Web Management

Memory Use

- RabbitMQ can report on its memory use, to know where the system is using memory.
- obtain the memory use report by invoking

```
# rabbitmqctl status.
```

Mnesia

- store information about queues, exchanges, bindings, and so on
- Mnesia keeps an in-memory copy of all its data (even on disc nodes).
- Typically this will only be large when there are a large number of queues, exchanges, bindings, users or virtual hosts.

Mnesia

- launch the Mnesia database at start up of RMQ
- if Mnesia fails to start, then RabbitMQ will fail too.
- MNESIA_BASE directory

Plugins

- extend the behavior of the server
- written in Erlang
- run together with the server in the same Erlang VM.

HP

Web management console

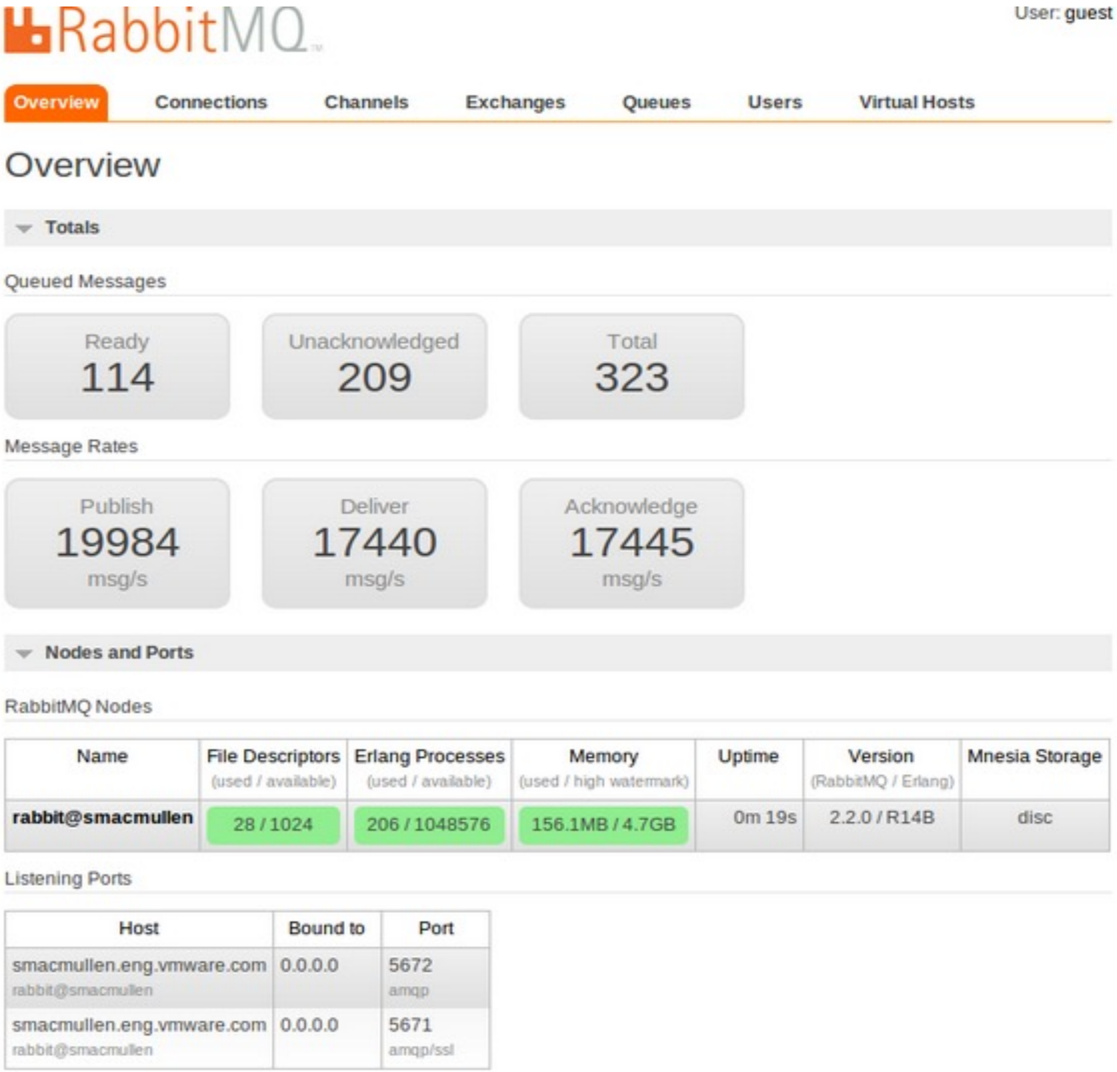
- The rabbitmq-management plugin :
  - provides an HTTP-based API for management and monitoring of RabbitMQ server
  - browser-based UI
  - a command line tool, **rabbitmqadmin**.

Web management console

- Features include:
  - Declare, list and delete exchanges, queues, bindings, users, virtual hosts and permissions.
  - Monitor queue length, message rates globally and per channel, data rates per connection, etc.
  - Send and receive messages.
  - Monitor Erlang processes, file descriptors, memory use.
  - Export / import object definitions to JSON.
  - Force close connections, purge queues.

