



## How-to:

---

# Configure ALB to Load Balance and Accelerate Oracle E-Business Suite

### **jetNEXUS Solutions Limited**

Grove Business Park  
Cedar Court  
Waltham Road  
Maidenhead  
Berkshire  
SL6 3LW

Phone: 0870 382 5050 or International +44 (0) 1628 820 630

Fax: 0870 382 55 20 or International +44 (0) 1628 820 647

**Author:** Andrew Knowlson  
**Review:** Greg Howett  
**Version:** 2.4  
**Date:** 21<sup>st</sup> June 2010  
**Created:** 21<sup>st</sup> June 2010  
**Modified** 7th November 2011

Copyright © 2005 - 2011 jetNEXUS, Inc. All Rights Reserved.

© 2005-2011 jetNEXUS, Inc. All rights reserved. JetNEXUS and the jetNEXUS logo are registered trademarks of jetNEXUS, Inc...

jetNEXUS, Inc. reserves all ownership rights for the jetNEXUS ALB/ALB-X product line including software and documentation

**Limitations:** This document and all of its contents are provided as-is. jetNEXUS has made efforts to ensure that the information presented herein are correct, but makes no warranty, express or implied, about the accuracy of this information. If any material errors or inaccuracies should occur in this document, jetNEXUS will, if feasible, furnish appropriate correctional notices which Users will accept as the sole and exclusive remedy at law or in equity. Users of the information in this document acknowledge that jetNEXUS cannot be held liable for any loss, injury or damage of any kind, present or prospective, including without limitation any direct, special, incidental or consequential damages (including without limitation lost profits and loss of damage to goodwill) whether suffered by recipient or third party or from any action or inaction whether or not negligent, in the compiling or in delivering or communicating or publishing this document.

Oracle is a registered trademark of Oracle Corporation in the United States and other countries. All other trademarks and service marks are the property of their respective owners.

## jetNEXUS

---

- 1) Synopsis
- 2) Overview
- 3) Oracle E-Business suite Description
- 4) Oracle E-Business suite Diagram
- 5) Load Balanced Oracle E-Business suite Diagram
- 6) Oracle E-Business suite Load Balanced Tier Diagram
- 7) Load Balancing Methods?
- 8) Configuring a Channel Service
- 9) Adding another service on the same Channel
- 10) Oracle Server Health Monitoring
- 11) Oracle Application Tier Caching
- 12) Caching Rulebase
- 13) Creating a Caching Rulebase
- 14) Associating Domains to a Cache Rulebase

## Synopsis:

This document explains briefly how to use an ALB to load balance Oracle Traffic.

## Overview:

The ALB, ALB-X & ALB-VA are all Application Delivery Controllers (ADC) sometimes referred to as a next generation load balancer.

This document assumes that you are already familiar with the process using the ALB interface.

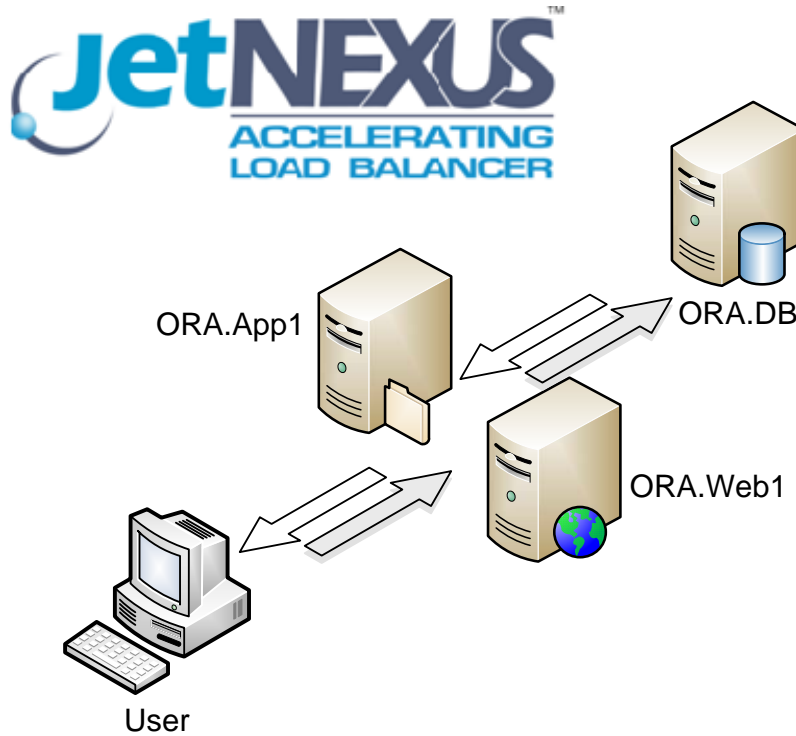
This document assumes that you are already familiar with the process of installing Oracle and you have an existing Oracle E-Business suite deployment running.

