



# Caché Parameter File Reference

Version 2017.2  
2020-06-26

*Caché Parameter File Reference*

Caché Version 2017.2 2020-06-26

Copyright © 2020 InterSystems Corporation

All rights reserved.

InterSystems, InterSystems IRIS, InterSystems Caché, InterSystems Ensemble, and InterSystems HealthShare are registered trademarks of InterSystems Corporation.

All other brand or product names used herein are trademarks or registered trademarks of their respective companies or organizations.

This document contains trade secret and confidential information which is the property of InterSystems Corporation, One Memorial Drive, Cambridge, MA 02142, or its affiliates, and is furnished for the sole purpose of the operation and maintenance of the products of InterSystems Corporation. No part of this publication is to be used for any other purpose, and this publication is not to be reproduced, copied, disclosed, transmitted, stored in a retrieval system or translated into any human or computer language, in any form, by any means, in whole or in part, without the express prior written consent of InterSystems Corporation.

The copying, use and disposition of this document and the software programs described herein is prohibited except to the limited extent set forth in the standard software license agreement(s) of InterSystems Corporation covering such programs and related documentation. InterSystems Corporation makes no representations and warranties concerning such software programs other than those set forth in such standard software license agreement(s). In addition, the liability of InterSystems Corporation for any losses or damages relating to or arising out of the use of such software programs is limited in the manner set forth in such standard software license agreement(s).

THE FOREGOING IS A GENERAL SUMMARY OF THE RESTRICTIONS AND LIMITATIONS IMPOSED BY INTERSYSTEMS CORPORATION ON THE USE OF, AND LIABILITY ARISING FROM, ITS COMPUTER SOFTWARE. FOR COMPLETE INFORMATION REFERENCE SHOULD BE MADE TO THE STANDARD SOFTWARE LICENSE AGREEMENT(S) OF INTERSYSTEMS CORPORATION, COPIES OF WHICH WILL BE MADE AVAILABLE UPON REQUEST.

InterSystems Corporation disclaims responsibility for errors which may appear in this document, and it reserves the right, in its sole discretion and without notice, to make substitutions and modifications in the products and practices described in this document.

For Support questions about any InterSystems products, contact:

**InterSystems Worldwide Response Center (WRC)**

Tel: +1-617-621-0700

Tel: +44 (0) 844 854 2917

Email: [support@InterSystems.com](mailto:support@InterSystems.com)

# Table of Contents

<b>About This Book</b> .....	<b>1</b>
<b>1 Introduction to the Caché Parameter File</b> .....	<b>3</b>
1.1 Filename and Location .....	3
1.2 Caché Configurations .....	3
1.3 Specifying a CPF File .....	4
1.4 Editing the Active CPF File .....	4
1.5 Configuration Security .....	4
1.6 Parameter File Format .....	5
1.6.1 Spaces .....	5
1.6.2 Sections .....	5
1.6.3 Parameters .....	5
1.7 Parameter Descriptions .....	6
1.8 New or Obsolete Parameters .....	6
1.9 List of Sections and Parameters .....	6
<b>[ComPorts]</b> .....	<b>7</b>
COMn .....	8
<b>[config]</b> .....	<b>11</b>
LibPath .....	12
MaxServerConn .....	13
MaxServers .....	14
Path .....	15
bbsiz .....	16
console .....	17
errlog .....	18
globals .....	19
gmheap .....	20
history .....	21
ijcbuff .....	22
ijcnum .....	23
jrnbufs .....	24
locksiz .....	25
memlock .....	26
netjob .....	28
nlstab .....	29
overview .....	30
pijdir .....	31
routines .....	32
udevtabsiz .....	33
wijdir .....	34
zfheap .....	35
<b>[ConfigFile]</b> .....	<b>37</b>
Version .....	38
<b>[Databases]</b> .....	<b>39</b>
Database .....	40

<b>[Debug]</b> .....	<b>43</b>
Dumpstyle .....	44
<b>[Devices]</b> .....	<b>45</b>
Devices .....	46
<b>[DeviceSubTypes]</b> .....	<b>49</b>
DeviceSubTypes .....	50
<b>[ECP]</b> .....	<b>53</b>
ClientReconnectDuration .....	54
ClientReconnectInterval .....	55
ServerTroubleDuration .....	56
<b>[ECPServers]</b> .....	<b>57</b>
ECPServers .....	58
<b>[IO]</b> .....	<b>59</b>
MagTape .....	60
Other .....	61
File .....	62
Mnemonic_TTY .....	63
<b>[Journal]</b> .....	<b>65</b>
AlternateDirectory .....	66
BackupsBeforePurge .....	67
CurrentDirectory .....	68
DaysBeforePurge .....	69
FileSizeLimit .....	70
FreezeOnError .....	71
JournalFilePrefix .....	72
JournalcspSession .....	73
<b>[LicenseServers]</b> .....	<b>75</b>
LicenseServers .....	76
<b>[MagTapes]</b> .....	<b>77</b>
MagTapes .....	78
<b>[Map]</b> .....	<b>79</b>
Global .....	80
Package .....	82
Routine .....	83
<b>[MapMirrors]</b> .....	<b>85</b>
MapMirrors .....	86
<b>[MapShadows.Name]</b> .....	<b>89</b>
MapShadows.Name .....	90
<b>[Mirrors]</b> .....	<b>91</b>
Mirrors .....	92
<b>[MirrorMember]</b> .....	<b>95</b>
AgentAddress .....	96
AsyncMemberGUID .....	97
AsyncMemberType .....	98
AsyncUseSystemPurgeInterval .....	99

JoinMirror .....	100
SystemName .....	101
ValidatedMember .....	102
VirtualAddressInterface .....	103
<b>[Miscellaneous] .....</b>	<b>105</b>
AsyncDisconnectErr .....	106
AsynchError .....	107
BreakMode .....	108
CollectResourceStats .....	109
DisconnectErr .....	110
EnableLongStrings .....	111
FileMode .....	112
GlobalKillEnabled .....	113
IEEEError .....	114
LineRecall .....	115
LogRollback .....	116
MVDefined .....	117
NodeNameInPid .....	118
NullSubscripts .....	119
OldZU5 .....	120
OpenMode .....	121
PopError .....	122
RefInKind .....	123
ScientificNotation .....	124
SetZEOF .....	125
ShutDownLogErrors .....	126
SwitchOSdir .....	127
SynchCommit .....	128
TelnetNUL .....	129
TruncateOverflow .....	130
Undefined .....	131
UseNagleAlgorithm .....	132
ViewPastData .....	133
ZaMode .....	134
ZDateNull .....	135
<b>[Monitor] .....</b>	<b>137</b>
PatrolCollectionInterval .....	138
PatrolDisplayMode .....	139
PatrolEnabled .....	140
PatrolTopProcesses .....	141
SNMPEnabled .....	142
WMIEnabled .....	143
<b>[Namespaces] .....</b>	<b>145</b>
Namespace .....	146
<b>[Shadows] .....</b>	<b>147</b>
Shadow .....	148
<b>[SQL] .....</b>	<b>149</b>
AllowRowIDUpdate .....	150

BiasQueriesAsOutlier .....	151
Comment .....	152
DBMSSecurity .....	153
DDLDefineBitmapExtent .....	154
DDLFinal .....	155
DDLNo201 .....	156
DDLNo30 .....	157
DDLNo307 .....	158
DDLNo311 .....	159
DDLNo315 .....	160
DDLNo324 .....	161
DDLNo333 .....	162
DDLSQLOnlyCompile .....	163
DDLUseSequence .....	164
DefaultSchema .....	165
DelimitedIds .....	166
DropDelete .....	167
ECPSync .....	168
ExtrinsicFunctions .....	169
FastDistinct .....	170
IdKey .....	171
IdTrxFrom .....	172
IdTrxTo .....	173
JavaClassPath .....	174
JavaHome .....	175
JDBCGatewayAddress .....	176
JDBCGatewayJVMArgs .....	177
JDBCGatewayLog .....	178
JDBCGatewayPort .....	179
JDBCGatewayUsePassphrase .....	180
LockThreshold .....	181
LockTimeout .....	182
ODBCVarcharMaxlen .....	183
QueryProcedures .....	184
ReferentialChecks .....	185
SaveMAC .....	186
TCPKeepAlive .....	187
TimePrecision .....	188
TODATEDefaultFormat .....	189
<b>[SqlSysDatatypes] .....</b>	<b>191</b>
System-Defined SQL Datatypes .....	192
<b>[SqlUserDatatypes] .....</b>	<b>195</b>
User-Defined SQL Datatypes .....	196
<b>[Startup] .....</b>	<b>197</b>
CallinHalt .....	198
CallinStart .....	199
CliSysName .....	200
DBSizesAllowed .....	201
DefaultPort .....	202

DefaultPortBindAddress .....	203
EnsembleAutoStart .....	204
ErrorPurge .....	205
FIPSMODE .....	206
IPv6 .....	207
JobHalt .....	208
JobServers .....	209
JobStart .....	210
MaxCacheTempSizeAtStart .....	211
MaxConsoleLogSize .....	212
ProcessHalt .....	213
ProcessStart .....	214
ShutdownTimeout .....	215
SystemHalt .....	216
SystemStart .....	217
TempDirectory .....	218
TerminalPrompt .....	219
WebServer .....	220
WebServerName .....	221
WebServerPort .....	222
WebServerURLPrefix .....	223
ZSTU .....	224
<b>[Telnet] .....</b>	<b>225</b>
DNSLookup .....	226
Port .....	227

# List of Tables

Table 1–1: Typical Location of the Default Caché Parameter File ..... 3



# About This Book

This book describes the settings in the Caché Parameter file, `cache.cpf`, also called the CPF file, which determine the Caché environment. Many of these parameters can also be set in the Management Portal; each of these settings includes a cross reference to the corresponding portal page.