HP

rabbitmq-management



RabbitMQ

User: guest

Overview

Totals

Queued Messages

Ready	Unacknowledged	Total
114	209	323

Message Rates

Publish	Deliver	Acknowledge
19984 msg/s	17440 msg/s	17445 msg/s

Nodes and Ports

RabbitMQ Nodes

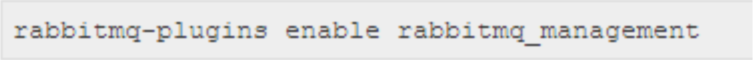
Name	File Descriptors (used / available)	Erlang Processes (used / available)	Memory (used / high watermark)	Uptime	Version (RabbitMQ / Erlang)	Mnesia Storage
rabbit@smacmullen	28 / 1024	206 / 1048576	156.1MB / 4.7GB	0m 19s	2.2.0 / R14B	disc

Listening Ports

Host	Bound to	Port
smacmullen.eng.vmware.com rabbit@smacmullen	0.0.0.0	5672 amqp
smacmullen.eng.vmware.com rabbit@smacmullen	0.0.0.0	5671 amqp/ssl

rabbitmq-plugins

- To enable it :



```
rabbitmq-plugins enable rabbitmq_management
```

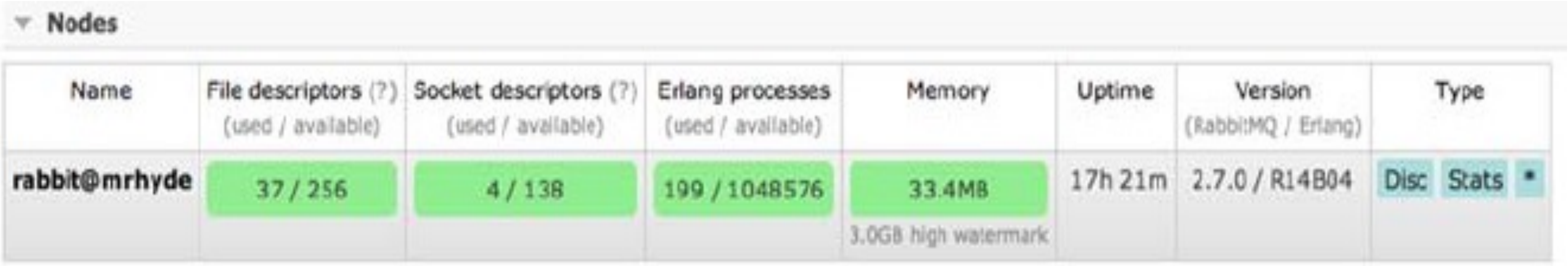
- The web UI :
  - http://*server-name*:15672/
- **rabbitmqadmin** :
  - http://*server-name*:15672/cli/

fresh installation the user "guest" is created with password "guest"

- May need to restart the broker for the changes to take effect

Managing RabbitMQ

- ***Monitoring the Erlang VM***
  - the number of Erlang processes
  - installed versions of RabbitMQ and Erlang.
  - port and host RabbitMQ is listening on

