

Spring XD

Management and Monitoring

Management and Monitoring

- Admin Console
- Monitoring
 - Streams
 - Jobs

Admin Console

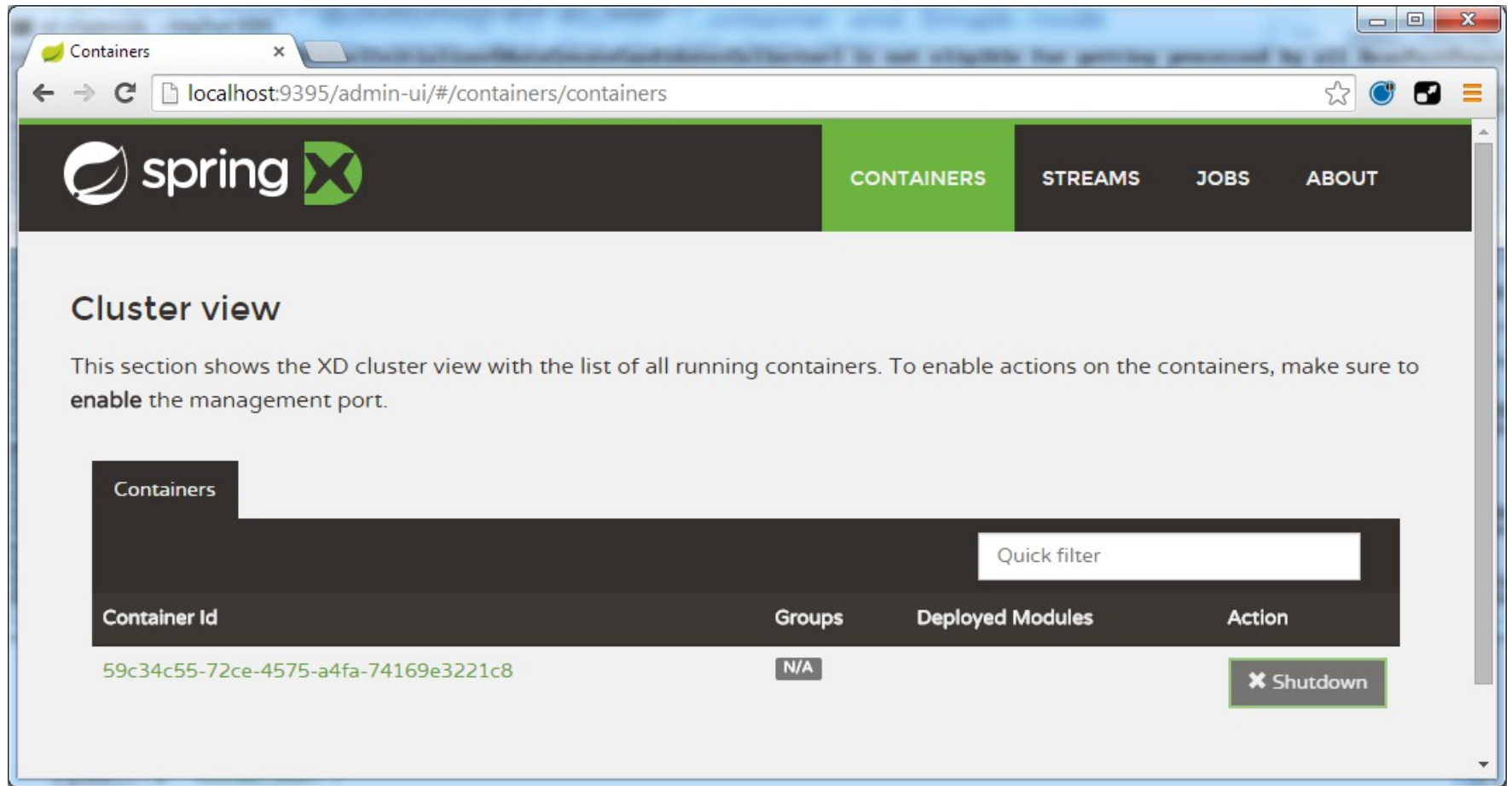
- Override default port 9393 by:
 - Changing servers.yml file:

```
...  
# Port that admin-ui is listening on  
server:  
  port: 9395  
...
```

- Or specifying `--httpPort` option on command line

```
$ xd-singlenode --httpPort 9395  
$
```

Admin Console



The screenshot shows a web browser window with the URL `localhost:9395/admin-ui/#/containers/containers`. The page features a dark header with the Spring XD logo and navigation links for CONTAINERS, STREAMS, JOBS, and ABOUT. The main content area is titled "Cluster view" and includes a descriptive paragraph. Below this is a "Containers" section with a table of running containers. The table has columns for Container Id, Groups, Deployed Modules, and Action. A single container is listed with the ID `59c34c55-72ce-4575-a4fa-74169e3221c8`, no groups, and a "Shutdown" button.