This book contains the following chapters. Except for the introduction and the appendix, each section in the book corresponds to a section in the CPF file.

- [Introduction to the Caché Parameter File](#)
- [ComPorts](#)
- [config](#)
- [ConFigFile](#)
- [Databases](#)
- [Debug](#)
- [Devices](#)
- [DeviceSubTypes](#)
- [ECP](#)
- [ECPServers](#)
- [IO](#)
- [Journal](#)
- [LicenseServers](#)
- [MagTapes](#)
- [Map](#)
- [MapMirrors](#)
- [MapShadows.Name](#)
- [Mirrors](#)
- [MirrorMember](#)
- [Miscellaneous](#)
- [Monitor](#)
- [Namespaces](#)
- [Shadows](#)
- [SQL](#)
- [SqlSysDatatypes](#)
- [SqlUserDatatypes](#)
- [Startup](#)
- [Telnet](#)

- The detailed [Table of Contents](#)

# 1

## Introduction to the Caché Parameter File

When Caché starts, it reads configuration values from the text file `cache.cpf`, the *Caché parameter file*, also called the CPF file. This file is a record of a particular Caché configuration. This chapter describes:

- The syntax and purpose of each line of text in the CPF file.
- How to choose values for parameters in the CPF file.
- The system default for every parameter in a CPF file.
- How to start Caché using a specific CPF file.
- How to change the CPF file using the Management Portal, Config class APIs, or a text editor.

### 1.1 Filename and Location

A Caché parameter file name can be any valid file name with a `.cpf` extension. A default parameter file called `cache.cpf` is provided with every new installation of Caché.

A Caché parameter file must reside in the *Caché installation directory*. This is the directory into which you installed Caché. The exact pathname varies by platform and by choice of directory name during Caché installation. The following table provides some examples.

**Table 1–1: Typical Location of the Default Caché Parameter File**

Platform	Pathname
Windows	C:\install-dir\cache.cpf . By default C:\Intersystems\Cache\cache.cpf.
UNIX® and Linux	/bin/install-dir/cache.cpf
Mac	/bin/install-dir/cache.cpf

### 1.2 Caché Configurations

You can keep as many parameter files as you like in the Caché installation directory. Each represents a different *Caché configuration*. You can start Caché with different configurations as needed for developing and testing new features, for

troubleshooting problems, and for running a production system. For example, on Windows, if the Caché installation directory is `C:\MyCache` you might have the following CPF files:

```
C:\MyCache\cache_original.cpf
C:\MyCache\production.cpf
C:\MyCache\development.cpf
C:\MyCache\testapps.cpf
C:\MyCache\cache_customerbug.cpf
```

At shutdown, the system automatically saves the last known error-free configuration to a file called `_LastGood_.cpf` in the installation directory. You can use this file, if you need to, for recovery purposes.

## 1.3 Specifying a CPF File

Each installation of Caché is called an *instance*. Each instance has a name, which is the name you assigned to Caché when you installed it. The default instance name is `CACHE`.

You can start Caché using any CPF file contained in the installation directory. To start Caché with a CPF file other than `cache.cpf`, specify the name on the command line.

The following command starts Caché with a CPF file called `mycache.cpf`:

```
ccontrol start SPECIALCONFIG mycache.cpf
```

Where:

- `ccontrol start` starts a Caché instance from the command line
- `SPECIALCONFIG` is the Caché instance name
- `mycache.cpf` is the parameter file (that is, the configuration) to use

To use a different configuration for the Caché instance, you must stop Caché and start it again with a different parameter file.

For further details about the Caché command line, including many additional `ccontrol` command line options, see the “[Using Multiple Instances of Caché](#)” chapter in the *Caché System Administration Guide*.

## 1.4 Editing the Active CPF File

When you edit and save configuration settings using the Management Portal, the settings are saved to the CPF file that was used when you started that Caché instance. Your changes in the portal overwrite existing values in the file.

You can also edit and save the CPF file using a text editor. We recommend that you save a backup copy of the CPF file before editing it.

## 1.5 Configuration Security

If Caché startup detects that the Caché parameter file has been modified by a text editor since the last time Caché was started, and if [Configuration Security is enabled](#), Caché startup requests a username and password to validate the changes. The username supplied must have `%Admin_Manage:Use` privileges. If the appropriate username and password cannot be provided, Caché allows the operator to choose as follows:

1. Re-enter the username and password.
2. Start using the last known good configuration.
3. Abort startup.

If the operator chooses option 2, Caché renames the parameter file that was invoked at startup (*file.cpf*) with the suffix *\_rejected* (*file.cpf\_rejected*). Caché then overwrites the *file.cpf* with the last known good configuration (*\_LastGood\_.cpf*) and starts up using this configuration.

For more information on other systemwide security parameters, see the “[System Management and Security](#)” chapter in the *Caché Security Administration Guide*.

## 1.6 Parameter File Format

A Caché parameter file is a line-oriented, UTF-8 text file. Each line ends with a carriage return and line feed. Long items cannot be continued on a following line. A line in the file may be any of the following:

- An empty line made up of zero or more spaces and tabs
- The name of a file section enclosed in square brackets []
- A Caché configuration parameter and its value(s)

### Comments

You can add a comment by beginning a line with a semi-colon.

### 1.6.1 Spaces

In general, spaces at the beginning and end of lines are without effect. Spaces within the line are usually considered significant. The best practice is to use no spaces in the line except where they are meaningful components of strings.

### 1.6.2 Sections

Related settings are collected into sections. The beginning of a section is marked by a line consisting of the name of the section enclosed in square brackets. For example:

```
[Devices]
```

All lines after the section name, up to the next section name (or the end of file), are in the current section.

### 1.6.3 Parameters

Each line that is not a section name is the definition of a parameter. Each parameter line uses the following syntax, where *keyword* is a parameter name and *value* is a string:

```
keyword=value
```

When there is a set of similar items to configure, the *keyword\_#* is the name of the item and the *value* is a string. Examples include namespaces, databases, devices, and anything else of which there is a group or set of similar items to configure, one per line. The syntax is usually:

```
keyword_1=value  
keyword_2=value  
keyword_3=value  
.  
.  
.
```

The syntax for the *value* string varies widely from parameter to parameter. The string may indicate true or false using 1 or 0; it may be a number of bytes, or a number of megabytes; it may be a single value, or it may contain multiple values separated by a delimiter character on the same line. If there is a delimiter within the string, it may be a comma, semicolon, tilde (~), slash (/), colon, or some combination of these, depending on the parameter.

The organization of this document follows the order in which sections and parameters typically appear in the file.

## 1.7 Parameter Descriptions

Each parameter description in this book includes the following information:

- *Synopsis*—The CPF file section in which this parameter is valid, followed by the name of the parameter and a synopsis of its syntax.
- *Description*—A formal description of the parameter and its possible values.
- *Examples*—Examples of the parameter line as it might be seen in various CPF files.
- *Range of Values*—Minimum, maximum, and default values for the parameter.
- *Management Portal*—How to change the value of this parameter using the browser-based Management Portal. If there is no portal equivalent, the section indicates this.

## 1.8 New or Obsolete Parameters

With each Caché release, a small number of parameters are removed or added to the .cpf file. This document only describes parameters for the current release. If you are an experienced Caché user, you might not recognize every parameter in this document and you might be familiar with .cpf file parameters that are now removed.

Parameters are added or removed because:

- New features require new configuration settings.
- Obsolete features have had their settings removed.

## 1.9 List of Sections and Parameters

To see a list of all parameters and sections in the Caché parameter file, in the order in which they usually appear in the file, see the “[Table of Contents](#)” at the beginning of this book.

# [ComPorts]

This topic describes the parameters found in the [ComPorts] section of the CPF file. The [ComPorts] section applies to Windows systems only.

# COMn

Default settings for COM ports.

```
[ComPorts]      COMn=a;b
```

## Description

The parameters `COMn` apply to Windows platforms only. These parameters define the default settings for COM ports. This enables remote logins to Caché through locally connected or modem connected serial ports.

The number *n* refers to the physical com port number. If the [ComPorts] section contains enough entries, *n* may have multiple digits.

Each `COMn` entry provides two semicolon-separated values that define the default settings for COM port number *n*. From left to right:

- A set of COM port control parameters (data bits, parity, etc.) in byte-positional format. Byte position is one-based. From left to right:
  - Byte 1: Modem Control
    - '1' Use modem control (selected in the 1801X11 example below)
    - '0' Do not use modem control
    - ' ' No change to modem control (this is the default)
  - Byte 2: Data Bits
    - '5' 5 data bits
    - '6' 6 data bits
    - '7' 7 data bits
    - '8' 8 data bits (selected in the 1801X11 example below)
    - ' ' No change to bit size (this is the default)
  - Byte 3: Parity
    - '0' No parity (selected in the 1801X11 example below)
    - '1' Odd parity
    - '2' Even parity
    - '3' Mark parity
    - '4' Space parity
    - ' ' No change to the parity setting (this is the default)
  - Byte 4: Stop Bits
    - '1' 1 stop bit (selected in the 1801X11 example below)
    - '5' 1.5 stop bits
    - '2' 2 stop bits
    - ' ' No change to the stop bit setting (this is the default)



- Byte 5: Flow Control
    - 'X' Use Xon/Xoff flow control (selected in the 1801X11 example below)
    - 'C' Use CTS/RTS flow control
    - 'D' Use DSR/DTR flow control
    - ' ' No change to flow control (this is the default)
  - Byte 6: DTR Setting
    - '0' Disable DTR (set it off, keep it off)
    - '1' Enable DTR (set it on, keep it on) (selected in the 1801X11 example below)
    - ' ' No change to the DTR state (this is the default)
  - Byte 7: \$ZA Error Reporting
    - '0' Disable \$ZA error reporting
    - '1' Enable \$ZA error reporting (selected in the 1801X11 example below)
    - ' ' No change to \$ZA error reporting (this is the default)
- The baud rate. If not supplied, the default baud rate is 19200.

