

# **Together Application Cluster Apache Modul**

---

## **Together Application Cluster Apache Modul**

Together Teamlösungen EDV-Dienstleistungen GmbH

Elmargasse 2-4

A-1190

Vienna

Austria

+43 (0) 5 04 04 - 122

+43 (0) 5 04 04 - 11 122

<office@together.at>

<http://www.together.at/together/index.html>

**Copyright © 2006 Together Teamlösungen EDV-Dienstleistungen GmbH**

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without the prior written permission of the Together Teamlösungen EDV-Dienstleistungen GmbH.

Together Teamlösungen EDV-Dienstleistungen GmbH DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

---

---

## Table of Contents

1. Installation Notes for Together Application Cluster Apache Web Servers on Unix/Linux .....	1
2. Installation Notes for Together Application Cluster Apache Web Servers on Windows .....	10

---

# Chapter 1. Installation Notes for Together Application Cluster Apache Web Servers on Unix/Linux

## Introduction

This document describes the steps required to install Together Application Cluster on the Apache Web Server on Unix or Linux systems. Installation of this module follows the same general procedure as other Apache modules, so familiarity with the configuration of modules on apache servers is helpful.

The current implementation of the Together Application Cluster module requires that the Apache server be built to support DSO (Dynamic Shared Object) modules. See the sections below for further details.

## System Requirements

### Hardware Requirements

#### CPU

Intel 80386 or later CPU, or compatible for the Windows NT server (Pentium class machine or better recommended) or Sun Sparc (Ultra Sparc class machine or better recommended) or Any other hardware capable of running Unix or Linux and Apache with DSO support (Not officially supported, but it should work without much/any porting).

#### RAM

64 MB RAM / 128+ MB Recommended.

## OTHER

Network Interface that supports TCP/IP (Ethernet, Token Ring, etc.).

## Software Requirements

### Operating System

Linux / kernel version 2.2 or later with recent glibc support (RedHat 6.1 or later Recommended) or other Unix/Posix systems. (May or may not require some porting).

### Web Server

Apache Web Server [<http://httpd.apache.org/download.cgi>] version 1.3.9 or later. Earlier versions of Apache may or may not work with the module or they may require some porting. The current 'supported' version is 1.3.28, available from [www.apache.org](http://www.apache.org).

### Other

TCP/IP Networking must be available. Gnu Make (gmake) is required in order to build the module. Perl 5 is required by the Apache 'apxs' program, which is used to determine configuration of the apache server while building. Perl can be obtained for free from '<ftp://ftp.gnu.org/pub/gnu/perl>'.

## Planning

Before installing Together Application Cluster it will be helpful for you to obtain the following information about your planned site:

- The list of machines on which you will be running Enhydra MultiServer instances.
- The list of applications you intend to run on the Enhydra MultiServers.

- For each application:
  - The name of the machine and the port number on which each instance of the application will run. The port number is the 'port' used when configuring a connection of type 'EnhydraDirector' using the MultiServer Admin application.
  - The URL prefix to be used to identify the application (For example '/demoApp').
  - Whether the application requires 'session affinity'. That is, whether requests for the same session to an application should always be routed to the application server instance that created the session. Unless your application supports distribution of its sessions among other instances, you should assume that session affinity is needed. (However, if your application does not use session information at all, you don't need session affinity).
- The name of the Apache web server instance on which you wish to configure Together Application Cluster.
- The port number of the Apache web server instance on which you intend to configure Together Application Cluster. By default, this is port 80 (http).
- The location of the configuration files for Apache. On Unix this is often '/usr/local/apache/conf'.
- The location of the 'bin' executables directory for your Apache server. This is often '/usr/local/apache/bin'.

## System Preparation

Linux Installation:

- Install your system software according to your own preferences. TAC requires that TCP/IP networking be available.

Linux Configuration:

- Make sure TCP/IP networking is configured and available on at least one network interface.

## Together Application Cluster Files

Apache Extension: mod\_enhydra\_director.so

- This is an Apache extension module. It is placed in the Apache 'libexec' directory and is loaded with the AddModule and LoadModule directives in 'httpd.conf'. It does the work of forwarding application requests to the Enhydra server and of forwarding the server response to the client.