## Oracle E-Business suite Description:

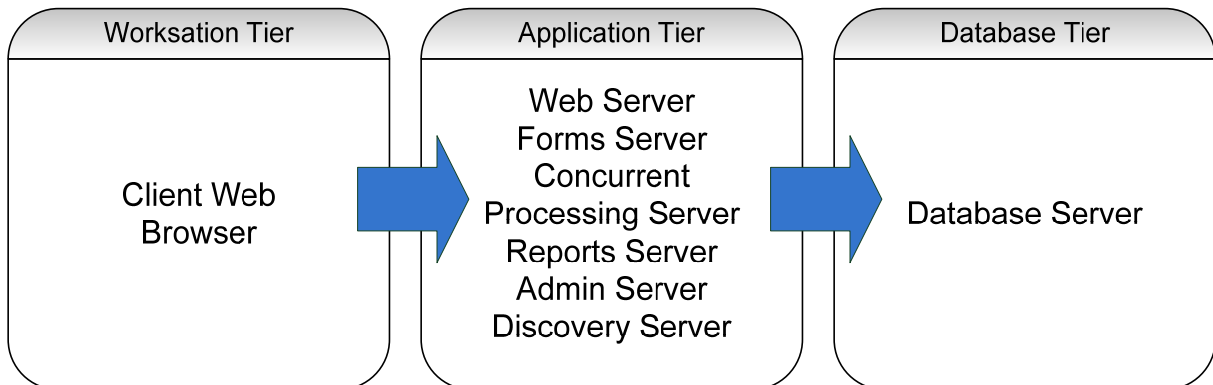
E-Business suite from Oracle consists of several product lines which utilise Oracle's Core Relational database management system technology.

Using the ALB-X you can provide layer 4 & 7 Load balancing, High Availability, at the application tier for the Forms server and Web server services.