## Examples

The following example of a [ComPorts] section shows how spaces can be used as values within the COM port control parameter:

```
[ComPorts]
COM1=      ;19200
```

The first example uses all defaults for the COM port control parameters, by providing seven spaces before the semicolon separator. The meaning is: No change to modem control, no change to bit size, no change to parity setting, no change to stop bit setting, no change to Flow control, no change to DTR state, no change to \$ZA error reporting.

```
[ComPorts]
COM2=1801X11;19200
```

The second example provides a value of 1801X11 for the COM port control parameters. The meaning is: Use modem control, 8 data bits, no parity, 1 stop bit, use Xon/Xoff flow control, enable DTR, enable \$ZA error reporting.

## Range of Values

As described above.

## Management Portal

No equivalent.



# [config]

This topic describes the parameters found in the [config] section of the CPF file.

# LibPath

---

Sets the LD\_LIBRARY\_PATH environment variable. UNIX® systems only.

```
[config]    LibPath=path
```

## Description

UNIX® systems only. Sets the LD\_LIBRARY\_PATH environment variable (DYLD\_LIBRARY\_PATH on Mac) to paths to search for third-party shared libraries. If you modify this setting, you must restart the system to make it active.

## Examples

```
LibPath=/user/lib
```

## Range of Values

Valid path.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Advanced Memory**, in the [LibPath](#) row, select **Edit**. Enter a path.

---

# MaxServerConn

---

Maximum connections from ECP clients.

```
[config]    MaxServerConn=n
```

## Description

The value  $n$  is the maximum number of ECP clients that can access this system simultaneously. This is the maximum number of connections that this system may accept when acting as an ECP server.

## Examples

```
MaxServerConn=14
```

```
MaxServerConn=1
```

## Range of Values

0–254 clients. The default is 1 client.

## Management Portal

On the page **System Administration > Configuration > Connectivity > ECP Settings**, in the **This System as an ECP Data Server** column, edit **Maximum number of application servers**

# MaxServers

---

Maximum number of connections to ECP servers.

```
[config]    MaxServers=n
```

## Description

The value *n* is the maximum number of ECP servers that can be accessed from this system. This is the maximum number of connections that this system can establish when acting as an ECP client.

## Examples

```
MaxServers=2
```

## Range of Values

0–254 servers. The default is 2 servers.

## Management Portal

On the page **System Administration > Configuration > Connectivity > ECP Settings**, in the **This System as an ECP Application Server** column, edit **Maximum number of data servers**.

# Path

Add to default path environment variable. UNIX® systems only.

```
[config]    Path=directory1[:directory2]
```

## Description

On UNIX® systems, Caché processes started during Caché system startup are assigned a UNIX® PATH environment variable set by default to

```
PATH=/usr/bin:/bin:/usr/sbin:/usr/local/bin:/usr/local/sbin
```

Processes that use this path include Caché system daemons, processes started by the SYSTEM^%ZSTART routine, and processes which will be started by the Super Server (such as JDBC/ODBC servers).

Customer applications may require that the PATH environment variable for these processes have additional search directories appended to the default PATH provided by Caché. You can append directories to this path using the Path variable. If you modify this setting, you must restart the system to make it active.

**Note:** Terminal processes do not set their PATH this way; their PATH should be set by their login scripts.

## Examples

```
Path=/usr/customerapp/bin
```

This sets the PATH environment variable for system processes to

```
PATH=/usr/bin:/bin:/usr/sbin:/usr/local/bin:/usr/local/sbin:/usr/customerapp/bin
```

## Range of Values

*directories* must be one or more full paths of valid, existing directories, separated by colons.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Advanced Memory**, in the **Path** row, select **Edit**. Enter a directory name.

## bbsiz

---

Maximum memory per process.

```
[config]    bbsiz=n
```

### Description

The value  $n$  is the maximum memory allocation permitted for a process (in kilobytes). This amount of process private memory is used for symbol table allocation and various other memory requirements, including I/O devices. It is allocated in increasing extents as required by the application until the maximum is reached. When a process starts, its initial allocation is 128 kilobytes.

Once this memory is allocated to the process, it is generally not deallocated until the process exits. However, when a large amount of memory is used (for example greater than 32MB) and then freed, Cache attempts to release deallocated memory back to the operating system where possible.

### Examples

```
bbsiz=16384
```

### Range of Values

128–2,147,483,647 kilobytes. The default is 262,144 kilobytes.

### Management Portal

On the page **System Administration > Configuration > System Configuration > Memory and Startup**, enter a number of kilobytes in the [Maximum per Process Memory \(KB\)](#) row, select **Edit**.



# console

Console log file.

```
[config]    console=a,b
```

## Description

Two comma-separated values configure the console log file.

- *a* — This value is ignored and may be left blank, as shown in the first of the two examples below.
- *b* — Name of the file in which to log system console messages. You must specify the filename and extension only. The path is assumed to be the Caché system management directory. If no value is specified, Caché writes to a file called cconsole.log.

To further configure the console log file, see [MaxConsoleLogSize](#).

## Examples

```
console=,
```

```
console=OPA0: ,
```

## Range of Values

Either or both values may be omitted, as shown in the examples. If so, the respective defaults take effect.

## Management Portal

You can view the console log on the Management Portal at **System Operation > System Logs > Console Log**.

You can use the portal to set values as follows:

On the page **System Administration > Configuration > Additional Settings > Advanced Memory**:

- Ignore *a*.
- For *b*, in the [ConsoleFile](#) row, select **Edit**. Enter a file name.

# errlog

---

Maximum entries in the error log.

```
[config]    errlog=n
```

## Description

The value  $n$  is the maximum number of entries in the Caché system error log (see [Caché System Error Log](#) in the “Monitoring Caché Using the Management Portal” chapter of the *Caché Monitoring Guide* for more information). The log file expires old entries as this limit is reached.

## Examples

```
errlog=500
```

```
errlog=1000
```

## Range of Values

10–10000 entries. The default is 500 entries.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Advanced Memory**, in the [errlog](#) row, select **Edit**. Enter a number of entries.

# globals

Size of shared memory for global cache.

```
[config]    globals=0,b,c,d,e,f
```

## Description

Six comma-separated values specify how many megabytes of shared memory to allocate for different sizes of global cache. From left to right, each value is the number of megabytes for global caches organized into:

- no longer used, always 0
- 4-kilobyte blocks
- 8-kilobyte blocks
- 16-kilobyte blocks
- 32-kilobyte blocks
- 64-kilobyte blocks

So, for example, the following setting:

```
globals=0,0,16,0,0,0
```

Allocates 16 megabytes of buffers for 8-kilobyte database blocks.

## Examples

Allocate 8mb of 4kb blocks and 32 mb of 8kb blocks:

```
globals=0,8,32,0,0,0
```

Allocate 32 mb of 8kb blocks. If a need for a 4kb block exists, then the system uses an 8kb block to store it since no 4kb blocks are configured:

```
globals=0,0,32,0,0,0
```

Let the system allocate memory itself by setting all parameters to 0.

```
globals=0,0,0,0,0,0
```

## Range of Values

If all the values are 0, space is automatically allocated based on an estimate of the machine capability and shared memory resources. On a 64-bit system, there is a limit of 16 TB.

## Management Portal

On the page **System Administration > Configuration > System Configuration > Memory and Startup**, select the option to **Configure Memory Settings Manually**. Enter number of megabytes for the amount of memory to allocate for each database block size.

# gmheap

---

Size of the generic memory heap.

```
[config] gmheap=n
```

## Description

Number of kilobytes allocated for the generic memory heap for Caché. This is the allocation from which the Lock table, the NLS tables, and the PID table are also allocated. For Asian locales, add one extra megabyte (1024 KB) to the total gmheap value.

**Note:** To ensure optimal performance during a journal restore, InterSystems recommends increasing the generic memory heap size; see [Restore Journal Files](#) in the “Journaling” chapter of the *Cache Data Integrity Guide* for more information.

The generic memory heap (gmheap) is sometimes known as the shared memory heap (SMH).

## Examples

```
gmheap=23232
```

```
gmheap=42368
```

## Range of Values

16 MB to one terabyte. The default is 37568 KB (37 MB). On IBM AIX and HP-UX, the default is 53952 KB (54 MB).

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Advanced Memory**, in the [gmheap](#) row, select **Edit**. Enter a number of kilobytes.

# history

Command line recall.

```
[config]    history=a,b
```

## Description

Two comma-separated values configure command line recall features.

- *a* — Maximum number of entries held in the command line/read line recall buffer, subject to the space limitation in the second parameter. The range is 0–256 entries. The default is 32 entries.
- *b* — Total size (in bytes) of all input strings to be stored in the command line/read line buffer. The range is 0–8192 bytes. The default is 1024 bytes.

## Examples

```
history=32,1024
```

## Range of Values

As described above.

## Management Portal

If you want `history=a,b` you can use the portal to set these values as follows:

On the page **System Administration > Configuration > Additional Settings > Advanced Memory**:

- For *a*, in the [LineRecallEntries](#) row, select **Edit**. Enter a number of entries.
- For *b*, in the [LineRecallBuffer](#) row, select **Edit**. Enter a number of bytes.

## ijcbuf

---

Size of an InterJob Communication buffer.

```
[config]    ijcbuf=n
```

### Description

The value *n* is the number of bytes allocated for each InterJob Communication (IJC) buffer. For details, see the “[Interprocess Communication](#)” chapter in the *Caché I/O Device Guide*. Also see [ijcnum](#).