Together Application Cluster Configuration File: `enhydra_director.conf`

- This file contains the configuration for Together Application Cluster. By default, the '`mod_enhydra_director.so`' module above looks for this file in the same directory as the '`httpd.conf`' file.

Together Application Cluster Daemon: `edir_daemon`

- (Apache Only) This program runs in the background and monitors the state of the shared memory region that is used by child instances of the module to coordinate load balancing. This is needed in Apache because it uses many separate processes to allow 'multithreading' of simultaneous requests. The daemon is automatically launched when Apache starts, and automatically cleans up the shared memory and exits when Apache is stopped.

Together Application Cluster Status Utility: `edir_status`

- (Apache Only) This utility directly reads the shared memory region used by the Together Application Cluster module. It reports the current state of each configured application and Enhydra server.

Together Application Cluster Scoreboard: `enhydra_director.ipc`

- This is the 'scoreboard' that contains the shared memory data used by all of the apache child processes when they are handling Together Application Cluster requests. Normally, the contents of this file are kept in memory as an `mmap()` based shared memory region. This file also serves as the scoreboard lock file, using `fcntl()` based locking. Its normal home is in '`edir`' subdirectory of the Apache '`logs`' directory. It is normally only accessible to the Apache child user id.

Apache Error Log: Various (usually '`error_log`')

- This is the standard error log for the apache server. The Together Application Cluster handler logs all errors encountered while handling requests here. Certain startup messages are also logged here, however most startup errors are logged to the system log by the daemon.

System Log: Various (often '`/var/log/messages`' or '`/var/adm/messages`')

- If the daemon encounters errors while it is initializing it has no access to the Apache error log, so it logs the error to the system log using the '`daemon`' logging channel.

## Installing Together Application Cluster

These instructions assume that you will install the TAC files into your Apache installation directories. In order to build, you must first obtain, build, and install Apache 1.3.9 or later. Moreover, you must configure Apache so that DSO support (`mod_dso`) is enabled. The build scripts use the Apache '`apxs`' utility to

automatically obtain the correct installation directories for the Together Application Cluster components, so you should take note of where this utility is installed. Finally, you should make sure that Perl 5 is installed, since `apxs` is a perl script.

## Step 1: Obtain the Together Application Cluster source code

The Together Application Cluster source code is available under the Enhydra Public License (EPL) and is included with the full Enhydra source distribution. A smaller source distribution containing only the source code for the Together Application Cluster is included with the pre-built distributions of Enhydra 3.0 or later. Obtain the Together Application Cluster source code from either of the above sources and extract it into a working directory. Here is where you will find the source code.

If you are using the Enhydra full source tree:

- Extract 'enhydra-src3.0.tar.gz' into the current directory. You will find the Together Application Cluster source in `./Enhydra/modules/EnhydraDirector/src/`

If you are using the Enhydra pre-built distribution:

- Extract 'enhydra3.0.tar.gz' or 'enhydra3.0.zip' into the current directory.
- You will find the Together Application Cluster source archive in `./TAC/TAC-7.0.tar.gz` or `./TAC/TAC-7.0.zip`.
- Change directory to `./TAC` and extract one of the above archives into the current directory.
- The extracted source code will be in `./TAC-7.0`
- From where you extracted the Enhydra distribution `./director/enhydra-director1.0`

Now, change directory to the newly extracted Together Application Cluster source code directory root. This is the directory named 'src' that contains the subdirectories 'common', 'apache', 'nsapi', etc. From this point on we will call this directory '\$SRCROOT'.

## Step 2: Build the Together Application Cluster Apache module

If you installed Apache in the default location on Linux, this step should be easy. Just run the './configure' script with no arguments:

```
cd $SRCROOT/apache
./configure
make
make install
```

This will build the Together Application Cluster Apache module and install the 'mod\_enhydra\_director.so' file into the 'libexec' directory for your Apache distribution.



Option for 'configure': `--with-apxs=/full/path/to/program/apxs`

- This option allows the location of the Apache 'apxs' script to be specified if the default configure script cannot find it.