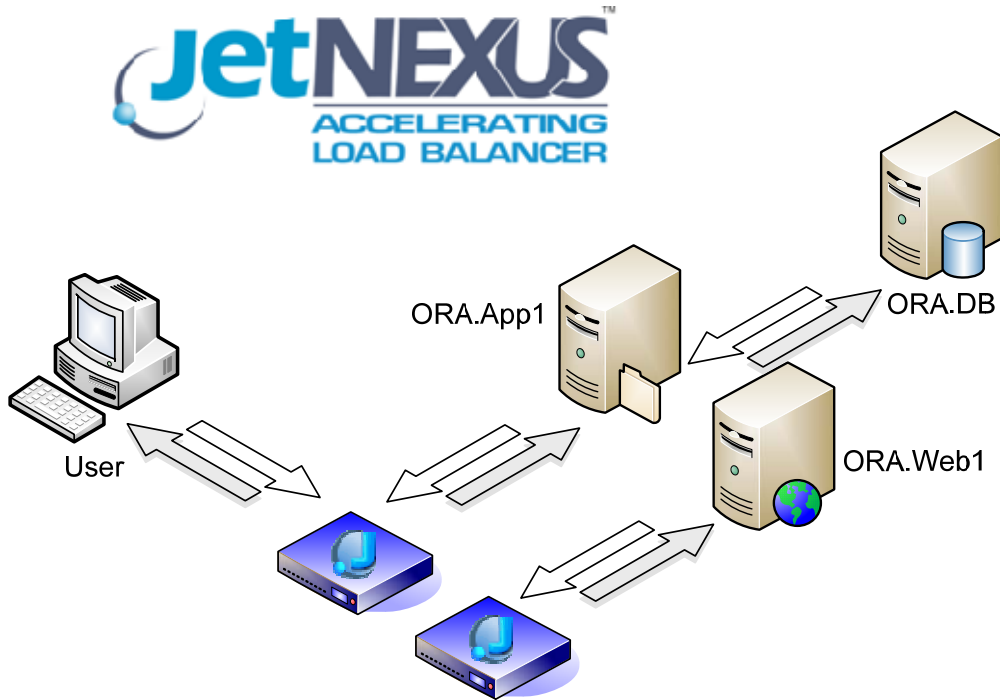
## Oracle E-Business suite Diagram



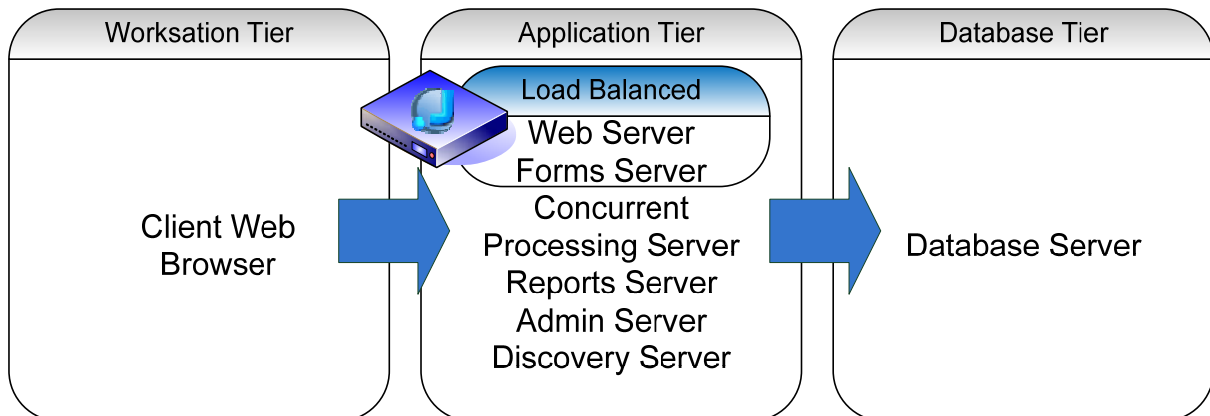
## Oracle E-Business suite Tier Diagram



## Load Balanced Oracle E-Business suite Diagram



## Oracle E-Business suite Load Balanced Tier Diagram



## What are Load Balancing Methods?

Typically a load balancing method or strategy is used to decide how the load balancer chooses where to send the request. Below are the methods used by jetNEXUS ALB:

**Least Connections:** The load balancer will keep track of the number of connections a back end server has and send the next request to the server with the least connections.

**Round Robin:** The simplest method, each back end server takes a turn.

**IP Based:** In this situation the clients IP address is used to select which back end server will receive the request. IP session persistence runs at layer 4 as such it can be used when load balancing non HTTP protocols. This method is useful for internal networks where the network topology is known and you can be confident that there are no “super proxies” upstream. If this is the case the all the requests will look like they are coming from one client, and as such the load would be uneven.

**Cookie Based:** This is the most popular persistence method for HTTP. In this situation, least connections load balancing is used for each first request. A cookie is inserted into the headers of the first http response. Thereafter, jetNEXUS ALB uses the client cookie to route traffic to the same back end server. Again this is used when the client must go to the same back end server each time.

**Classic ASP Session Cookie:** Active Server Pages (ASP) is a Microsoft server-side technology. With this option selected the ALB will maintain session persistence to the same server if an ASP cookie is detected and is found in its list of known cookies. If a new ASP cookie is detected then it will be load balanced using the least connections algorithm.

**ASP.NET Session Cookie:** ASP.NET is a Microsoft server-side technology. With this option selected the ALB will maintain session persistence to the same server if an ASP.NET cookie is detected and is found in its list of known cookies. If a new ASP.NET cookie is detected then it will be load balanced using the least connections algorithm.

**JSP Session Cookie:** Java Server Pages (JSP) is an Oracle server-side technology. With this option selected the ALB will maintain session persistence to the same server if a JSP cookie is detected and is found in its list of known cookies. If a new JSP cookie is detected then it will be load balanced using the least connections algorithm.

**JAX-WS Session Cookie:** Java web services (JAX-WS) is an Oracle server-side technology. With this option selected the ALB will maintain session persistence to the same server if a JAX-WS

cookie is detected and is found in its list of known cookies. If a new JAX-WS cookie is detected then it will be load balanced using the least connections algorithm.

**PHP Session Cookie:** Personal Home Page (PHP) is an open source server-side technology. With this option selected the ALB will maintain session persistence to the same server if a PHP cookie is detected.

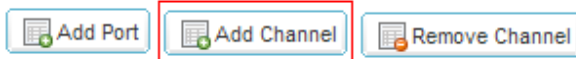
## Configuring a Channel Service

### Adding a Load balanced service

The IP service configuration screen is the main configuration screen for load balancing functionality and it can be found in (Setup → IP Services).



To configure a channel click the “Add Channel” button



This will now add a blank Channel IP service ready for configuration

**Channel Details**

Primary	Status	Enabled	IP Address	SubNet Mask	Port	Service Name	Service Type	Max. Conne...
		<input checked="" type="checkbox"/>					Accelerate H...	

This will add the fields ready to input the settings:

**Channel Details**

Primary	Status	Enabled	IP Address	SubNet Mask	Port	Service Name	Service Type	Max. Conne...
		<input checked="" type="checkbox"/>					Accelerate H...	

## Configuring a new Channel for Oracle Forms

Configure the new channel with the IP details for your new service.

In this example we will configure the following IP details, based on a One-Armed Configuration.