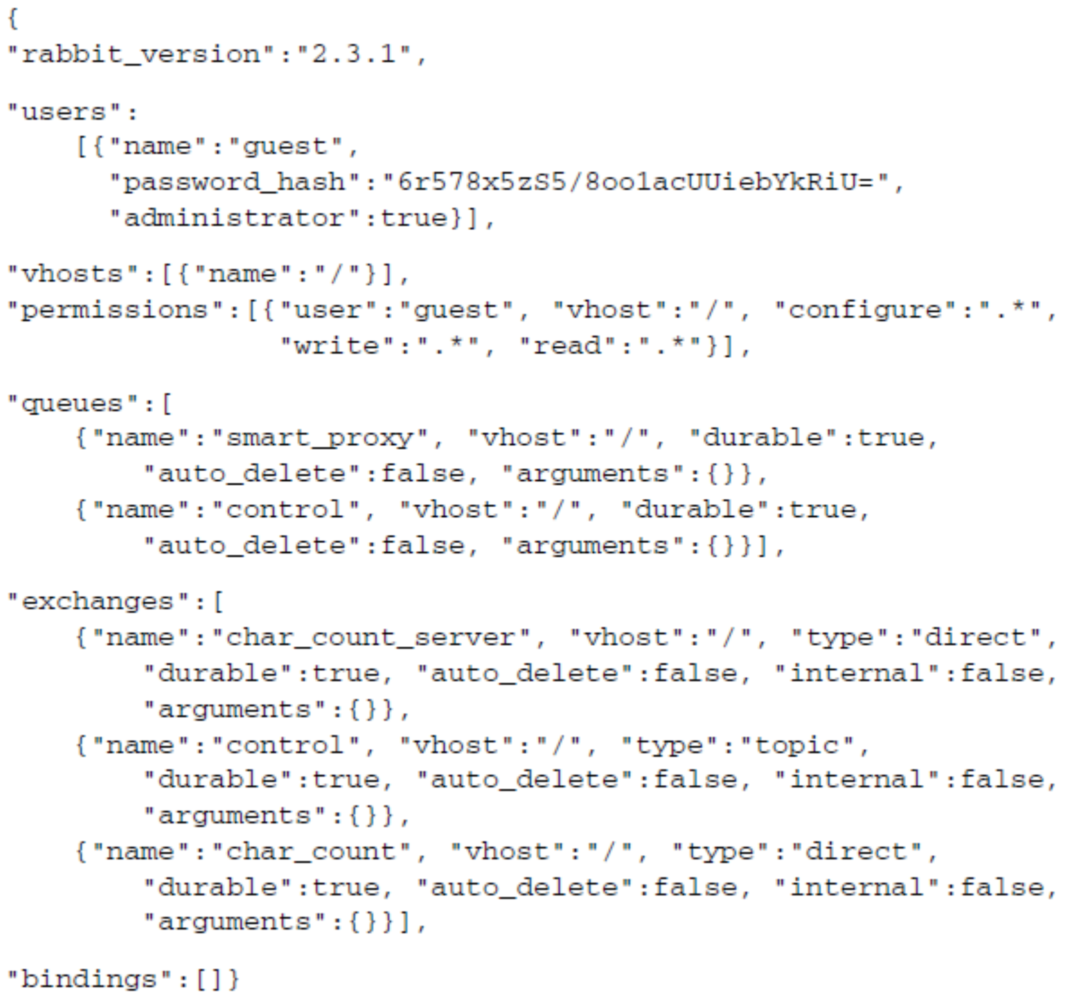


Nodes

Name	File descriptors (?) (used / available)	Socket descriptors (?) (used / available)	Erlang processes (used / available)	Memory	Uptime	Version (RabbitMQ / Erlang)	Type
rabbit@mrhyde	37 / 256	4 / 138	199 / 1048576	33.4MB 3.0GB high watermark	17h 21m	2.7.0 / R14B04	Disc Stats *

Importing configuration - JSON

- Download Broker Configuration → Save on disk

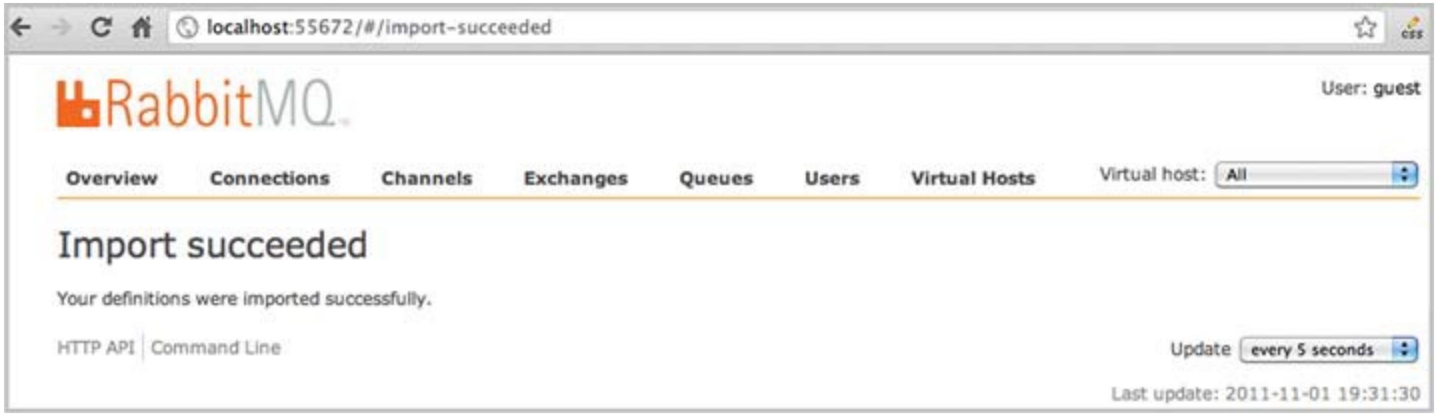


```
{
"rabbit_version":"2.3.1",

"users":
    [{"name":"guest",
      "password_hash":"6r578x5zS5/8oo1acUUiebYkRiU=",
      "administrator":true}],

"vhosts":[{"name":"/"}],
"permissions":[{"user":"guest", "vhost":"/", "configure":".*",
                "write":".*", "read":".*"}],

"queues":[
    {"name":"smart_proxy", "vhost":"/", "durable":true,
        "auto_delete":false, "arguments":{}},
    {"name":"control", "vhost":"/", "durable":true,
        "auto_delete":false, "arguments":{}}],

"exchanges":[
    {"name":"char_count_server", "vhost":"/", "type":"direct",
        "durable":true, "auto_delete":false, "internal":false,
        "arguments":{}},
    {"name":"control", "vhost":"/", "type":"topic",
        "durable":true, "auto_delete":false, "internal":false,
        "arguments":{}},
    {"name":"char_count", "vhost":"/", "type":"direct",
        "durable":true, "auto_delete":false, "internal":false,
        "arguments":{}}],

"bindings":[]}
```

HP

Importing configuration - JSON

- Modify the vhost line as in the following snippet:
  - "vhosts":[{"name":"/"}, {"name":"book"}],
- Upload



localhost:55672/#/import-succeeded

User: guest

Virtual host: All

Import succeeded

Your definitions were imported successfully.