Option for 'configure': `--enable-debugging`

- Adds debugging messages to the build. This allows the 'EnhydraDirectorDebug' option to be specified in 'httpd.conf', or '<Options debug="0xNNNNNN"/>' to be specified in 'enhydra\_director.conf'. Enabling debugging can be useful if you experience problems with setting up the module.

After running 'make install', you will find 'mod\_enhydra\_director.so' in your Apache installation's 'libexec' directory. You will also find the file 'enhydra\_director.conf.default' in the 'conf' directory.

*Note:* If you like doing things the hard way: (Installing by hand) Just install the 'edir\_daemon', 'edir\_status', and 'enhydra\_director.conf' files wherever you like. Install the 'mod\_enhydra\_director.so' file in the 'libexec' directory for your Apache installation. Note that the 'libexec' directory may not be named 'libexec' on all installations. It's much easier to just './configure' and 'make install', using the `--with-apxs` if necessary to locate the 'apxs' script.

The following 'httpd.conf' options can be used to specify the locations of Together Application Cluster configuration and data files.

EnhydraDirectorConfigFile

- Specifies the full path to the 'enhydra\_director.conf' file. By default this is 'enhydra\_director.conf' in the 'conf' subdirectory of your Apache installation.

EnhydraDirectorLockFile

- Specifies the location of the file used for 'locking' the scoreboard during shared access. If you set this parameter, you should not need to set the 'EnhydraDirectorData' option, unless you are a developer or are porting to a strange platform, in which case you really know what you are doing.
- Setting this parameter may be necessary if you installed TAC on an NFS filesystem. Due to locking bugs in many/most NFS implementations, TAC will not work if the lock file is located on an NFS file system.
- By default, the lock file is located under the 'logs' directory. In order to allow the load balancing code to run as the Apache 'Child' user ID, the actual lock file is located under 'logs/edir', where 'edir' has permissions set to allow the child processes to read and write. The permissions of the 'logs' directory are unaffected.

EnhydraDirectorDataFile (Expert - Developers and Geeks Only)

- You should not ever need to set this parameter unless you are a developer or are experimenting with alternate shared memory modules (see `edir_shmem.h`). This allows the scoreboard shared data file to be specified as a different file than the lock file, or even as a System V IPC region.
- By default, the 'mmap' shared memory module uses the same file as the 'filesys' mutex module. (see `edir_mutex.h`). This means the locking and data sharing use the same file on most modern unix imple-

mentations, including Linux.

#### EnhydraDirectorDaemonPath (Advanced)

- If you must install the Together Application Cluster daemon (edir\_daemon) in a different location than the Apache 'bin' directory, this option can be used to specify the full path to the daemon program.

#### EnhydraDirectorDebug (Advanced)

- If you used '--enable-debugging' when building Together Application Cluster, then this option can be used to set the debugging mask in the module. The debug mask is also propagated to the daemon when it is started. See edir\_debug.h or enhydra\_director.conf.default for more information on possible values. To enable full, highly voluminous debugging, including hexdumps of all communication between the module and Enhydra, specify the value '0xffffffff'. Be forewarned that doing this will severely limit Together Application Cluster's performance, and will dump hundreds of megabytes of logging data to the Apache error log in a real hurry.
- This option is identical to <Option debug="0xNNNNNNNN"/> in enhydra\_director.conf, except that it gets set much earlier, and allows debugging of the code that loads the config file and scoreboard, as well as the daemon launching code.

## Configuring Your Application

This document assumes that you have already successfully built your application and installed your Enhydra multiserver and the application. It also assumes that you are using the MultiServer Admin application to add the connections for the application.

### Step 1: Create Together Application Cluster connections for your application.

- Bring up the MultiServer Administration page for your Enhydra server instance.
- Select your application in the application list.
- Click on the 'Connections' tab to manage connections for your application instance.
- Within the 'Connections' tab, click 'Create'. A 'new connection' dialog box will appear.
- In the dialog box, check 'EnhydraDirector' as the connection type.
- Fill in the URL prefix and port. Also, if session affinity is NOT to be enabled, change the session affinity box from 'true' to 'false'.