**IP Address:** 10.0.34.100 (Oracle Forms Server)  
**Subnet Mask:** 255.255.0.0  
**Ports:** 80  
**Data Method:** Accelerate  
**Primary:** n/a  
**Cache:** OFF  
**SSL:** No SSL  
**Service description:** Port 80 Traffic-Test  
**Monitoring:** Ping/ICMP Echo  
**Max connections:** n/a  
**LB Policy:** Least Connections  
**Pool:** n/a  
**Connection Server IP:** 10.0.34.11  
**Connection Server Port:** 80

Our new channel has been setup, click on the update button to save the configuration.

Server Monitoring:	TCP Connection	<input type="button" value="Update"/>
Load Balancing Policy:	Least Connections	
Connectivity:	Managed	
Caching Strategy:	Off	
SSL:	No SSL	
Enable Connection Pooling:	<input type="checkbox"/>	
Connection pool Size:	2000	

IP Services

**Channel Details**

Primary	Status	Enabled	IP Address	Subnet Mask	Port	Service Name	Service Type	Max. Connections
	●	<input checked="" type="checkbox"/>	10.0.34.100	255.255.0.0	80	Port 80 Traffic test	Accelerate HTTP	

---

**Destination** | **Actions**

**Content Server Details**

Content Server Group Name:

---

**Content Servers**

Status	Enabled	IP Address	Port
●	<input checked="" type="checkbox"/>	10.0.34.11	80

Our Channel has now been configured correctly; the Listening IP and Connection server IP are now shown as Green.

IP Services

**Channel Details**

Primary	Status	Enabled	IP Address	SubNet Mask	Port	Service Name
	●	<input checked="" type="checkbox"/>	10.0.34.100	255.255.0.0	80	Port 80 Traffic test

---

**Destination** | **Actions**

**Content Server Details**

Content Server Group Name:

---

**Content Servers**

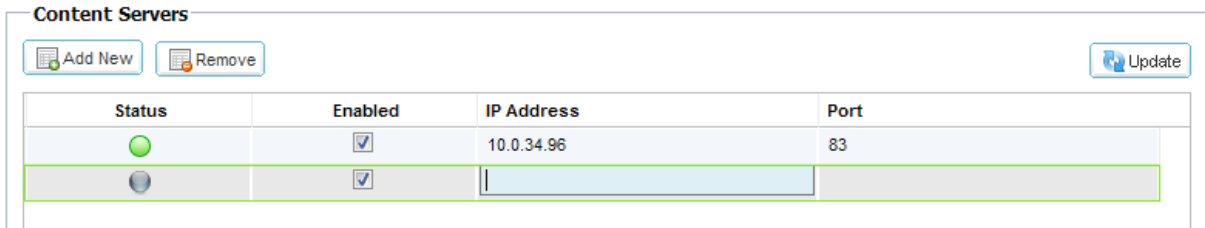
Status	Enabled	IP Address	Port
●	<input checked="" type="checkbox"/>	10.0.34.11	80

If you had additional Oracle Forms servers you can click on the + to add a secondary server

**Content Servers**

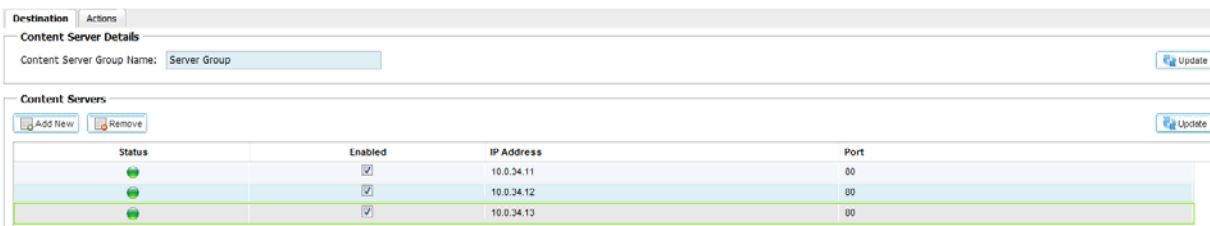
Status	Enabled	IP Address	Port
●	<input checked="" type="checkbox"/>	10.0.34.96	83

You can now add your additional Oracle Forms servers.



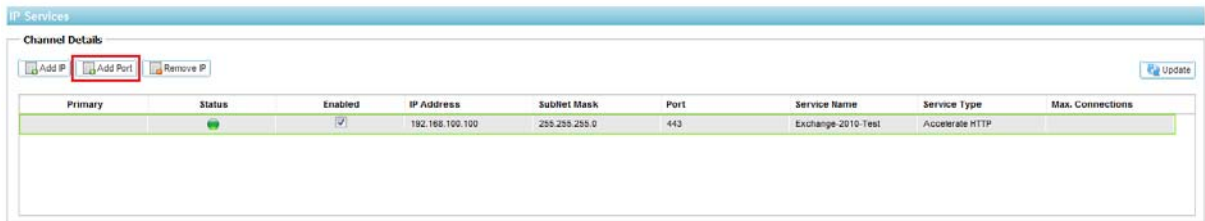
Click on the Update button next to the “Add Content server “to enable the new content server.