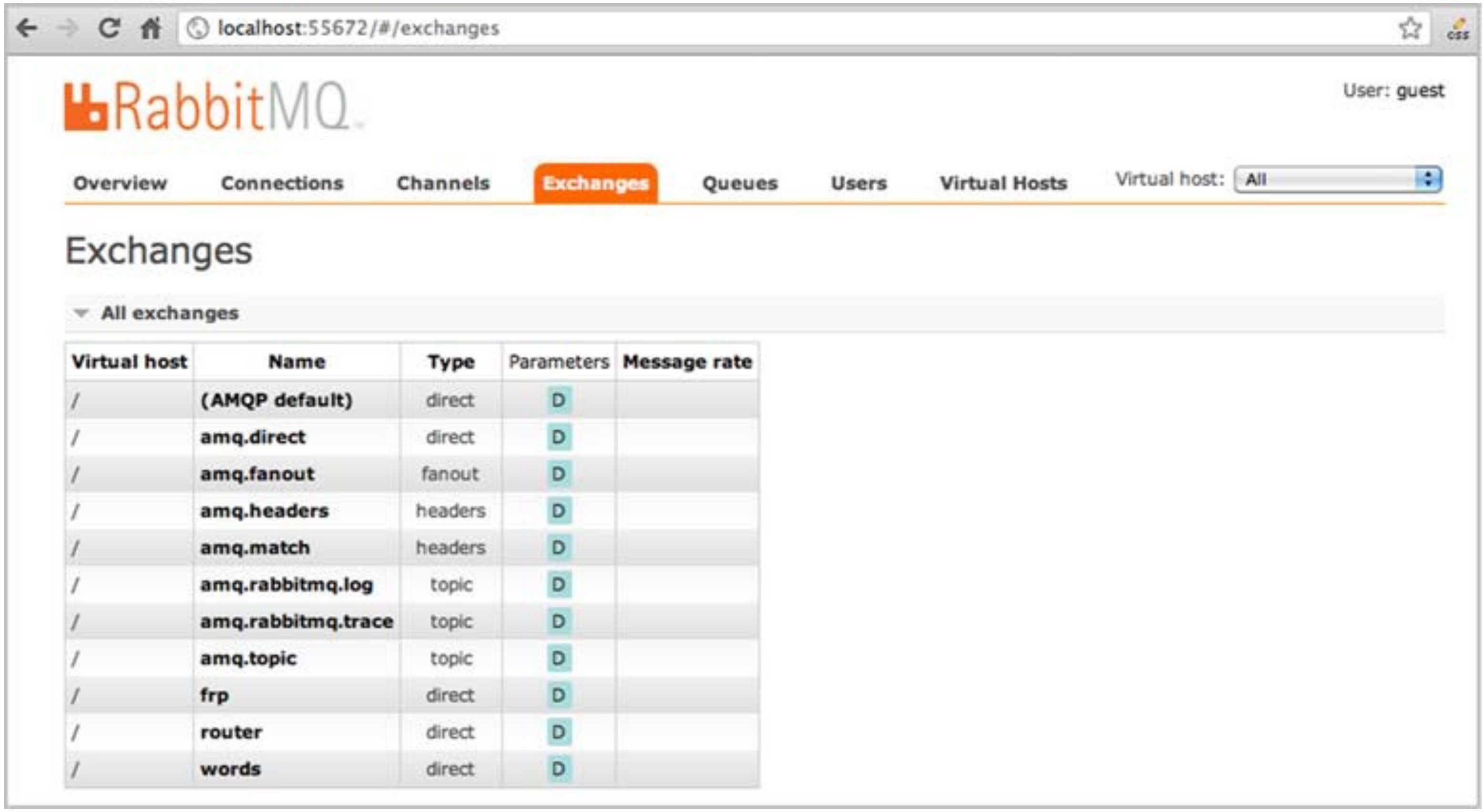
Update every 5 seconds

Last update: 2011-11-01 19:31:30

Web Console

- *Managing users from the web console will be discussed in the later part of the session*
  - Security and Access Control- Chapter