- Click 'OK' to accept the new connection.
- In the main MultiServer admin page, click the 'Save State' floppy-disk icon to save your changes to the MultiServer configuration file. Click 'OK' to accept the rewrite of the configuration file.

## Troubleshooting

Problem: The apache extension fails to load when Apache is started.

- Check that the 'httpd.conf' for correct configuration of the module, as described in the previous sections.
- Check the system log file for errors reported by the edir\_daemon program.
- Check the Apache error log for errors reported by the Together Application Cluster module.
- Ensure that the enhydra\_director.conf file exists and contains valid and syntactically correct configuration data. If the syntax is not valid, Together Application Cluster will report the specific errors and line numbers to the Apache error log.

Problem: Together Application Cluster starts up OK but I can't connect to my application.

- If you configured the <Status> section, check the status prefix for connection failures to your back-end Enhydra servers.
- Make sure that the prefixes configured in enhydra\_director.conf match the prefixes configured for your application on the Enhydra server. The prefixes MUST match EXACTLY, including case! Trailing slashes in the prefix are not significant and are ignored.
- Make sure that the application servers and ports in the enhydra\_director.conf file refer to valid and running instances of your application.
- Make sure your application is running on the Enhydra server.
- Your application must use the 'EnhydraDirector' connection method on ports that the Apache Together Application Cluster extension will dispatch requests to.
- It is highly recommended to configure the '<Status>' section in the configuration file so you can examine the runtime state and configuration of the handler.

Problem: Apache child processes dump core. (crash)

- Hopefully you will never see this. :)
- This is evidence of an error in the handler or filter DLL that needs the attention of Lutris engineers.
- There's also the possibility that there is a bug elsewhere in Apache, or in some other module. If possible, remove all unneeded modules from the 'httpd.conf' file to try to narrow the problem down to the Enhydra module.

- If you're an engineer and can give us stack traces from gdb, we would be VERY appreciative. ESPECIALLY stack traces from '--enable-debugging' compilations. :)
- Please report any bugs of this nature to the Enhydra.org folks, at [www.enhydra.org](http://www.enhydra.org). The more details you can provide about what led to the failure, the better. Also, machine state information and configuration information is helpful.
- If you have a way to reproduce the bug consistently, PLEASE include the necessary steps to do so if it is at all possible. This is VERY helpful when trying to resolve software problems.

---

# Chapter 2. Installation Notes for Together Application Cluster Apache Web Servers on Windows

## Introduction

This document describes the steps required to install Together Application Cluster on the Apache Web Server on Windows systems. Installation of this module follows the same general procedure as other Apache modules, so familiarity with the configuration of modules on apache servers is helpful.

The current implementation of the Together Application Cluster module requires that the Apache server be built to support DSO (Dynamic Shared Object) modules. See the sections below for further details.

## System Requirements

### Hardware Requirements:

#### CPU

Intel 80386 or later CPU, or compatible for the Windows NT server. (Pentium class machine or better recommended)

#### RAM

64 MB RAM / 128+ MB Recommended

## OTHER

Network Interface that supports TCP/IP (Ethernet, Token Ring, etc.)

## Software Requirements:

### Operating System

Microsoft Windows NT 4.0, Service Pack 6 or later.

### Web Server

Apache Web Server version 1.3.9 or later. Earlier versions of Apache may or may not work with the module or they may require some porting. The current 'supported' version is 1.3.23, available from [www.apache.org](http://www.apache.org).

### Other

TCP/IP Networking must be available.

Visual C++ 6.0 is required in order to build the module.

## Planning

Before installing Together Application Cluster it will be helpful for you to obtain the following information about your planned site:

- The list of machines on which you will be running Enhydra MultiServer instances.
- The list of applications you intend to run on the Enhydra MultiServers.

- For each application:
  - The name of the machine and the port number on which each instance of the application will run. The port number is the 'port' used when configuring a connection of type 'EnhydraDirector' using the MultiServer Admin application.
  - The URL prefix to be used to identify the application. (For example '/demoApp')
  - Whether the application requires 'session affinity'. That is, whether requests for the same session to an application should always be routed to the application server instance that created the session. Unless your application supports distribution of its sessions among other instances, you should assume that session affinity is needed. (However, if your application does not use session information at all, you don't need session affinity)
- The name of the Apache web server instance on which you wish to configure Together Application Cluster.
- The port number of the Apache web server instance on which you intend to configure Together Application Cluster. By default, this is port 80 (http).
- The location of the configuration files for Apache. On Windows this is often 'C:\Program Files\Apache Group\Apache\conf'.
- The location of the 'bin' executables directory for your Apache server. This is often 'C:\Program Files\Apache Group\Apache\bin'.