In the example below I have added my 3 Oracle Forms servers to my load balanced channel.

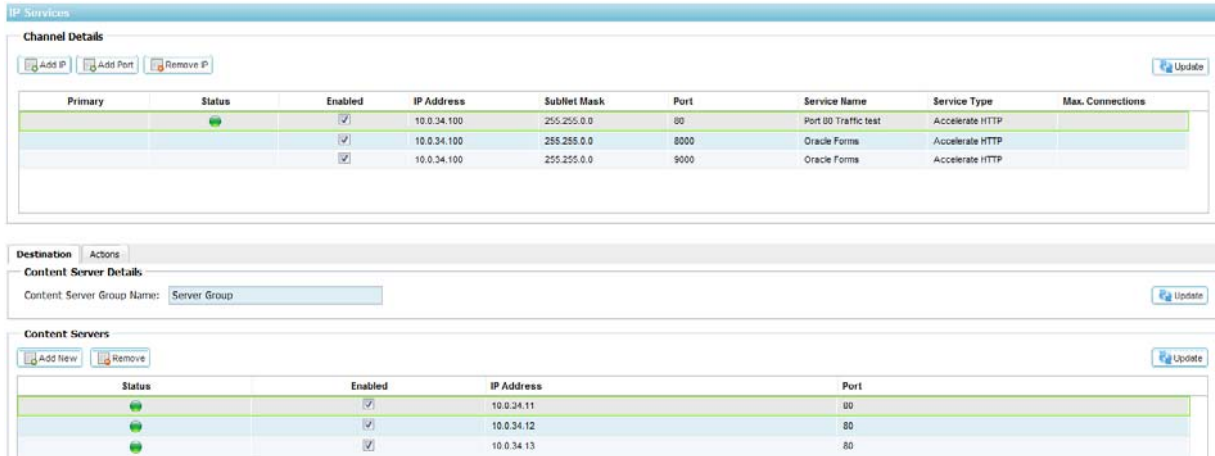


## Adding another service on the same Channel

To set up another service on the same IP address and a different port click the “Add Port” Oracle Forms uses ports 443, 8000 & 9000 so these will also need to be added.



This will add another grouping of settings similar to the first set. This time you don't need to specify the listening IP or subnet as it has already been added.



This now allows you to set up ports for 8000 & 9000 using the same channel IP and you can specify your Oracle forms servers again.


## Oracle Server Health Monitoring

When in a normal running situation the ALB will initiate active connections to the applications serves when a user makes a request. In addition to this the ALB can also perform active server monitoring. This active monitoring can be configured to perform more advanced test to validate if the web server can provide service. These tests occur irrespective of any load.

The choices for tests are as follows:

- None** - No active monitoring
- Ping** - Perform a ping
- HTTP 200 OK** - Check the response code form the web server is 200 OK
- HTTP Response** - Check the body of the response for a particular string

## Configuring Server Health Monitoring

Using the Navigation bar on the left of the web interface, go to (Configure → monitoring) this will open the  tab allowing you to access the monitoring options.

To add a new monitor you will be presented with the following screen

**Details**

Name	Description	Monitoring Method	Page Location	Required Content
200OK	Check home page for 200 ...	HTTP 200 OK	/	

You will need to fill in the Content Server Monitoring method screen with the following information.

**Name:**

Give the new method a useful name.

**Description:**

Give the new method a useful description .

**Monitoring Method:**

Type of Monitoring HTTP 200 ok or HTTP Response

**Page Location:**

The page location is the page that you would like to test from the ALB

In the example below I have configured a health monitor using HTTP Response to check for a test.asp page on my content servers.

**Details**

Name	Description	Monitoring Method	Page Location	Required Content
HealthCh...	This will check the conten server .asp page	HTTP 200 OK	/test.asp	

Next we need to apply this new method to our service. Go back to the IP services screen and select the channel service you want this monitor enabled on.

Next we need to apply this new method to our service. Go back to the IP services screen and select the new method from the dropdown and click on update.

## Oracle Application Tier Caching

Using the Navigation bar on the left of the web interface go to (Configure → Cache). This will open the Cache tab allowing you to access the caching configuration screen.

## How jetNEXUS Caching Works

Upon receipt of a request, the cache is searched for the requested page. The request is forwarded to a local content server group if the page cannot be found in the cache.

If it is in cache, but expired, a revalidation request is used with the most recent “If-Modified-Since” value. If the cache entry is OK (i.e. if the content matches “If-Modified-Since”) a cache local “304 Not Modified” is returned, otherwise a cache local “full response” is sent.