### Examples

```
ijcbuf=512
```

### Range of Values

512–8192 bytes. InterSystems recommends you use the default of 512 bytes.

### Management Portal

On the page **System Administration > Configuration > Additional Settings > Advanced Memory**, in the [ijcbuf](#) row, select **Edit**. Enter a number of bytes.

# ijcnum

Number of InterJob Communication devices.

```
[config]    ijcnum=n
```

## Description

The value  $n$  is the number of InterJob Communication (IJC) devices. Each device corresponds to one InterJob Communication buffer of the size defined by `ijcbuf`. For details, see the “[Interprocess Communication](#)” chapter in the *Caché I/O Device Guide*.

## Examples

```
ijcnum=16
```

## Range of Values

0–256 devices. InterSystems recommends you use the default of 16 devices.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Advanced Memory**, in the `ijcnum` row, select **Edit**. Enter a number of devices.

# jrnbufs

Size of memory for journal buffers.

```
[config]    jrnbufs=n
```

## Description

Amount of memory allocated for journal buffers. The default is 64 MB. Increasing this setting means increasing the amount of journal data that can be held in memory, which improves journaling performance but increases the amount of journal data that could potentially be lost in the event of a system failure. If you edit this setting, you must restart Caché to apply the change.

## Examples

```
jrnbufs=64
```

```
jrnbufs=256
```

## Range of Values

8 MB (for 8-bit instances) or 16 MB (for Unicode instances) up to 1024 MB.

## Management Portal

On the **Advanced Memory Settings** page (**System Administration** > **Configuration** > **Additional Settings** > **Advanced memory**), in the [jrnbufs](#) row, select **Edit** and enter a number of megabytes.



# locksiz

Maximum size of shared memory for locks.

```
[config]    locksiz=n
```

## Description

Size (in bytes) of memory allocated for locks. The system rounds up the value to the next multiple of 64 kilobytes. The default is 16777216 bytes. (On the IBM AIX and HP-UX platforms, the default is 33554432 bytes.) The range is 65536 bytes up to the [gmheap](#) size. If you need more room for the lock table, increase the [gmheap](#) parameter as well. If you edit this setting, changes will take effect immediately.

## Examples

```
locksiz=786432
```

```
locksiz=1179648
```

## Range of Values

65536 bytes up to the size of [gmheap](#). Since memory used to allocate locks is taken from [gmheap](#) (the Generic Memory Heap), you cannot use more memory for locks than exists in [gmheap](#). Memory from [gmheap](#) is also used for NLS tables and the PID table. If you increase the size of [locksiz](#), increase the size of [gmheap](#) as well.

On all platforms, except for AIX and HP-UX, the default amount of memory allocated for locks (the maximum size of the lock table) is 16777216 bytes. On the AIX and HP-UX platforms, the default amount of memory allocated for locks is two times the default on other platforms; that is  $2 * 16777216$  (33554432) bytes.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Advanced Memory**, in the [locksiz](#) row, select **Edit**. Enter a number of bytes.

# memlock

Enable/disable locking shared memory or the text segment into memory.

```
[config]    memlock=n
```

At startup, Caché allows a shared memory segment to use for control structure, global buffers, routine buffers, and generic memory heap. the memlock parameter offers detailed control over how that allocation occurs.

This flag controls how Caché allocates shares memory. This parameter is a set of bit flags that control the shared memory setting.

## Description

When the value of this parameter is set to 0 (the default), Caché attempts to allocated shared memory from large pages on platforms that support large pages (Windows, Linux, and AIX). If large pages are not available, or are insufficient for the configured shared memory size, then shared memory is allocated in standard (small) pages. If memory can still not be allocated, startup repeatedly tries with standard pages, reducing the global buffer, routine buffers, and generic memory heap (sometimes referred to as shared memory heap), until it succeeds. When shared memory is allocated in standard pages, no attempt is made to lock those pages into physical memory. Large pages are automatically locked in physical memory at the operating system level.

The following bit flags modify behavior as described below:

- 7 (LockedSharedMemory) indicates shared memory to be locked in physical memory. (Applies to all operating systems, except for Microsoft Windows, Solaris, or macOS.) Bits 1, 2, and 4 are always set together by convention, but they are interchangeable.

**Note:** For legacy reasons, this flag interacts with the use of large/huge pages in the following ways:

On Windows or AIX when this flag is on, it disables the use of large pages. This is true unless bit 64 (LargePagesRequired) is also turned on, which overrides it.

On Linux, when this flag is on and huge pages are set up at the operating system level, Caché will abort startup if the configured amount of memory cannot be allocated in huge pages.

It is NOT recommended to use this flag for those legacy purposes. Use this flag only when the intention is to get small pages locked in memory. When using large/huge pages, there is no need to use this flag because large/huge pages are automatically locked in physical memory. To control whether large/huge pages are disabled or required, use bits 32 (LargePagesDisabled) or 64 (LargePagesRequired)

- 8 (LockTextSegment) indicates the text segment (executable image) to be locked in physical memory (on some UNIX platforms.)
- 32 (LargePagesDisabled) indicates on platforms supporting large/huge pages, disable use of them for shared memory. Technically, this means that Caché adopts a neutral disposition towards page size, taking no action to request large pages. When this flag is off on platforms supporting large pages, Caché attempts to allocate memory in large pages and will back to standard pages if large pages cannot be allocated at the requested size..
- 64 (LargePagesRequired) indicates on platforms supporting large/huge pages (Windows, AIX, and Linux), require use of them for shared memory. Ignored on other platforms or if large pages are disabled by the LargePageDisabled flag. When LargePagesRequired is True and not ignored, if memory cannot be allocated in large/huge pages, startup is aborted rather than falling back to small pages. Caché retries with a small reduction in memory size, but not as much of a reduction as could occur in absence of this flag.
- 128 (BackoffDisabled) indicates on failure to allocate memory, do not retry with a reduced amount. If memory cannot be allocated at its configured size, startup is aborted.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Advanced Memory**, in each of the following rows, select **Edit** and choose a value:

- [BackoffDisabled](#) (true or false)
- [LargePagesDisabled](#) (true or false)
- [LargePagesRequired](#) (true or false)
- [LockSharedMemory](#) (true or false)
- [LockTextSegment](#) (true or false)

## netjob

---

Enable/disable remote job requests.

```
[config]    netjob=n
```

### Description

The value  $n$  may be 1 or 0:

- When 1 (true) incoming remote job requests via ECP will be honored on this server.
- When 0 (false) they will not.

### Examples

```
netjob=1
```

### Range of Values

0 or 1. The default is 1.

### Management Portal

On the page **System Administration > Configuration > Additional Settings > Advanced Memory**, in the [netjob](#) row, select **Edit**. Choose true or false.

---

# nlstab

---

Number of NLS collation tables.

```
[config]    nlstab=n
```

## Description

The value  $n$  is the number of NLS collation tables to allocate when Caché starts up. This parameter refers to loadable national collation tables and does not include built-in collations such as Caché Standard.

## Examples

```
nlstab=10
```

## Range of Values

0–64 tables. The default is 30 tables.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Advanced Memory**, in the [nlstab](#) row, select **Edit**. Enter a number of tables.

## overview

---

Caché platform and version summary.

```
[config]    overview=a~b
```

### Description

Two tilde-separated (~) values provide platform and version information. From left to right:

- The specific operating system on which Caché is running.
- The general type of the operating system.

### Examples

```
overview=Windows(Intel)~Windows
```

```
overview=Linux (Intel)~UNIX®
```

### Range of Values

As described above.

### Management Portal

No equivalent.

# pijdir

Pre-Image Journal directory.

```
[config]    pijdir=n
```

## Description

The value *n* is the name of the directory in which to store the Caché Pre-Image Journal (PIJ) file for clusters. This setting applies only to Caché instances that support clusters. This parameter is ignored on other platforms.

There is no default directory; you must enter a valid, existing directory name for this setting or the system you are configuring is unable to join the cluster correctly. Therefore, before starting the first Caché instance in a cluster, create a directory for the PIJ file on a disk that is accessible from all cluster members, and specify this directory as the PIJ Directory in the Caché configuration for each cluster member. If the Caché instance that you are configuring will not be a Caché cluster member, you may omit a value for the `pijdir` parameter.

## Examples

```
pijdir=$1DKA500:[CACHE.PIJ]
```

## Range of Values

If provided, *n* must be a valid, existing directory path.

## Management Portal

On the page **System Administration > Configuration > Database Backup > Cluster Settings**, in the [pijdir](#) row, select **Edit**. Enter a directory name.

# routines

Size of shared memory for caching routine buffers.

```
[config]    routines=n          or  routines=n1, n2, n3, n4, n5, n6
```

## Description

You can specify the amount of memory to allocate for caching routine buffers in two ways — either specifying a single value or specifying values for all six buffer sizes.

You can specify only a single value, *n*, for the total size, in megabytes, of shared memory that Caché allocates for routine buffers. Caché divides this memory into the various buffer sizes for optimum performance.

Or you can specify the amount of memory to allocate for each routine buffer size. If you use this format, you must specify all 6 values (some of them can be 0). If fewer than 6 are given, Caché reverts to the first format using *n=n1*.

- *n1* is the number of MB for 2 KB routine buffers
- *n2* is the number of MB for 4 KB routine buffers
- *n3* is the number of MB for 8 KB routine buffers
- *n4* is the number of MB for 16 KB routine buffers
- *n5* is the number of MB for 32 KB routine buffers
- *n6* is the number of MB for 64 KB routine buffers

If a non-zero memory size is specified for any buffer size, the minimum number of buffers allocated for that size is 430. Therefore, the actual memory allocated for buffers of a given size could be bigger than configured. Because the Caché system requires 64K buffers, at least the minimum number of 64K buffers is always allocated even if 0 is specified.