## System Preparation

Windows NT Installation:

- Install Windows NT 4.0 and Service Pack 6 according to the Microsoft documentation and your organization's requirements. Together Application Cluster Requires:

TCP/IP Networking Support

Windows NT/2000/XP Configuration:

- If you have not already done so, configure TCP/IP networking on at least one of the network interfaces on your system.

## Together Application Cluster Files

Apache Extension: mod\_enhydra\_director.so

- This is an Apache extension module. It is placed in the Apache 'module' directory and is loaded with the AddModule and LoadModule directives in 'httpd.conf'. It does the work of forwarding application requests to the Enhydra server and of forwarding the server response to the client.

Together Application Cluster Configuration File: enhydra\_director.conf

- This file contains the configuration for Together Application Cluster. By default, the 'mod\_enhydra\_director.so' module above looks for this file in the same directory as the 'httpd.conf' file.

Apache Error Log: Various (usually 'error\_log')

- This is the standard error log for the apache server. The Together Application Cluster handler logs all errors encountered while handling requests here. Certain startup messages are also logged here.

## Installing Together Application Cluster

These instructions assume that you will install the TAC files into your Apache installation directories. In order to build, you must first obtain, build, and install Apache 1.3.9 or later. Moreover, you must configure Apache so that DSO support (mod\_dso) is enabled.

### Step 1: Obtain the Together Application Cluster source code or download precompiled binaries.

Obtain the Together Application Cluster source code or download precompiled binaries.

### Step 2: Build the Together Application Cluster Apache module.

To build the Win32 version of the Apache module, start Visual Studio, and open the Together Application Cluster Apache workspace:

<TAC>/src/apacheWin32/ApacheDirectorWin32.dsw

Set 'EnhydraDirectorApaWin32' as the active project, then set the active configuration to 'Debug' or 'Release' according to your needs. If you installed Apache server in a non-default location, you also need to edit the build settings (C Preprocessor) to set the correct 'include' directories for the apache header files. At this point, just do a 'Build All' and rebuild the .SO. In the same directory as the workspace, there is a copy of 'enhydra\_director.conf.default'. The newly built .SO will be found in the 'Debug' or 'Release'



subdirectory, according to the usual Visual Studio convention.

## Step 3: Install the Together Application Cluster files. (Win32)

Regardless of how you obtain the final .so and files, install:

<Apache>\modules\mod\_enhydra\_director.so <Apache>\conf\enhydra\_director.conf.default

## Configuring Your Application

This document assumes that you have already successfully built your application and installed your Enhydra multiserver and the application. It also assumes that you are using the MultiServer Admin application to add the connections for the application.

## Step 1: Create Together Application Cluster connections for your application.

- Bring up the MultiServer Administration page for your Enhydra server instance.
- Select your application in the application list.
- Click on the 'Connections' tab to manage connections for your application instance.
- Within the 'Connections' tab, click 'Create'. A 'new connection' dialog box will appear.
- In the dialog box, check 'EnhydraDirector' as the connection type.
- Fill in the URL prefix and port. Also, if session affinity is NOT to be enabled, change the session affinity box from 'true' to 'false'.
- Click 'OK' to accept the new connection.
- In the main MultiServer admin page, click the 'Save State' floppy-disk icon to save your changes to the MultiServer configuration file. Click 'OK' to accept the rewrite of the configuration file.

## Configuring Together Application Cluster

Step 1: Locate the 'httpd.conf' file.

Find out where 'httpd.conf' is located for your Apache installation. Usually this is something like 'C:\Program Files\Apache Group\conf\httpd.conf'. However, Apache is very flexible in how it can be installed, so this may differ.

Step 2: Determine where Apache stores its DSO modules.