Note that the cache stores only uncompressed content as page acceleration is performed after the page is taken from the cache.

Upon filling the cache, the oldest (least recently used) content is retired from cache in order to make room for the new content. When the cache size meets the maximum size, or when the timer triggers a check, or when the “check cache” button is clicked, a cache check is performed.

Whilst the size of the cache is greater than the desired, content is removed from the cache as described above. Once the desired size is reached, the cache is left to grow naturally until the next check is triggered.

The Caching screen is divided into three parts:

### Parameters to configure overall cache behaviour

Cache	
Maximum Cache Size (MB):	<input type="text" value="50"/>
Desired Cache Size (MB):	<input type="text" value="30"/>
Default Caching Time (D/HH:MM):	<input type="text" value="1/00:00"/>
Cachable HTTP Response Codes:	<input type="text" value="200 203 301 304 410"/>
Cache Checking Timer (D/HH:MM):	<input type="text" value="03:00"/>
Cache-Fill Count:	<input type="text" value="20"/>

Check Cache Update

Force a check on the cache size

Clear Cache

Remove all items from the cache

Note.

The Check Cache button, clear cache of expired items no longer being served

The Clear Cache button, clear the cache of all content

### Parameters to defined Rule Bases to domains

**Apply Cache Rule**

Name	Image Url

### Parameters to define one or more caching Rule Bases

**Create Cache Rule**

Cache Content Selection Rulebases:

Name	Description	Add Condition	Conditions
Images	Caches most images	+	include *.jpg include *.gif include *.png

### Cache Settings

**Cache**

Maximum Cache Size (MB):

Desired Cache Size (MB):

Default Caching Time (D/HH:MM):

Cachable HTTP Response Codes:

Cache Checking Timer (D/HH:MM):

Cache-Fill Count:

The parameters related to overall caching behaviour are as follows:

**Maximum Cache Size (MB)** Maximum RAM that the cache can consume.

The jetNEXUS Cache is an in-memory cache that is also periodically backed onto hard disk to

maintain cache persistence after restarts, reboots and shutdowns. This means that the maximum cache size must fit within the memory footprint of the appliance (rather than disk space) and should be no more than half of available memory.

Recent jetNEXUS ALBs are equipped with 2GB of RAM while older models are fitted with 1GB. For 1GB of RAM the cache size should be no more than 512MB, while appliances with 2GB of RAM should have a cache size no larger than 1GB. Increasing the cache size beyond this limit could adversely affect page acceleration if the compressed content is large, so care must be exercised.

Calculating the best cache size for your site(s) will obviously depend on the amount of cacheable content you have, but you may find it equally effective to start with the default cache size (50MB). A periodic review of the stats for caching under “Monitor -> Statistics” (where the cache contents are described in terms of bytes used and percentage filled) will then help you decide whether to reduce or expand the maximum cache size. Be aware that this value is also complemented by a desired cache size (see below).

**Desired Cache Size (MB)** Optimum RAM that the cache will be trimmed to.

While the maximum cache size represents the absolute upper boundary of the cache, the desired cache size is intended as the optimum size that the cache should attempt to attain whenever an automatic or manual check on the cache size is made.

The gap between the maximum and desired cache size exists to accommodate the arrival and overlap of new content between periodic checks on cache size for the purpose of trimming expired content. Once again, it may be more effective to accept the default value (30 MB) and periodically review the size of the cache under “Monitor -> Statistics” for appropriate sizing.

Setting the desired and maximum cache sizes to the same figure will cause the cache to be trimmed with every request fetching new content into the cache. There are conditions where this may be desirable, but there will also be an impact on performance in that cache content is being shuffled in and out of the cache continuously.

**Default Caching Time (D/HH:MM):** Life of content without an explicit expiry value

The default caching time is the period content will be stored in the cache for items that don't have a “no-store” directive, but also have no explicit expiry time in the traffic header. The field entry takes the form “D/HH:MM” - so an entry of “1/00:00” (the default) means to store the item for one day, “01:00” for one hour and “00:01” for one minute.

**Cacheable HTTP Response Codes:** HTTP responses that will be cached

This field should be edited with caution as the most common cacheable response codes are already listed:

200 - Standard response for successful HTTP requests.

203 - Headers are not definitive, but are gathered from a local or a 3<sup>rd</sup> party copy.

301 - The requested resource has been assigned a new permanent URI.

304 - Not modified since the last request, and the locally cached copy should be used instead.

410 - Resource is no longer available at the server and no forwarding address is known.

**Cache Checking Timer (D/HH:MM):** Interval between cache trim operation

The default caching time is the period content will be stored in the cache for items that don't have a "no-store" directive, but also have no explicit expiry time in the traffic header. The field entry takes the form "D/HH:MM" - so an entry of "1/00:00" (the default) means to store the item for one day, "01:00" for one hour and "00:01" for one minute.