The total number of allocated buffers is limited to 65528. If the number of buffers allocated exceeds this number, it is reduced to the maximum. The total amount of shared memory space on a 64-bit system cannot exceed 1023 MB.

## Examples

```
routines=256
```

Allocate 256 MB for routine buffers, divided into various sizes by Caché's internal formula.

```
routines=0
```

To allocate a minimum number of buffers, enter a single value of 0.

## Range of Values

The default is 0, but Caché allocates a minimum amount of buffer space according to an internal formula. Minimum number of routine buffers for each size is 430 (if a smaller number is specified, the system adjusts up to 430). Total number of buffers cannot exceed 65528. Total amount of shared memory space on a 64-bit system cannot exceed 1023 MB.

For *routines=n*, Caché allocates a minimum of 35 MB. The maximum value depends on how much shared memory is allocated for global buffers and other uses. The total shared memory size cannot exceed 1023 MB. For *routines=n1,n2,n3,n4,n5,n6*, values other than 0 are subject to a minimum value of 300. The total number of buffers cannot exceed 65528.

## Management Portal

On the page **System Administration > Configuration > System Configuration > Memory and Startup**, select the option to **Configure Memory Settings Manually**. In the **Memory Allocated for Routine Cache (MB)** field, enter a number of megabytes. See [Memory and Startup Settings](#) in the “Configuring Caché” chapter of the *Caché System Administration Guide* for more information about allocating memory for routine buffers using the management portal.



# udevtabsiz

---

Maximum size of the device table.

```
[config]    udevtabsiz=n
```

## Description

The value *n* is the maximum size (in bytes) of the device table. This table maps device numbers (traditional logical unit numbers) to device names, so that ObjectScript code can open devices by number.

## Examples

```
udevtabsiz=24576
```

## Range of Values

0–65535 bytes. The default is 24576 bytes.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Advanced Memory**, in the [udevtabsiz](#) row, select **Edit**. Enter a number of bytes.

## wjdir

---

Write Image Journal files directory.

```
[config]    wjdir=n
```

### Description

Enter the name of a directory in which to store the write image journal file. The directory name may be up to 214 characters long. InterSystems recommends that the journal directory be located in a different partition from your databases. The default is the Caché installation manager's directory.

### Examples

```
wjdir=C:\InterSystems\Cache\mgr
```

### Range of Values

A valid, existing directory path.

### Management Portal

On the page **System Administration > Configuration > System Configuration > Journal Settings**, in the **Write image journal directory** row, select **Browse**. Select a directory name.

# zfheap

Size and configuration of the \$ZF heap.

```
[config]    zfheap=a,b
```

## Description

The \$ZF heap is used for output parameters in a callout function using \$ZF. For details, see the chapter “[Creating a Caché Callout Library](#)” in *Using the Caché Callout Gateway*.

Two comma-separated values configure the \$ZF heap. From left to right:

- *zfstrsz* is the number of characters Caché allows for a single string parameter on the \$ZF heap. How many bytes this actually requires depends on whether you are using 8-bit characters, Unicode (2-byte characters), or 4-byte characters on UNIX®. The range is 0–32767. A value of 0 tells Caché to determine an appropriate value for *zfstrsz* automatically. The default is 32767 characters.
- *zfheapsz* is the number of bytes Caché allocates for the \$ZF heap for all purposes. The \$ZF heap consists of the total number of bytes allocated in virtual memory for all \$ZF input and output parameters, including the space for strings allowed by the first value. The range is 0–270336 bytes. The default is 0. A value of 0 tells Caché to calculate an appropriate value for *zfheapsz* automatically, based on the value of *zfstrsz*.

The formula for calculating *zfheapsz* based on *zfstrsz* is as follows:

$$zfheapsz = (\text{BytesPerCharacter} * zfstrsz) + 2050$$

For example, if *zfstrsz* has the default value, 32767:

- If you are using Unicode, a single character is 2 bytes. The value for *zfheapsz* is then 67584 (or  $2 * 32767 + 2050$ ) bytes.
- On UNIX®, a single character is 4 bytes. The value for *zfheapsz* is then 133118 (or  $4 * 32767 + 2050$ ) bytes.

Intersystems recommends that this parameter be set to 0,0.

## Examples

```
zfheap=32767,67584
```

## Range of Values

As described above.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Advanced Memory**:

- For *zfstrsz*, in the [ZFString](#) row, select **Edit**. Enter a number of characters.
- For *zfheapsz*, in the [ZFSize](#) row, select **Edit**. Enter a number of bytes.



# [ConfigFile]

This topic describes the parameters found in the [ConfigFile] section of the CPF file.

## Version

---

Caché parameter file version.

```
[ConfigFile]    Version=n
```

### Description

The value *n* is the Caché parameter file version number. This value is independent of the Caché release number. It is updated automatically during Caché upgrade or installation. The number refers to the syntax and structure of the parameter file, and not to the changes that you might make to the parameter values within the file.

### Examples

```
Version=2009.1
```

### Range of Values

The value *n* is any string of numbers, usually in dot format, similar to the example above.

### Management Portal

No equivalent.

# [Databases]

This topic describes the parameters found in the [Databases] section of the CPF file.

# Database

Define Caché databases.

```
[Databases]      Name=a,b,c,d,e,f
```

## Description

Each entry has up to six comma-separated arguments that define a database. The only argument that is required is the first, the directory specification. If the other arguments are not specified, then the default is used. From left to right:

- Full directory specification for database file. Required.
- Remote system name (empty for local system). If the system is remote (non-empty name), subsequent fields are ignored. Default is the local system.
- Must mount at startup. 0 – Database is not mounted at startup (default). 1 – Database must be mounted at startup, or startup fails.
- Cluster mount mode – 0 – Private mount (default), 1 – cluster mount (valid only for clustered systems.)
- Mount at startup – 0 – Database is mounted when first accessed (default), 1 database attempts to be mounted at startup. If it fails, startup continues. Valid only for cluster databases; ignored for non-cluster databases. (To mount non-cluster databases at startup, use the third value, the Must mount at startup flag. Or mount the database explicitly in the ^%ZSTART routine using the SYS.Database class.
- Stream Location. Directory where the streams associated with this database go. The default is blank — "" (which Intersystems recommends.) This default location is in the subdirectory stream, underneath the database directory. For example, for a database located in c:\abc, the default stream location is c:\abc\stream.

By default even if they are not specified, the CACHESYS, CACHELIB, CACHETEMP, and CACHEAUDIT databases must be local to the system, private mounted, and mount required at startup.

## Examples

In the [Databases] section, each line is a database with arguments *Name=a,b,c,d,e,f*. Here is an example from Windows:

```
[Databases]
CACHESYS=c:\cache\mgr\
CACHELIB=c:\cache\mgr\cachelib\
CACHETEMP=c:\cache\mgr\cachetemp\
CACHEAUDIT=c:\cache\mgr\cacheaudit\
DOCBOOK=c:\cache\mgr\docbook\
SAMPLES=c:\cache\mgr\samples\
USER=c:\cache\mgr\user\
SALES=c:\sales\,SALESERVER
; Database is on system SALESERVER
BILLING=/usr/billing/,1
; Database is local and mount required
CLUSTER=dka300:[cluster],,1 ; Database is cluster mounted
```

Example of remote databases without mirroring:

```
PRDAUDIT=c:\cache\mgr\prdaudit\,PRD
PRDDCIFC=c:\cache\mgr\prddata\,PRD
PRDERR=c:\cache\mgr\prderr\,,1
```

Example of remote databases with mirroring. Note in this mirrored example the two remote databases, both mirrored and non-mirrored, are formatted differently than they would be in a non-mirrored environment.

```
PRDAUDIT=:mirror:PRDMIRROR:PRDAUDIT,PRD - Mirrored remote database
PRDDCIFC=:ds:PRDDCIFC,PRD - Non-mirrored remote database
PRDERR=c:\cache\mgr\prderr\,,1 - Local database
```



## Range of Values

As described above.

## Management Portal

On the page **System Administration > Configuration > System Configuration > Local Databases**, to add a new entry, select **Create New Database**. To edit an existing entry, select **Edit** in that entry's row.



# [Debug]

The Caché parameter file may include a [Debug] section. The [Debug] settings can be used for different kind of diagnostics. This topic describes one parameter that may be found in the [Debug] section of the Caché parameter file. All other [Debug] settings are InterSystems proprietary and this book does not document them.

# Dumpstyle

Style of core dump.

```
[Debug]    dumpstyle=n
```

## Description

When Caché performs a core dump, you can set the style of the dump using this option. Values and their meanings are listed in the following.

**Note:** On Unix, all dump styles generate a core file. The process cleans itself up as much as possible before exiting.

- 0
  - On Windows this is the *pid.dmp* file.
- 1
  - On Windows this is a Windows minidump file (type = `MiniDumpWithFullMemory`) named *cache<sub>pid</sub>.dmp*, which can be read by **WinDbg**, a Microsoft debugger. This is the most complete dump option, but it can create a huge dump file.
- 2
  - On Windows this is the old style exception processing where the process does minimal cleanup (**deqallresources** and **GRETRELEASE**) and then resignals the exception. The intention here is to catch the exception in a debugger and preserve as much information as possible to analyze.
  - On Unix this detaches shared memory before the Abort, so the core file does not contain the shared memory area.
- 3
  - On Windows this is a Windows minidump file (type = `MiniDumpWithDataSegs | MiniDumpWithPrivateReadWriteMemory | MiniDumpWithIndirectlyReferencedMemory`) named *cache<sub>pid</sub>.dmp*, which can be read by **WinDbg**. This creates a fairly large but useful dump file. This is the new default on Windows if `dumpstyle` is not specified.
- 4
  - On Windows this is a Windows minidump file (type = `MiniDumpNormal`) named *cache<sub>mpid</sub>.dmp*, which can be read by **WinDbg**. This creates a small dump file containing minimal information.

