }'
```

Check that it shows up in the listing.

Can also get individual message by visiting direct URL.

Do you see differences if you are anonymous or root?



Core features of service available:

- Service configuration and authentication
- Access control via simple ACLs
- Standard workflows for CIMI resources
- Ability to bridge CIMI and persistent disk service

Near future

- Integration with standard StratusLab CLI
- Replacement of service interfaces with CIMI

Expect version with CIMI interface before end of 2013.

Exercises



- 1. Install Couchbase and interact with it via console**
- 2. Configure authentication and install CIMI**
- 3. Verify CIMI works by generating and listing service messages**

Questions and Discussion

website <http://stratuslab.eu>

twitter [@StratusLab](https://twitter.com/StratusLab)

support support@stratuslab.eu

StratusLab source <http://github.com/StratusLab>

SlipStream source <http://github.com/slipstream>



<http://stratuslab.eu/>

Copyright © 2013, Members of the StratusLab collaboration.

This work is licensed under the Creative Commons Attribution 3.0 Unported License (<http://creativecommons.org/licenses/by/3.0/>).