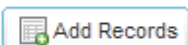
**Cache-Fill Count:** Count of 304s for a cached item before re-fetching

The default caching time is the period content will be stored in the cache for items that don't have a "no-store" directive, but also have no explicit expiry time in the traffic header. The field entry takes the form "D/HH:MM" - so an entry of "1/00:00" (the default) means to store the item for one day, "01:00" for one hour and "00:01" for one minute.

### Adding a Caching Rule

The (Configure → Cache) screen also supports the creation of custom rule bases that can later be applied against one or more domain names being managed by the cache (more on this in the last part of this section).

Name	Description	Add Condition	Conditions
Images	Caches most images	+	include *.jpg include *.gif include *.png

Press the  button to add a rule base and series of fields are displayed

**Create Cache Rule**

Cache Content Selection Rulebases: include  directory

Name	Description	Add Condition	Conditions
New Rule	New Rule Description	+	

Most caching rules can be added using the menu below, the options are explained below.

Cache Content Selection Rulebases: include  **directory**

- Directory:** Anything in the named directory anywhere on the site
- File:** Any file so named please include any file extension
- Anything starting:** The prefix to the URL (including the leading / character)
- Anything ending:** The suffix to the URL (including file name or trailing / )
- Anything containing:** A significant piece of the URL anywhere within it
- Exact match:** The exact URL of a piece of content
- All JPEG Images:**
- All GIF Images:**
- All Bitmap Images:**
- All PNG Images:**
- All HTML Pages:**
  
- Name:** Short name of the rule base
- Description:** Informative description of the rule base.
- Add Condition:** + to add rule
- Conditions:** Added by Cache content

selection

Name	Description	Add Condition	Conditions
New Rule	New Rule Description	+	

Most rules are "include" rules, but you can subsequently "exclude" subsets of content if required. Leave "include" selected for the first rule.

Cache Content Selection Rulebases:

Name	Description	Add Condition	Conditions
All jpeg images	Cache for all .jpeg images	+	include*.jpg

Then select the type of content you'd like to include in caching, in the example above we have decided to cache all \*.jpg images.

The Add condition + will add whatever content you have selected in the Cache Content Selection Rulebase screen and update the conditions section for you.

### Creating a Caching Rulebase

To Set up Caching you will need to access the settings in the following location (Configure → Cache). This will open the Caching tab

**Create Cache Rule**

Cache Content Selection Rulebases:


Name	Description	Add Condition	Conditions
All jpeg images	Cache for all .jpeg images	+	include*.jpg
New Rule	New Rule Description	+	

Our first example rule will be called “Cache Graphics” and will restrict content to the known graphic types already shown in the dropdown list (see below):

**Create Cache Rule**

Cache Content Selection Rulebases:

Name	Description	Add Condition	Conditions
All jpeg images	Cache for all .jpeg images	+	include*.jpg
Cache Graphics	Cache only Known graphic types	+	include*.png include*.jpg include*.gif include*.bmp


Pressing the  will add the “Cache Graphics” rulebase.


Our next example rule will be called “apps in URL” and will restrict caching to content whose text contains “/apps/” anywhere in the URL:

**Create Cache Rule**

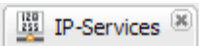
Cache Content Selection Rulebases:

Name	Description	Add Condition	Conditions
All jpeg images	Cache for all .jpeg images	+	include*.jpg
Cache Graphics	Cache only Known graphic types	+	include*.png include*.jpg include*.gif include*.bmp ...
Apps in URL	Cache only content that includes /apps	+	include */apps*

If there's a mistake in any of the values, you can either edit or delete the line later. Click the  button to make the changes take effect.

Remember that you can edit the content and click the  button again at any time to revise the rule base.

### Associating Domains to a Cache Rule base

You will need to first activate the caching checkbox in (Setup → IP Services). This will open the IP Service tab 

Select the channel you wish to enable caching on, and click on the actions tab.

**Channel Details**

Enabled	IP Address	SubNet Mask	Port	Service Name	Service Type	Max. Connecti...	Status
<input checked="" type="checkbox"/>	10.0.34.100	255.255.0.0	80	Port 80 Traffic-Test	Accelerate HTTP	10000	
<input checked="" type="checkbox"/>	10.0.34.100	255.255.0.0	443	Port 443 Traffic-Test	Accelerate HTTP	10000	
<input checked="" type="checkbox"/>					Accelerate HTTP	10000	

Destination | **Actions**

**Basic**

Flightpath

Server Monitoring:    
 Load Balancing Policy:   
 Select Caching rule:   
 Select SSL Certificate:   
 Enable Connection Pooling:   
 Connection pool Size:

You are then given 3 options

Select Caching rule:

Select SSL Certificate:

Enable Connection Pooling:

## By Host

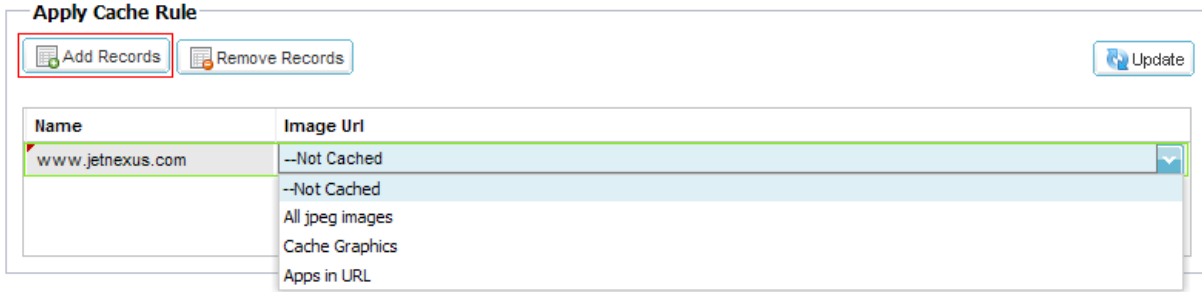
Once By host is enabled the ALB will then begin monitoring the domain names served to clients via the defined Channel.

Prior to activation, the middle section of the (Configure → Cache) screen was empty:

**Apply Cache Rule**


Name	Image Url

With caching activated and traffic passing through the ALBX, any domains served (though not yet cached) via the Channel will appear in the middle section (in our example [www.jetNEXUS.com](http://www.jetNEXUS.com)):




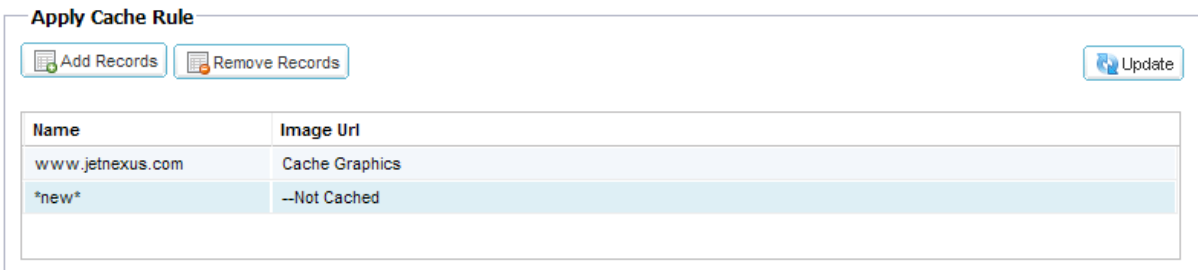
By pressing the “<-“arrow next to the domain name “[www.jetNEXUS.com](http://www.jetNEXUS.com)”, the domain is added to the cached domains list and a caching rulebase can be applied:

Note that the dropdown list of available rulebases includes the two examples we defined earlier in this section.

Select one of the rulebases in the list and press the  button at the right of this section to apply the change:



Domain names can also be added manually (rather than waiting for the ALBX to collect them for you) by pressing the  button:



Enter the domain name you want to add, select a caching rulebase from the dropdown list and press the “[Update]” button:

**Apply Cache Rule**

Name	Image Url
www.jetnexussolutions.com	Apps in URL
www.jetnexus.com	Cache Graphics

This completes the activation and configuration of ALBX by host caching.

### By Channel

Prior to activation, the middle section of the (Configure →Cache) screen was empty:

Once By Channel is enabled the ALB will then begin monitoring the Listening IP and port.  
(Example: 10.0.34.100:80)

**Apply Cache Rule**

Name	Image Url
10.0.34.100:80	Cache Graphics

You can monitor caching behaviour via the (Monitor →Cache) screen:

### Content Statistics

Caching		
<b>Content Caching</b>	<b>Hits</b>	<b>Bytes</b>
From Cache	= 0 / 0%	= 164.2 kB / 0.3%
From Server	= 0 / 0%	= 0 / 0%
Cache Contents	= 22 entries	= 164.2 kB / 0.3%

## Troubleshooting

Further help can be found on the jetNEXUS websites

<http://www.jetNEXUS.com/support.html>

<http://forum.jetNEXUS.com/>

## Contact Us

I hope you have found this How-To guide informative, but if you need any clarification or further information, please do not hesitate to get in contact with jetNEXUS Support:

E-mail [support@jetnexus.com](mailto:support@jetnexus.com)

Support +44 (0870) 382 5529

Phone+44 (0870) 382 5550

*Check out our blog*

<http://jetNEXUS.blogspot.com/>