The active value can be changed with `$system.Config.ModifyDumpStyle(NewValue)`. This changes the value for all new Cache processes. It does not change the value in *cache.cpf*.

## Examples

```
dumpstyle=2
```

## Range of Values

0, 1, 2, 3, 4

## Management Portal

No equivalent.

# [Devices]

This topic describes the parameters found in the [Devices] section of the CPF file.

# Devices

Device types.

```
[Devices]      Name=a^b^c^d^e^f^g^h
```

## Description

Each entry provides eight up-arrow (^) values that define a device. The maximum length of all strings is 128 characters, except for the **Description** string, which can be up to 256 characters. From left to right:

- *PhysicalDevice*—The physical name used to refer to the device. The *PhysicalDevice* value specifies the *device* argument for this device's **OPEN** command. The name can contain up to 128 alphanumeric characters; it can contain space characters as well. For example, for a printer you could enter the following, where *MYNAME* is the computer name.

```
| PRN | \MYNAME\ISF-HP5SiMX7
```

Or:

```
| PRN | \MYNAME\Canon PIXMA
```

- *Type*—The type of device. Options: TRM=Terminal. SPL=Spooling device. MT=Magnetic Tape drive. BT=Cartridge tape drive. OTH=Any other device including printers and sequential files. The default depends on the device type.
- *SubType*—Used to refine the definition of your device subtypes. Subtypes specify terminal characteristics. They are used to create the appropriate **OPEN** command for the device. There should be subtype information for every terminal type.
- *Prompt*—1, 2, or NULL (a blank or empty field) to choose a prompt. Options: 1=Automatically uses this device if it is the current device. 2=Automatically uses this device with predefined Right Margin and Parameter settings. If this field is left blank, the user sees the device selection prompt with the default device defined.
- *OpenParameter*—A colon-separated string that provides the *parameters*, *timeout*, and *mnespace* arguments for this device's **OPEN** command. The syntax for the **OpenParameter** string is:

```
( parameters ) : timeout : " mnespace "
```

Inside the parentheses for *parameters*, individual items are colon-separated, as follows:

```
param1 : param2 : param3
```

Resulting in:

```
( param1 : param2 : param3 ) : timeout : " mnespace "
```

*timeout* and *mnespace* are optional, but if they are provided, the correct number of colons must separate them from previous entries in the **OpenParameter** string.

*parameters* must be contained within parentheses only if there is more than one parameter. If there are no parameters, or if there is only one parameter, the parentheses may be omitted from the string. Thus the following is a correct and complete **OpenParameter** string:

```
: timeout : " mnespace "
```

If provided, *mnespace* must be contained within double quotes, as shown.

For details about the **OPEN** command and its arguments, including a large variety of syntax examples, see the [Caché ObjectScript Reference](#).

- *AlternateDevice*—The device ID of another device. The value entered for *AlternateDevice* must be a defined mnemonic such as the *Name* supplied for another device.

Specifying an *AlternateDevice* value for the device allows users of the %IS utility to specify “A” to tell Caché to use the alternate device. %IS is a general device selection utility for character-based applications. For details about %IS

see the section “[Allowing Users to Specify a Device](#)” in the “I/O Devices and Commands” chapter of the *Caché I/O Device Guide*. The topic of most interest is “[%IS Mnemonics](#),” which describes the conventions for entering the “A” code for %IS.

- *Description*—A text description of where the device is located. This field is for your own reference to help you identify what machine you are configuring.
- *Alias*—An alternate device ID (number) for this device. All aliases must be unique. You can use this value as the *device* argument in an [OPEN](#) command.

## Examples

In the [Devices] section, each entry *Name=a^b^c^d^e^f^g^h^i* appears all on one line:

```
[Devices]
0=0^TRM^C-Terminal^^^^Principal device^
2=2^SPL^PK-DEC^^^^Spool LA120^
47=47^MT^M/UX^^("auv":0:2048)^^Magnetic tape^
48=48^MT^M/UX^^("avl":0:2048)^^Magnetic tape^
57=57^BT^M/UX^^("auv":0:2048)^^Magnetic tape^
58=58^BT^M/UX^^("avl":0:2048)^^Magnetic tape^
SPOOL=2^SPL^PK-DEC^^^^Spool LA120^
TERM=0^TRM^C-Terminal^^^^Windows Console^
|PRN|=|PRN|^OTH^P-DEC^^"W"^^Windows Printer^
|TNT|=0^TRM^C-VT220^^^^Principal device^
|TRM|=0^TRM^C-Terminal^^^^Windows Console^
```

## Range of Values

As described above. The maximum length of strings is 128 characters, except for **Description**, which can be 256 characters.

## Management Portal

On the page **System Administration > Configuration > Device Settings > Devices** is a list of existing devices. Select **Create New Device**, **Edit**, or **Delete** to modify the list.





# [DeviceSubTypes]

This topic describes the parameters found in the [DeviceSubTypes] section of the Caché parameter file.

# DeviceSubTypes

Device subtypes.

```
[DeviceSubTypes]      Name=n=a^b^c^d^e^f^g^h^i^j
```

## Description

Each entry provides nine up-arrow (^) values that define a device subtype. From left to right:

- *RightMargin*—The number that represents the location of the right margin. Device output wraps at that number of characters.
- *FormFeed*—The ASCII code that represents a form feed on the selected device in the form #,\$C(code1,code2...). This setting is used by the Caché CHUI utilities.
- *ScreenLength*—The number of lines that comprise one screen or page for the device.
- *Backspace*—The ASCII code that represents the backspace character on the selected device in the form \$C(code1). This setting is used by the Caché CHUI utilities.
- *CursorControl*—The ASCII code that represents the cursor on the selected device in the form \$C(code1).
- *EraseEOL*—The ASCII code that represents erasing the end of line characters on this device in the form \$C(code1,code2).
- *EraseEOF*—The ASCII code that represents erasing the end of file character on the selected device in the form \$C(code1,code2...).
- *ZU22FormFeed*—The ASCII code that represents a form feed on the selected device in the form \$C(code1,code2). This setting is used by Caché for Terminal output.
- *ZU22Backspace*—The ASCII code that represents a backspace on the selected device in the form \$C(code1). This setting is used by Caché for Terminal output.

## Examples

The following is a sample [DeviceSubTypes] section. This example wraps long lines to fit them onto the viewing page. In the .cpf file itself, each entry appears all on one line:

```
[DeviceSubTypes]
C-ANSI=80^#,$C(27,91,72,27,91,74)^25^$C(8)^W $C(27,91)_(DY+1)_" ;"_(DX+1)_"H"
S $X=DX,$Y=DY^$C(27,91,74)^$C(27,91,75)^$C(27,91,72,27,91,74)^$C(8,32,8)
C-Terminal=80^#,$C(27,91,72,27,91,74)^24^$C(8)^W $C(27,91)_(DY+1)_" ;"_(DX+1)_"H" S $X=DX,$Y=DY^$C(27,91,74)^$C(27,91,75)^$C(27,91,72,27,91,74)^$C(8,32,8)
C-TV925=80^#,$C(27,44)^24^$C(8)^W $C(27,61,DY+32,DX+32) S $X=DX,$Y=DY^^^$C(27,44)^$C(8,32,8)
C-VT100=80^#,$C(27,91,72,27,91,74)^24^$C(8)^W $C(27,91)_(DY+1)_" ;"_(DX+1)_"H" S $X=DX,$Y=DY^$C(27,91,74)^$C(27,91,75)^^
C-VT101W=132^#,$C(27,91,72,27,91,74)^14^$C(8)^W $C(27,91)_(DY+1)_" ;"_(DX+1)_"H" S $X=DX,$Y=DY^$C(27,91,74)^$C(27,91,75)^^
C-VT132=132^#,$C(27,91,72,27,91,74)^24^$C(8)^W $C(27,91)_(DY+1)_" ;"_(DX+1)_"H" S $X=DX,$Y=DY^$C(27,91,74)^$C(27,91,75)^^
C-VT220=80^#,$C(27,91,72,27,91,74)^24^$C(8)^W $C(27,91)_(DY+1)_" ;"_(DX+1)_"H" S $X=DX,$Y=DY^$C(27,91,74)^$C(27,91,75)^$C(27,91,72,27,91,74)^$C(8,32,8)
C-VT240=80^#,$C(27,91,72,27,91,74)^24^$C(8)^W $C(27,91)_(DY+1)_" ;"_(DX+1)_"H" S $X=DX,$Y=DY^$C(27,91,74)^$C(27,91,75)^$C(27,91,72,27,91,74)^$C(8,32,8)
C-VT52=80^#,$C(27,72)^24^$C(8)^W $C(27,89,DY+32,DX+32) S $X=DX,$Y=DY^^^
M/UX=255^#^66^$C(8)^^^^^
MAIL=132^#^11^$C(8)^^^^^
P-DEC=132^#^66^$C(8)^^^^^
PK-DEC=150^#^66^$C(8)^^^^^
PK-QUME=150^#^66^$C(8)^^^^^
```

## Range of Values

All default values depend on the device type.

## Management Portal

On the page **System Administration > Configuration > Device Settings > Device Subtypes** is a list of existing subtypes. Select **Create New Sub Type**, **Edit**, or **Delete** to modify the list.



# [ECP]

This topic describes the parameters found in the [ECP] section of the CPF file.

# ClientReconnectDuration

How long to attempt ECP reconnection before giving up.

```
[ECP] ClientReconnectDuration=n
```

## Description

The value  $n$  is the number of seconds an Application Server (ECP client) should keep trying to reestablish a connection before giving up or declaring the connection failed. The Application Server (ECP client) continues reconnection attempts at intervals scheduled by the [ClientReconnectInterval](#) until the full `ClientReconnectDuration` expires.