This is usually 'modules' under the Apache installation root. For example, on many Apache installations, the DSO module directory is 'C:\Program Files\Apache Group\Apache\modules'.

Step 3: Add the Together Application Cluster module to the Apache configuration.

- Open 'httpd.conf' with your favorite editor.
- Search to the last line in the file starting with 'LoadModule'. (The default 'httpd.conf' should at least have a comment line, but if not even a commented LoadModule line is found, just go to the end of the file.)
- Using the last component of the 'modules' directory located in step 2, add a line to the list of Load-Modules directives (if any), that looks like:

```
LoadModule enhydra_director_module modules/mod_enhydra_director.so
```

- If the last component of you Apache installation's 'modules' path ends in something else (say, 'mod'), then you would instead add the line:

```
LoadModule enhydra_director_module mod/mod_enhydra_director.so
```

- If there are other 'LoadModule' lines, they usually will contain the correct path to the modules path, so you can use the other directives as a guide.

NOTE: If you are using 'mod\_rewrite', then you should make sure that the 'LoadModule rewrite\_module ...' directive comes BEFORE the 'LoadModule enhydra\_director\_module ...' directive if you intend to use 'mod\_rewrite' to redirect URLs to Together Application Cluster. Also you should use the 'mod\_rewrite' '[PT]' tag to force pass-through of redirected URLs.

- Now, search for the last line starting with 'AddModule'. Again, if such a line does not even exist as a commented-out line, then just go to the end of the file.
- Add the line:

```
AddModule mod_enhydra_director.c
```

- Add any desired Enhydra-specific directives such as 'EnhydraDirectorDebug' to the 'httpd.conf' file. These directives are optional in almost all cases.
- Save the 'httpd.conf' file. Together Application Cluster is now configured!

Step 4: Create the Together Application Cluster configuration file.

The Together Application Cluster configuration file for Apache is normally located in the same directory as 'httpd.conf'. However, you can specify the 'EnhydraDirectorConfigFile' directive in httpd.conf to specify a different file.

The full description of the syntax and all of the directives available in the configuration file is beyond the scope of this document. However, the file 'enhydra\_director.conf.default' contains several useful examples that can be easily copied to match your site setup. The comments within the default file also describe most of the available options. An important thing to note is that the configuration file is based on

an XML DTD (EnhydraDirectorConfig.dtd) and is strictly validated. This means the file MUST contain valid XML, and it MUST follow the format defined in the DTD. Case is important in all section and property names, and order is important among different sections.

Step 5: Restart the Apache server.

Step 6: Verify correct startup.

Use 'Test Configuration' from Apache Program Group.

Step 7: Verify connectivity

- Start up a web browser.
- If you have configured a <Status> section in the configuration, you can connect to its URL prefix to ensure that the handler is functioning. See the default configuration file for more information on the <Status> section.
- Connect to one of your configured application prefixes.
- Errors encountered while handling requests are reported in the Apache error log.

## Troubleshooting

Problem: The apache extension fails to load when Apache is started.

- Check that the 'httpd.conf' for correct configuration of the module, as described in the previous sections.
- Check the system log file for errors reported by the edir\_daemon program.
- Check the Apache error log for errors reported by the Together Application Cluster module.
- Ensure that the enhydra\_director.conf file exists and contains valid and syntactically correct configuration data. If the syntax is not valid, Together Application Cluster will report the specific errors and line numbers to the Apache error log.

Problem: Together Application Cluster starts up OK but I can't connect to my application.

- If you configured the <Status> section, check the status prefix for connection failures to your back-end Enhydra servers.
- Make sure that the prefixes configured in enhydra\_director.conf match the prefixes configured for your application on the Enhydra server. The prefixes MUST match EXACTLY, including case! Trailing slashes in the prefix are not significant and are ignored.
- Make sure that the application servers and ports in the enhydra\_director.conf file refer to valid and running instances of your application.
- Make sure your application is running on the Enhydra server.

- You application must use the 'EnhydraDirector' connection method on ports that the Apache Together Application Cluster extension will dispatch requests to.
- It is highly recommended to configure the '<Status>' section in the configuration file so you can examine the runtime state and configuration of the handler.