Managing exchanges and queues



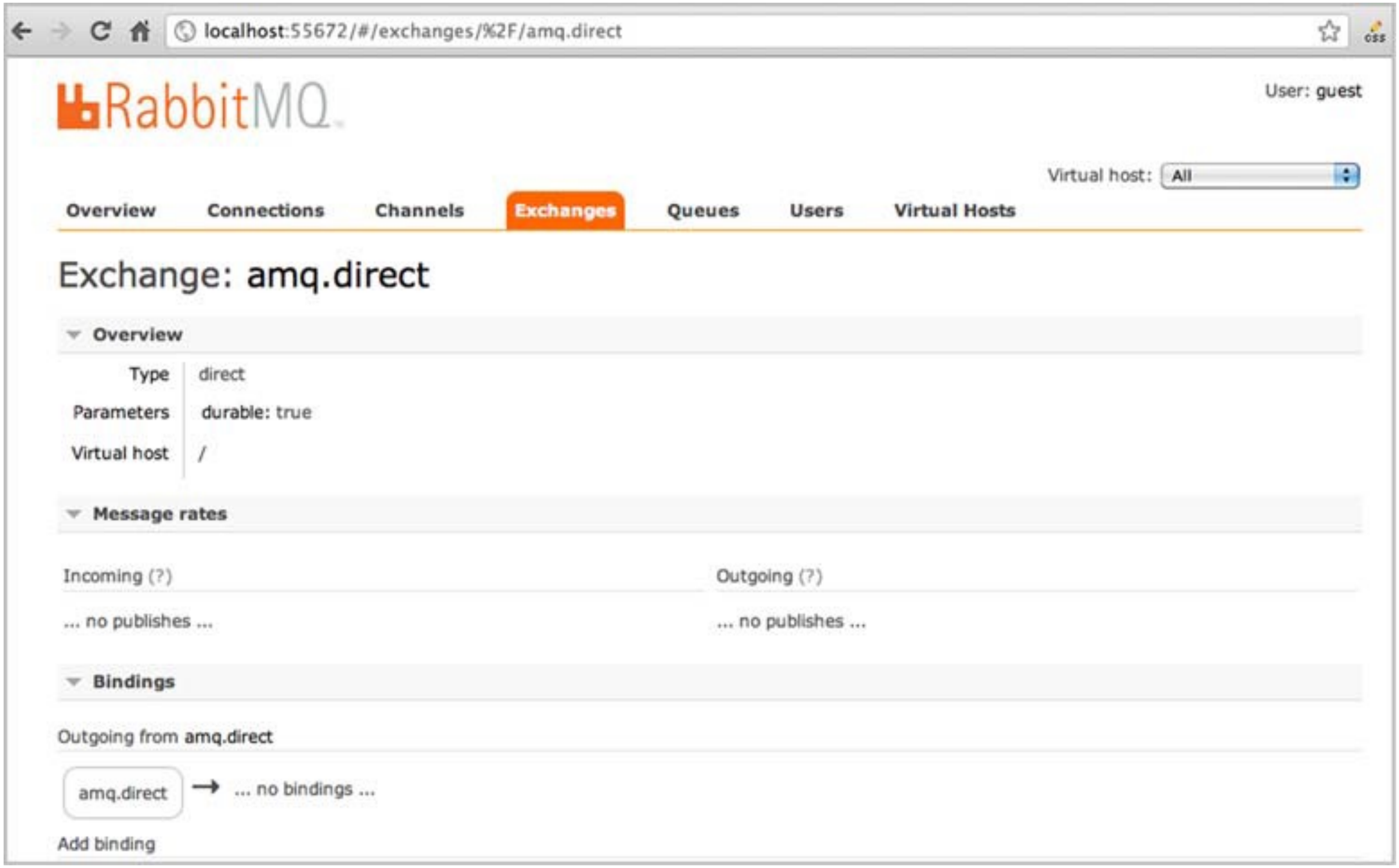
Exchanges

All exchanges

Virtual host	Name	Type	Parameters	Message rate
/	(AMQP default)	direct	D	
/	amq.direct	direct	D	
/	amq.fanout	fanout	D	
/	amq.headers	headers	D	
/	amq.match	headers	D	
/	amq.rabbitmq.log	topic	D	
/	amq.rabbitmq.trace	topic	D	
/	amq.topic	topic	D	
/	frp	direct	D	
/	router	direct	D	
/	words	direct	D	

HP

Exchange details



Exchange: amq.direct

Overview

Type	direct
Parameters	durable: true
Virtual host	/

Message rates

Incoming (?)

... no publishes ...

Outgoing (?)

... no publishes ...

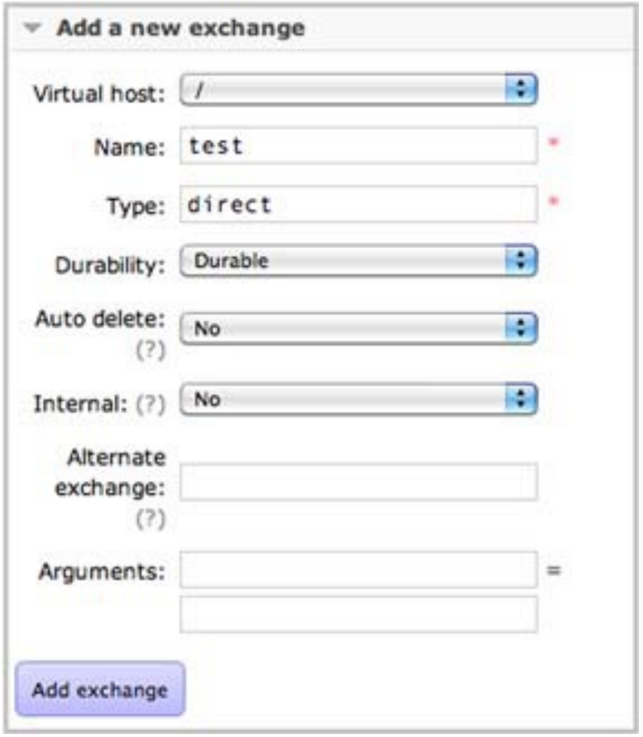
Bindings

Outgoing from amq.direct

amq.direct → ... no bindings ...

Add binding

Add new bindings and delete the exchange



Virtual host: /

Name: test *

Type: direct *

Durability: Durable

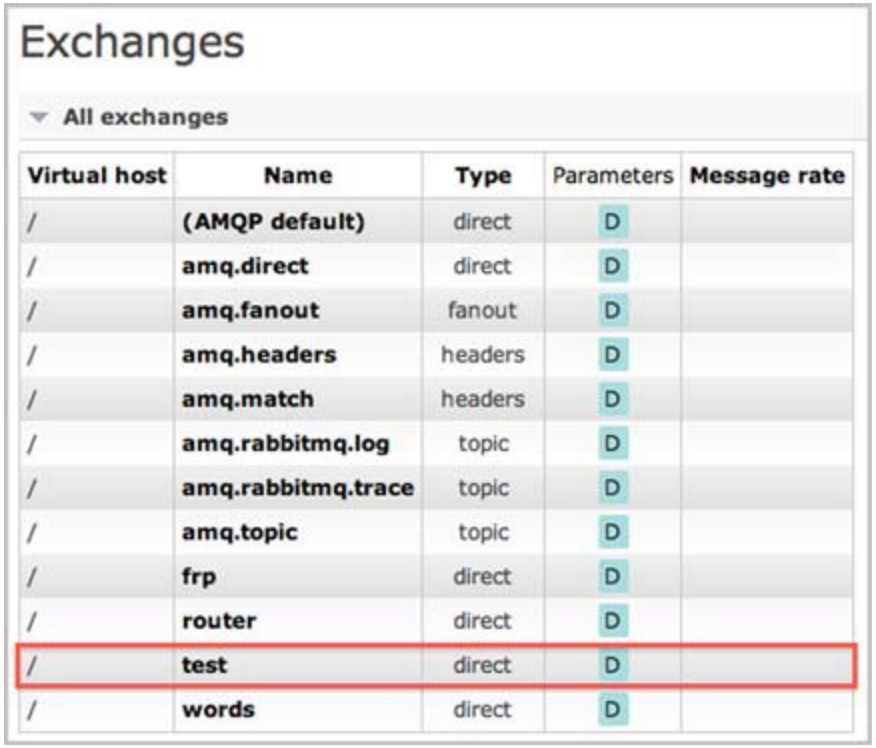
Auto delete: (?) No

Internal: (?) No

Alternate exchange: (?)