## Examples

```
ClientReconnectDuration=1200
```

## Range of Values

10–65535 seconds. The default is 1200 seconds (20 minutes).

## Management Portal

On the page **System Administration > Configuration > Connectivity > ECP Settings**, in the **This System as an ECP Application Server** column, edit **Time to wait for recovery**

# ClientReconnectInterval

Number of seconds to wait between ECP reconnection attempts.

```
[ECP] ClientReconnectInterval=n
```

## Description

The value  $n$  is the number of seconds to wait between each reconnection attempt when a Data Server (ECP server) is not available. The Application Server (ECP client) continues reconnection attempts at intervals scheduled by `ClientReconnectInterval` until the full `ClientReconnectDuration` expires.

## Examples

```
ClientReconnectInterval=5
```

## Range of Values

1–60 seconds. The default is 5 seconds.

## Management Portal

On the page **System Administration > Configuration > Connectivity > ECP Settings**, in the **This System as an ECP Application Server** column, edit **Time between reconnections**

## ServerTroubleDuration

---

Number of seconds an ECP connection stays in troubled state.

```
[ECP]    ServerTroubleDuration=n
```

### Description

The value  $n$  is the number of seconds an ECP connection stays in a troubled state. Once this period of time has elapsed, the Data Server (ECP server) declares the connection dead and presumes that recovery is not possible.

### Examples

```
ServerTroubleDuration=60
```

### Range of Values

20–65535 seconds. The default is 60 seconds.

### Management Portal

On the page **System Administration > Configuration > Connectivity > ECP Settings**, in the **This System as an ECP Data Server** column, edit **Time interval for Troubled state**



# [ECPServers]

This topic describes the server entries found in the [ECPServers] section of the CPF file.

# ECPServers

---

ECP servers.

```
[ECPServers]      Name=Address,Port, MirrorConnection
```

## Description

Entries defines the address and port of the ECP server to connect to and the way in which connections should be redirected if the ECP server is a mirror primary. (See [Configuring ECP Connections to a Mirror](#) in the “Mirroring” chapter if the *Caché High Availability Guide* for important information about configuring a mirror primary as a data server.)

- *Address* – Address of the ECP server to connect to.
- *Port* – Port # of the ECP server to connect to.
- *MirrorConnection* – Behavior when connecting to a mirror primary. Default is 0 (or blank), indicating that the data server is not a mirror member. A value of 1 indicates the ECP server is a mirror failover member, and the mirror connection redirects to whichever member is primary in the event of a failover. A value of -1 indicates the ECP server is either a failover member or DR async, and the mirror connection is restricted to that specific ECP server; if the ECP server becomes the backup member, it does not accept the connection until it becomes primary.

## Example

```
SALESDATA=168.27.14.9,1972,1
```

## Management Portal

On the page **System Administration > Configuration > Connectivity > ECP Settings** is a list of ECP data servers. Select **Add Remote Data Server** to add a new ECP data server.

# [IO]

This topic describes the parameters found in the [IO] section of the Caché parameter file.

# MagTape

Default mnemonic for WRITE commands to magnetic tape.

```
[ IO ]      MagTape=n
```

## Description

When an OPEN or USE command includes no mnemonic space argument, Caché uses the default mnemonic for that device type. The `Mnemonic_MT` setting specifies the default mnemonic for magnetic tape.

The value *n* is a Caché routine name. `^%XMAG` is the default for magnetic tape. You can use this default, or reset it by changing the value of as needed. See the “[Controlling Devices with Mnemonic Spaces](#)” chapter in the *Caché I/O Device Guide*.

## Examples

```
MagTape=^%XMAG
```

## Range of Values

As described above.

## Management Portal

On the page **System Administration > Configuration > Device Settings > IO Settings**, in the in the [MagTape](#) row, enter a Caché routine name.

# Other

Default mnemonic for WRITE commands to device types other than magnetic tape, terminal, or sequential file.

```
[ IO]      Other=n
```

## Description

When an OPEN or USE command includes no mnemonic space argument, Caché uses the default mnemonic for that device type. The `Others` setting specifies the default mnemonic for device types other than magnetic tape, terminal, or sequential file.

The value *n* is a Caché routine name. `^%X364` is the default when the device is not a magnetic tape drive, terminal, or sequential file. You can use this default, or reset it by changing the value of as needed. See the “[Controlling Devices with Mnemonic Spaces](#)” chapter in the *Caché I/O Device Guide*.

## Examples

```
Other=^%X364
```

## Range of Values

As described above.

## Management Portal

On the page **System Administration > Configuration > Device Settings > IO Settings**, in the [Other](#) row, enter a Caché routine name.

---

# File

---

Default mnemonic for WRITE commands to a sequential file.

```
[IO]      File=n
```

## Description

When an OPEN or USE command includes no mnemonic space argument, Caché uses the default mnemonic for that device type. The `File` setting specifies the default mnemonic for sequential files.

The value *n* is a Caché routine name. `^%X364` is the default for sequential files. You can use this default, or reset it by changing the value of as needed. See the “[Controlling Devices with Mnemonic Spaces](#)” chapter in the *Caché I/O Device Guide*.

## Examples

```
File=^%X364
```

## Range of Values

As described above.

## Management Portal

On the page **System Administration > Configuration > Device Settings > IO Settings**, in the [File](#) row, enter a Caché routine name.

---

# Mnemonic\_TTY

---

Default mnemonic for WRITE commands to a terminal device.

```
[ IO]      Mnemonic_TTY=n
```

## Description

When an OPEN or USE command includes no mnemonic space argument, Caché uses the default mnemonic for that device type. The Mnemonic\_TTY setting specifies the default mnemonic for terminal devices.

The value *n* is a Caché routine name. ^%X364 is the default for terminal devices. You can use this default, or reset it by changing the value of as needed. See the “[Controlling Devices with Mnemonic Spaces](#)” chapter in the *Caché I/O Device Guide*.

## Examples

```
Mnemonic_TTY=^%X364
```

## Range of Values

As described above.

## Management Portal

On the page **System Administration > Configuration > Device Settings > IO Settings**, in the [Terminal](#) row, enter a Caché routine name.





# [Journal]

This topic describes the parameters found in the [Journal] section of the Caché parameter file.

# AlternateDirectory

---

Alternate location of the journal file.

```
[Journal]    AlternateDirectory=n
```

## Description

The value *n* is the name of an alternate (secondary) directory in which to store journal files.

This alternate directory is used if the primary journal directory specified by [CurrentDirectory](#) is unavailable; for example, if the disk partition for the primary journal directory is full, offline, or has some other problem. For these reasons, InterSystems recommends that the alternate journal directory be located on a different disk than the current journal directory.

When installed, the initial value of this field is "". After Caché starts for the first time, the actual journal directory is filled in here, such as c:\intersystems\cache\mgr\journal.

## Examples

```
AlternateDirectory=
```

## Range of Values

The name of an existing directory.

## Management Portal

On the page **System Administration > Configuration > System Configuration > Journal Settings**, in the **Secondary journal directory** row, select **Browse**. Select the name of an existing directory.

# BackupsBeforePurge

Number of backups before Caché purges finished journal files.

[Compatibility]      BackupsBeforePurge=*n*

## Description

The value defines when Caché purges a finished journal file (that is, a journal file that is no longer in progress). The value *n* is a number of successful Caché system backups that must take place before the corresponding journal files can be purged. The range is 0–10 backups. The default is 2 backups.

BackupsBeforePurge relates to DaysBeforePurge. If both are greater than 0, files are purged after *n* days or *n* successful backups, whichever indicates the shorter time period. If BackupsBeforePurge is 0, purging is done solely based on DaysBeforePurge; if DaysBeforePurge is 0, then purging is done solely based on BackupsBeforePurge. If both are 0, the automatic purging of journal files (and journal history) is disabled and journal files are not purged.

No journal file containing currently open transactions is purged, even if it meets the above criteria.

For details about journal files, see the “[Journaling](#)” chapter in the *Caché Data Integrity Guide*.

## Examples

BackupsBeforePurge=3

## Range of Values

0–10. The default is 2.

## Management Portal

On the page **System Administration > Configuration > System Configuration > Journal Settings**, in the **When to purge journal files** category, choose one of the following. Setting one of these values to a non-zero value in the portal causes the other to be set to 0:

- To purge journal files based on a number of days, choose **After this many days** and enter a number of days.
- To purge journal files based on a number of backups, choose **After this many successive successful backups** and enter a number of backups.

## CurrentDirectory

---

Primary location of the journal file.

```
[Journal]    CurrentDirectory=n
```

### Description

The value *n* is the name of a directory in which to store the journal files (the primary directory). When installed, the initial value of this field is “”. After Caché starts for the first time, the actual journal directory is filled in here, such as `c:\intersystems\cache\mgr\journal`.

### Examples

```
CurrentDirectory=
```

### Range of Values

The name of an existing directory. The default is the subdirectory `journal` under the system manager directory.

### Management Portal

On the page **System Administration > Configuration > System Configuration > Journal Settings**, in the **Primary journal directory** row, select **Browse**. Select the name of an existing directory.

# DaysBeforePurge

Number of days before Caché purges finished journal files.

[Compatibility]      DaysBeforePurge=*n*

## Description

The value defines when Caché purges a finished journal file (that is, a journal file that is no longer in progress). The value *n* is a number of days that elapse before the corresponding journal files can be purged. The range is 0–100 days. The default is 2 days.