Containers

Quick filter

Container Id	Groups	Deployed Modules	Action
59c34c55-72ce-4575-a4fa-74169e3221c8	N/A		✕ Shutdown

Spring Boot HTTP Endpoints

- Spring Boot has built-in HTTP JSON interface for monitoring application health
- Includes the following endpoints: `autoconfig`, `beans`, `configprops`, `dump`, `env`, `health`, `info`, `metrics`, `mappings`, `shutdown`, **and** `trace`
- Endpoints are accessible through management URI
- Some endpoints are secured
- URI format is:

```
http://<hostname>:<mgmt-port>/management/<endpoint>
```

Spring Boot HTTP Endpoints

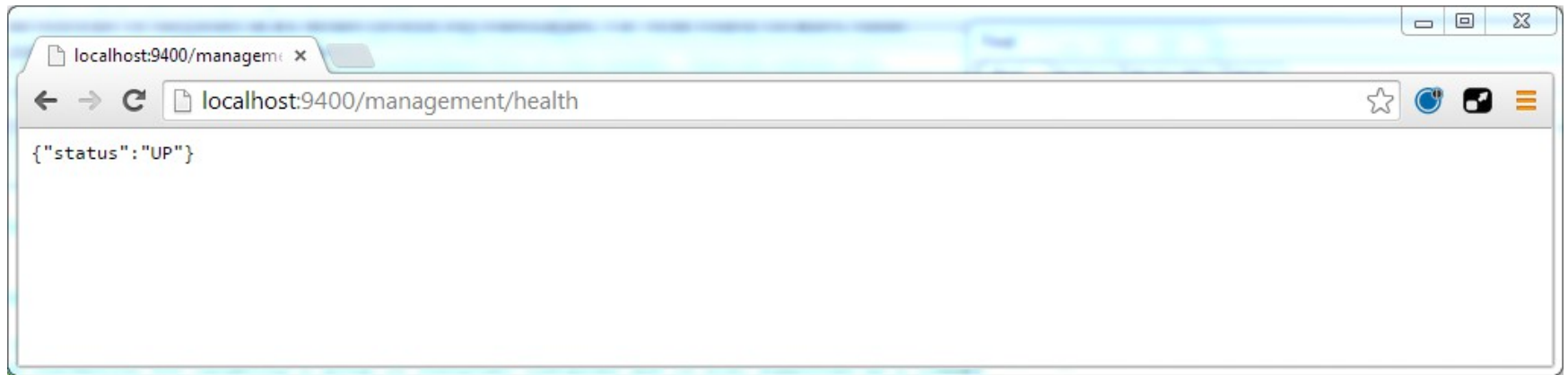
- Override default port 9393 by:
 - Changing servers.yml file:

```
...  
# Port management server is listening on  
management:  
  port: 9400  
...
```

- Or specifying `--mgmtPort` option on command line

```
$ xd-singlenode --mgmtPort 9400  
$
```

Spring Boot Management HTTP Endpoints



- Set `management.port=-1` to disable Spring Boot endpoints
- Set `XD_JMX_ENABLED=false` to disable JMX management

Enabling Container Shutdown from UI

- To allow containers to be shutdown from the Admin UI, add following content to config/servers.yml:

```
---  
# Enable container shutdown in Admin UI  
spring:  
  profiles: container  
management:  
  port: 0
```


Monitoring Deployed Streams

- Stream components in modules are exposed for JMX monitoring
- Based on Spring Integration
- Object naming strategy:
 - Domain name format is `xd.<stream name>`
 - Object name is `<module name>.<module index>`