Arguments: =

Adding an exchange

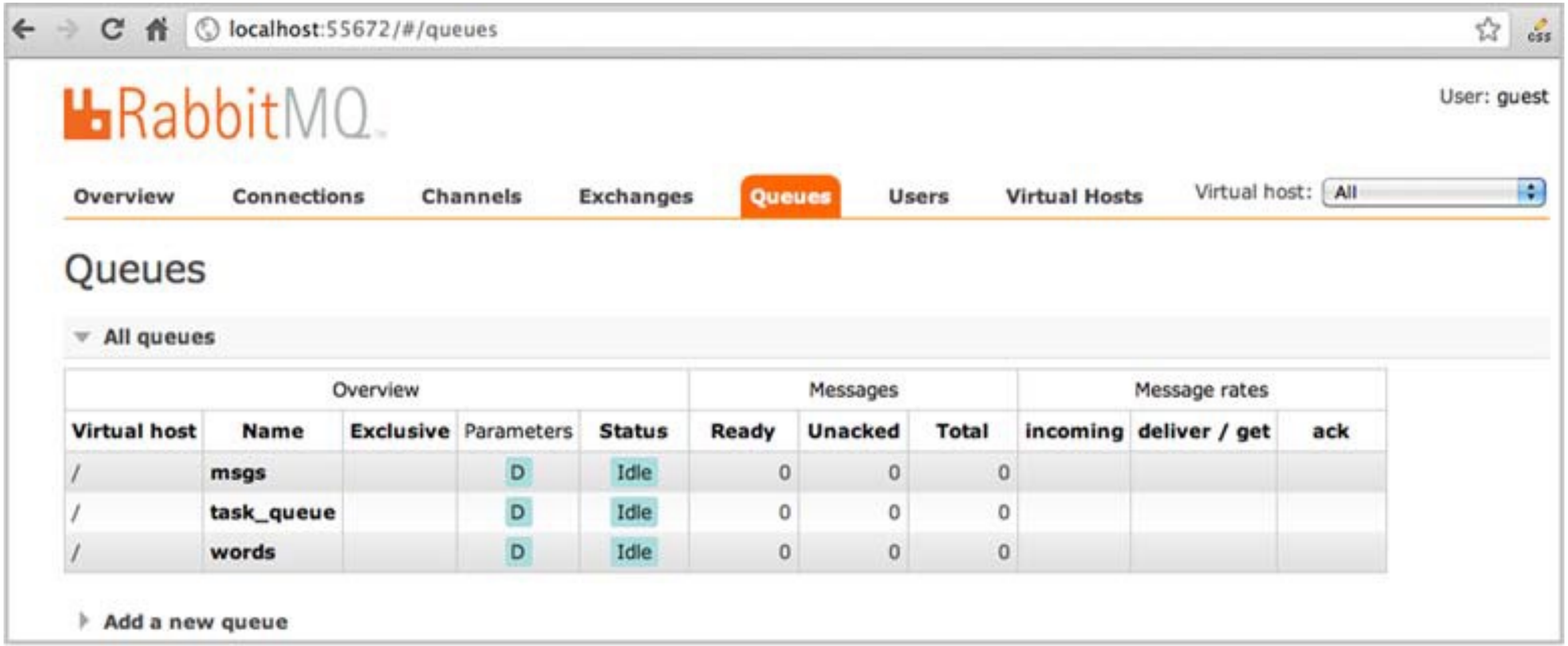

Exchanges

Virtual host	Name	Type	Parameters	Message rate
/	(AMQP default)	direct	D	
/	amq.direct	direct	D	
/	amq.fanout	fanout	D	
/	amq.headers	headers	D	
/	amq.match	headers	D	
/	amq.rabbitmq.log	topic	D	
/	amq.rabbitmq.trace	topic	D	
/	amq.topic	topic	D	
/	frp	direct	D	
/	router	direct	D	
/	test	direct	D	
/	words	direct	D	

New exchange

HP

Queues



Queues

All queues

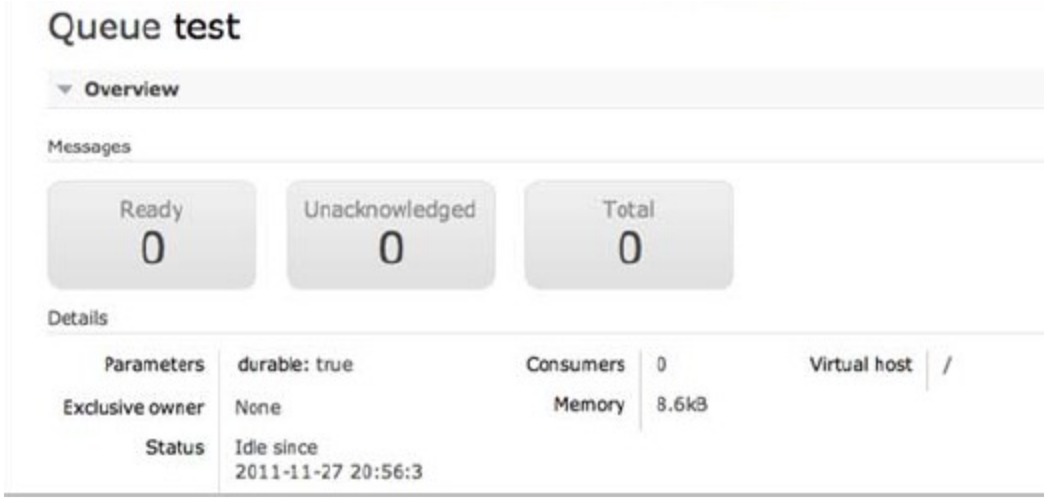
Overview					Messages			Message rates		
Virtual host	Name	Exclusive	Parameters	Status	Ready	Unacked	Total	incoming	deliver / get	ack
/	msgs		D	Idle	0	0	0			
/	task_queue		D	Idle	0	0	0			
/	words		D	Idle	0	0	0			

Add a new queue

Creating queues

- can delete it or purge it directly from the browser.
- can’t do from the rabbitmqctl script

Queue details



Queue test

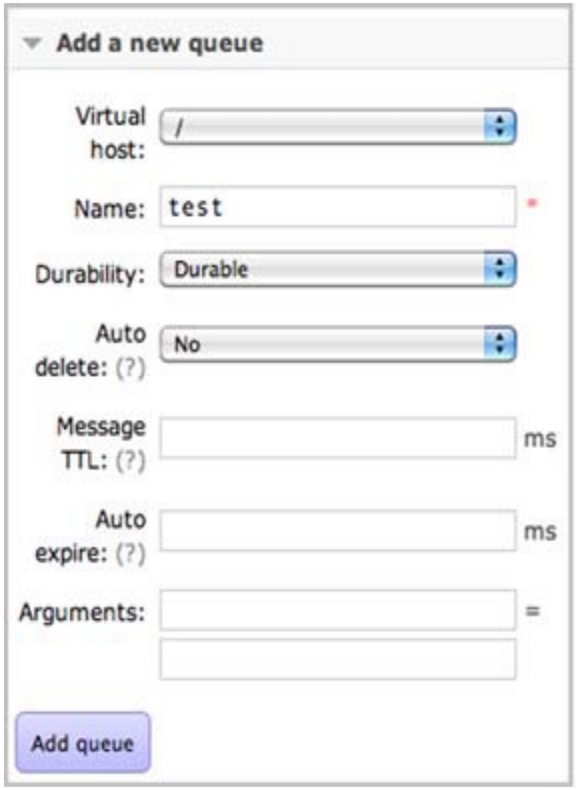
Overview

Messages

Ready	Unacknowledged	Total
0	0	0

Details

Parameters	durable: true	Consumers	0	Virtual host	/
Exclusive owner	None	Memory	8.6kB		
Status	Idle since 2011-11-27 20:56:3				



Add a new queue

Virtual host:	/
Name:	test *
Durability:	Durable
Auto delete: (?)	No
Message TTL: (?)	ms
Auto expire: (?)	ms
Arguments:	=