Monitoring Deployed Streams

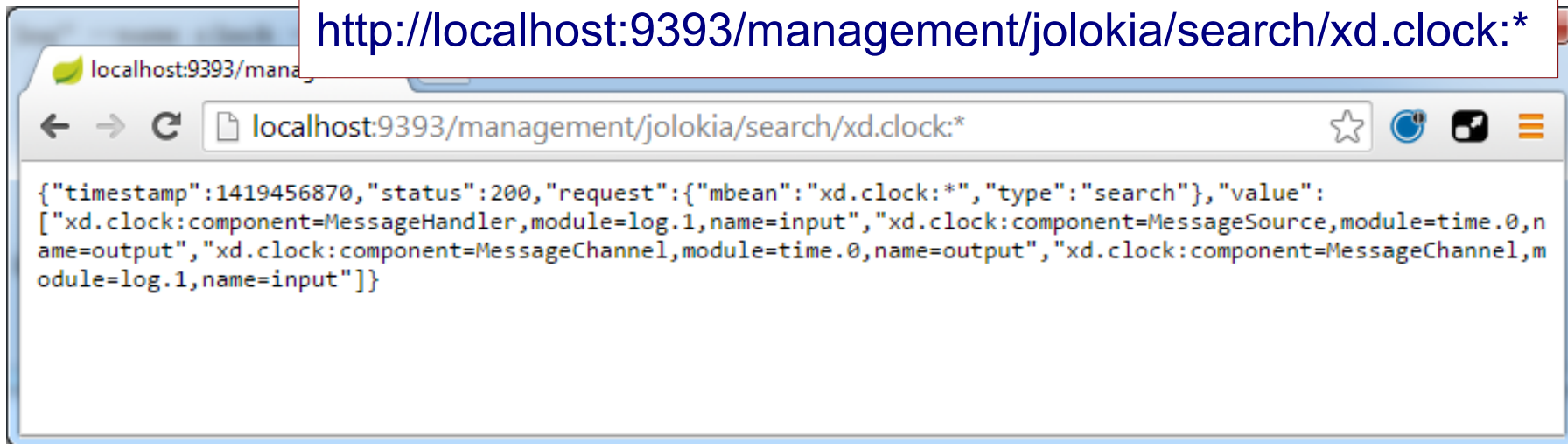
- Example: Fetch all modules for deployed stream “clock”

```
xd:>stream create --definition "time | log" --name clock  
Created new stream 'clock'  
xd:>stream deploy --name clock  
Deployed stream 'clock'
```

- Creates Mbeans with domain name xd.clock
- Creates objects time.0 and log.1

Monitoring Deployed Streams

http://localhost:9393/management/jolokia/search/xd.clock:*



```
{ "timestamp": 1419456870, "status": 200, "request": {
  "mbean": "xd.clock:*", "type": "search", "value":
  [ "xd.clock:component=MessageHandler,module=log.1,name=input",
    "xd.clock:component=MessageSource,module=time.0,name=output",
    "xd.clock:component=MessageChannel,module=time.0,name=output",
    "xd.clock:component=MessageChannel,module=log.1,name=input" ]
}
```

Stream Attributes and Operations

- Source modules have `MessageSourceMetrics` attributes and operations available
- Processor, sink modules have `MessageHandlerMetrics` attributes and operations available
- Channels have `MessageChannelMetrics` attributes and operations available

Stream Attributes and Operations

- Source modules have `MessageSourceMetrics` attributes and operations available
- Processor, sink modules have `MessageHandlerMetrics` attributes and operations available
- Channels have `MessageChannelMetrics` attributes and operations available

Monitoring Jobs

- Similar naming convention as stream Mbeans
 - Domain name format is `xd.<job name>`
 - Object name is `<module name>.<module index>`
- REST API available

Monitoring Jobs

- Similar naming convention as stream Mbeans
 - Domain name format is `xd.<job name>`
 - Object name is `<module name>.<module index>`
- REST API available to:
 - Return job names (`jobName`)
 - Return configurations (`job/configuration`)
 - Return execution status (`jobs/execution`)
 - Stop or restart jobs
 - Return job steps (`/jobs/executions/{jobExecutionId}/steps`)
 - Return job step progress (`/stepExecutionId/progress`)

Overview

- Spring XD leverages:
 - Spring Boot's HTTP and JMX support
 - Spring Integration's Mbean Exporters
- By default:
 - JMX is enabled on port
 - Spring Boot management endpoints exposed using JMX over HTTP on port 9393
 - Spring Integration components are exposed over JMX
 - Mbeans are accessed over HTTP using Jolokia
 - Jolokia is “remote JMS with JSON over HTTP” - www.jolokia.org