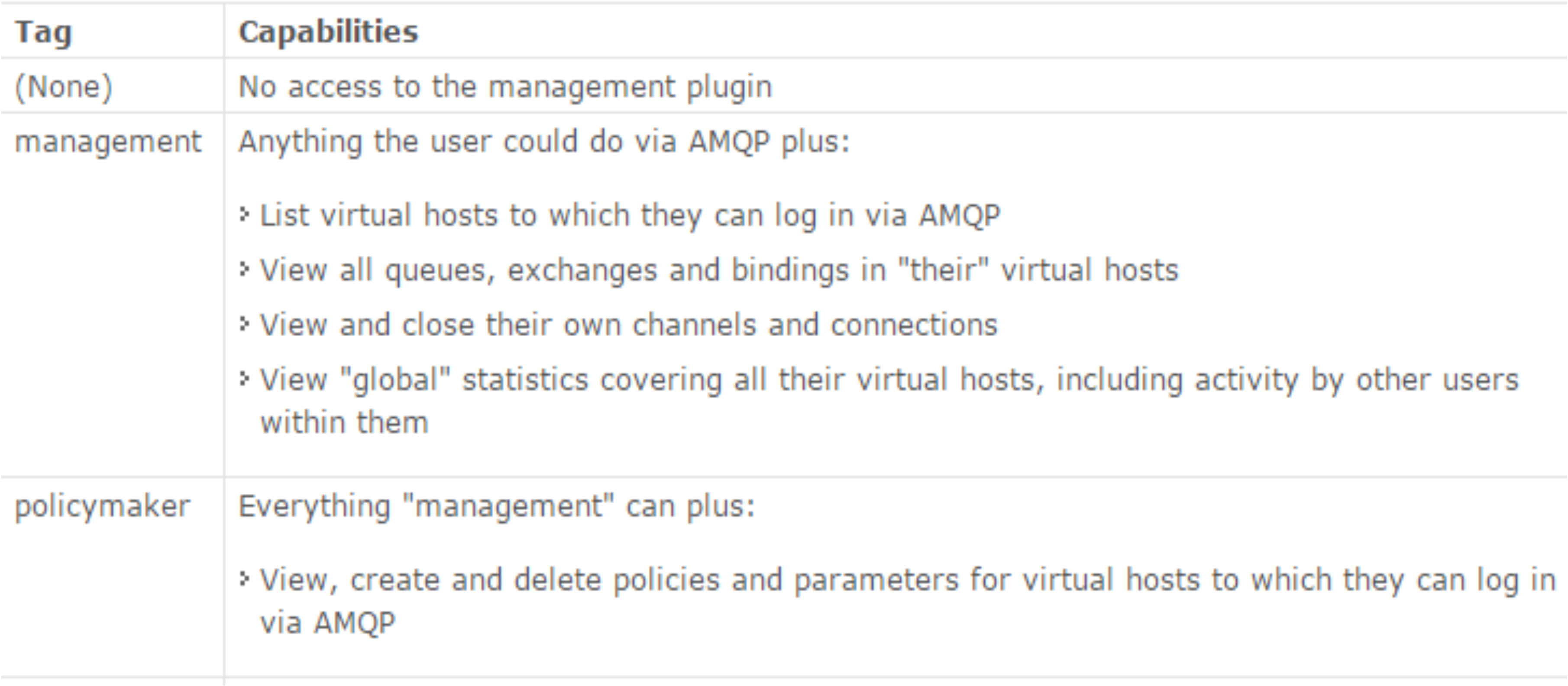
Adding a queue

HP

Permissions

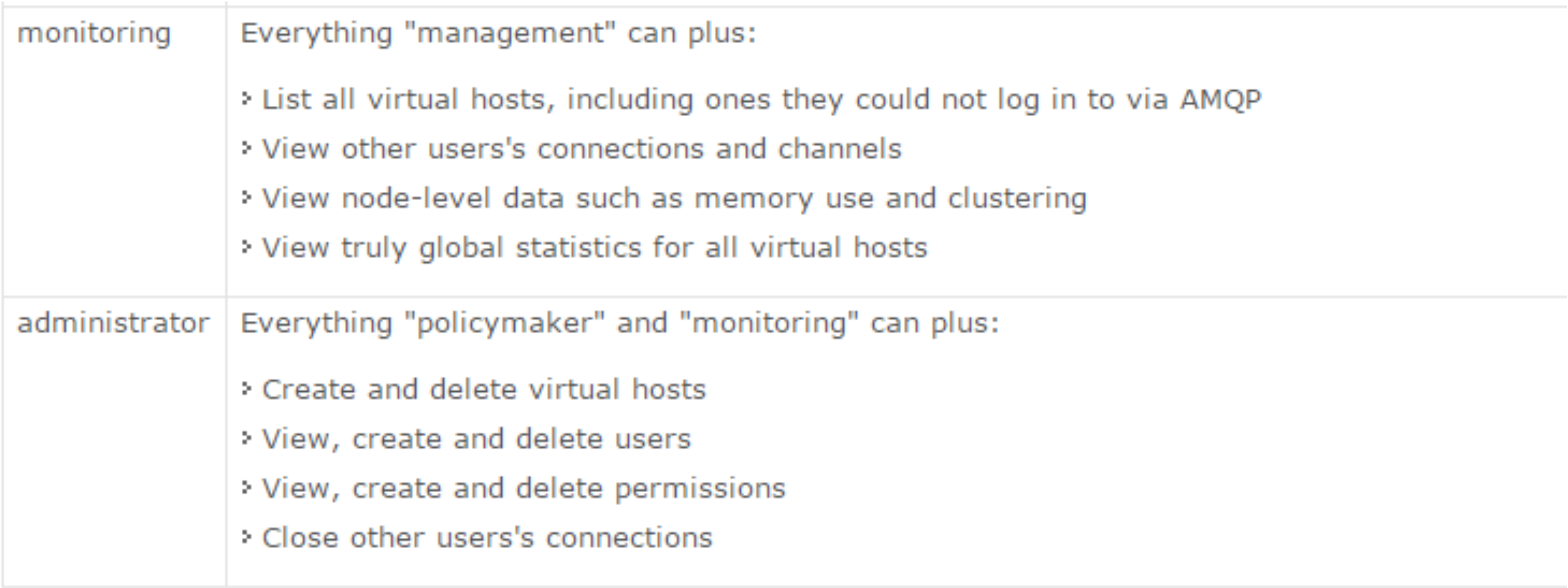
- Users can be given arbitrary tags
- The management plugin makes use of :
  - "management",
  - "policymaker",
  - "monitoring" and
  - "administrator".

Permissions



Tag	Capabilities
(None)	No access to the management plugin
management	Anything the user could do via AMQP plus: › List virtual hosts to which they can log in via AMQP › View all queues, exchanges and bindings in "their" virtual hosts › View and close their own channels and connections › View "global" statistics covering all their virtual hosts, including activity by other users within them
policymaker	Everything "management" can plus: › View, create and delete policies and parameters for virtual hosts to which they can log in via AMQP

Permissions..



monitoring	Everything "management" can plus: › List all virtual hosts, including ones they could not log in to via AMQP › View other users's connections and channels › View node-level data such as memory use and clustering › View truly global statistics for all virtual hosts
administrator	Everything "policymaker" and "monitoring" can plus: › Create and delete virtual hosts › View, create and delete users › View, create and delete permissions › Close other users's connections

Multitenancy with virtual hosts

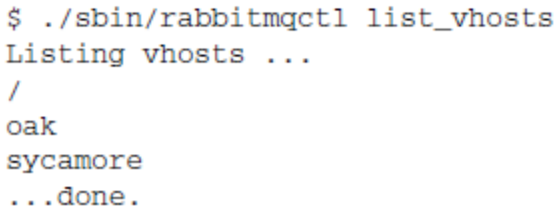
- The **vhost**.
  - able to create virtual message brokers called *virtual hosts (vhosts).*
  - *a mini-RabbitMQ server with its own queues,* exchanges, and bindings … and, more important, its own permissions.
  - you can safely use one RabbitMQ server for multiple applications
- fundamental to the concept of AMQP require to specify one when you connect.
- A default vhost called /

Multitenancy with virtual hosts

- *permissions are per vhost.*
- To create a vhost:
  - *rabbitmqctl add_vhost* [vhost_name],

    ,where *[vhost_name] is the vhost you want to create.*

    rabbitmqctl add_vhost henry



```
$ ./sbin/rabbitmqctl list_vhosts
Listing vhosts ...
/
oak
sycamore
...done.
```

Lab – Web Amin Console