BackupsBeforePurge relates to DaysBeforePurge. If both are greater than 0, files are purged after *n* days or *n* successful backups, whichever indicates the shorter time period. If BackupsBeforePurge is 0, purging is done solely based on DaysBeforePurge; if DaysBeforePurge is 0, then purging is done solely based on BackupsBeforePurge. If both are 0, the automatic purging of journal files (and journal history) is disabled and journal files are not purged.

No journal file containing currently open transactions is purged, even if it meets the above criteria.

For details about journal files, see the “[Journaling](#)” chapter in the *Caché Data Integrity Guide*.

## Examples

DaysBeforePurge=3

## Range of Values

0–100. The default is 2.

## Management Portal

On the page **System Administration > Configuration > System Configuration > Journal Settings**, in the **When to purge journal files** category, choose one of the following. Setting one of these values to a non-zero value in the portal causes the other to be set to 0:

- To purge journal files based on a number of days, choose **After this many days** and enter a number of days.
- To purge journal files based on a number of backups, choose **After this many successive successful backups** and enter a number of backups.

# FileSizeLimit

---

Maximum size of a journal file.

```
[Journal]      FileSizeLimit=n
```

## Description

The value  $n$  is the maximum size of the journal file, in megabytes. When a journal file grows to this size it will be closed and a new journal file will be created. For more information about journal file rollover, see the [Caché Data Integrity Guide](#).

## Examples

```
FileSizeLimit=1024
```

## Range of Values

0–4079. A number of megabytes. A value of 0 (the default) means use the system default (currently 1024).

## Management Portal

On the page **System Administration > Configuration > System Configuration > Journal Settings**, in the **Start new journal file every... (MB)** row, enter a number of megabytes.

# FreezeOnError

Enable/disable suspending Caché when an error occurs in writing to the journal.

```
[Compatibility]      FreezeOnError=n
```

## Description

The value *n* may be 1 or 0:

- If `FreezeOnError=1` (true) then, on a journal file I/O error, the Journal daemon freezes journaling immediately. The Journal daemon unfreezes journaling after it succeeds with the failed I/O operation. As soon as the error occurs all global activities that are normally journaled are blocked, which causes other jobs to hang; typically until you resolve the journaling problem. While Caché is hanging, you can take corrective measures, such as freeing up space on a full disk or switching the journal to a new working disk. This prevents the loss of journal data at the expense of system availability.
- If `FreezeOnError=0` (false), then Caché does not freeze on a journal file I/O error. Caché disables journaling if it is not able to recover in a timely manner from the error. This prevents the system from hanging. Journaling is first retried, and then disabled, while Caché continues running. If journaling is disabled, you want to backup your databases as soon as possible. Running without journaling is a calculated risk, as it means the activity that occurs during this period cannot be restored. If Journaling is disabled, you **MUST** restart it. Ways to restart it include running the [^JRN-START](#) routine or selecting option 1, Begin Journaling, from the [^JOURNAL](#) routine menu. See the “[Start Journaling Using ^JRNSTART](#)” section for details.

If `FreezeOnError=0` and you are shadowing to another system and a journal error occurs, then the shadow may fall out of sync with the master.

For details, see the section “[Journal I/O Errors](#)” in the “Journaling” chapter of the *Caché Data Integrity Guide*.

## Examples

```
FreezeOnError=0
```

## Range of Values

0 or 1. The default is 0.

## Management Portal

On the page **System Administration > Configuration > System Configuration > Journal Settings**, in the **FreezeOnError** box, choose Yes (true) or No (false).

## JournalFilePrefix

---

Specify a journal file name prefix.

```
[Journal]    JournalFilePrefix=n
```

### Description

Enter an alphanumeric string to distinguish the journal file name.

### Examples

```
JournalFilePrefix=Oct
```

### Range of Values

An alphanumeric string. The default is blank.

### Management Portal

On the page **System Administration > Configuration > System Configuration > Journal Settings**, at the **Journal File Prefix** row, enter a prefix.



---

# JournalcspSession

---

Enable/disable journaling of CSP sessions.

```
[Journal]      JournalcspSession=n
```

## Description

The value *n* may be 1 or 0:

- When 0 (false) the %cspSession global is mapped to CACHETEMP and not journaled. Caché kills the %cspSession global on system restart or upgrade to a new Caché software version, so that any record of ongoing [Caché Server Page \(CSP\)](#) sessions is removed.
- Changing this setting to 1 (true) prevents this from happening. Set this to 1 so that the CSP session global is journaled if you want the CSP session global to be replicated onto another machine for failover or if you want a CSP session to survive a Cache restart.

## Examples

```
JournalcspSession=0
```

## Range of Values

0 or 1. The default is 0.

## Management Portal

On the page **System Administration > Configuration > System Configuration > Journal Settings**, for the **Journal CSP Session** setting, choose Yes or No.



# [LicenseServers]

This topic describes LicenseServer, the parameter found in the [LicenseServer] section of the CPF file. There can be multiple license servers defined.

# LicenseServers

License servers.

```
[LicenseServers]      Name=ipaddress, port
```

## Description

You can define one or more license servers in the [LicenseServers] section. Each license server is defined as follows:

*Name=IpAddress,Port*

The section looks like this:

```
[LicenseServers]
LOCAL=127.0.0.1,4001 - Default
LOCAL1=127.0.0.1,4020
```

The license software selects one of the License Servers to be the active server. The other servers are available to take over should the active server fail. A monitor process sends update messages to the license server, which coordinates license allocation when more than one Caché instance shares a license. All instances on one or more hosts that share a multiserver license key are required to use the same set of license servers.

The format for the value *ip address, port* is:

```
addr1,port1[:addr2,port2[...:addrx,portx]]
```

Where a semicolon separates each address/port pair, and within the pair, a comma separates the address and port. (Square brackets [] do not appear in the syntax; they enclose optional portions of the string in the syntax example.)

The values in each comma-separated pair are, from left to right:

- *IpAddress*— IP address of the License Server.
- *Port*—Port number. The UDP port number used by the License Server. The port numbers used at different IP addresses do not need to be different. However, the License Server port number used at each IP address must be different from any UDP port number used at that IP address.

## Examples

```
LicenseServer=127.0.0.1,4001
```

```
LicenseServer=127.0.0.1,4321
```

## Range of Values

Valid IP addresses and ports.

## Management Portal

On the page **System Administration > Licensing > License Servers**, select **Add**, **Edit**, or **Delete**. If you select **Add** or **Edit** you can enter data in the **Name/IP Address** and **Port** fields and select **Save**.

# [MagTapes]

This topic describes the parameters found in the [MagTapes] section of the CPF file.

# MagTapes

Magnetic tape drives.

```
[MagTapes]    Name=a
```

## Description

Each entry provides a value that defines a magtape drive.

- *Name*—Name of this magtape drive.
- *a*—The pathname to the device in the system. This string may be up to 128 characters long.

## Examples

The following are some sample [MagTapes] sections on various platforms:

- On Windows:

```
[MagTapes]
47~\\.\TAPE0
48~\\.\TAPE1
57~\\.\TAPE0
58~\\.\TAPE1
```

- On UNIX® or Linux:

```
[MagTapes]
47=/dev/st0
48=/dev/st1
49=/dev/st01
50=/dev/st11
51=/dev/st0a
52=/dev/st1a
```

## Range of Values

As described above.

## Management Portal

On the page **System Administration > Configuration > Device Settings > Magnetic Tape Devices** is a list of existing magnetic tape devices. Select **Create New Mag Tape**, **Edit**, or **Delete** to modify the list.

# [Map]

This topic describes the parameters found in the [Map] section of the CPF file. It also describes parameters that may be present in sections called [Map.*w*], where *w* is the name of a Caché namespace.

# Global

Map globals to namespaces.

```
[Map.w]      Global_name(s)=Database,Collation,LockLocation
```

## Description

Caché adds entries to the CPF file automatically when you define mappings for globals using the Management Portal (see the instructions below). The entries appear in a section called [Map.w] where *w* is the namespace to which the global is being mapped.

Each entry provides comma-separated values that map a global to a namespace. From left to right:

- Database location of the global.
- Collation of the global (Default=5, Caché standard collation).
- Lock database location for the global (Default=Database location). (It is recommended that the lock database be the same as the Database location.).

The collation setting is ignored if the global is not mapped at the subscript level. If the actual collation of the global does not match the defined collation in the namespace definition (for subscript level mapped globals), a <COLLATEMISMATCH> error is generated when it is referenced. This is because Caché requires the global to have the same collation across all the databases it lives in, regardless of the default collation for the particular database.

Only the first field is required. If the other fields are not specified, they are set to the system default.

Note that if a subscript mapping is specified, a higher level mapping of the global itself must also exist. So if you want to create a mapping ^X(9), then a mapping for ^X must also exist.

Subscripts (*s*) can be specified as part of the global name. *s* can be any of the following:

```
( 1 )
( "A" )
( 1 ) : ( 5 ) — from 1 up to, but not including 5
( "A" ) : ( "Z" ) — from A up to, but not including, Z
( BEGIN ) : ( "X" ) — from the beginning up to, but not including, X
( "Y" ) : ( END ) — from Y up to the end
```

## Examples

Map the global ^SALES to the SALES Database:

```
Global_SALES=SALES
```

Subscript map the global ^SALES("MA") to the SALESMA database:

```
Global_SALES( "MA" )=SALESMA
```

Map the global ^ACCOUNT to the database ACCOUNTS:

```
Global_ACCOUNT=ACCOUNTS
```

Subscript map the global ^ACCOUNT(1) up to but not including ACCOUNT(5) to database ACCOUNTS1TO4:

```
Global_ACCOUNT( 1 ) : ( 5 )=ACCOUNTS1TO4
```



Map all globals starting with ABC to database ABC:

```
Global_ABC*=ABC
```

## Range of Values

Valid global, namespace, and database names.

## Management Portal

On the page **System Administration > Configuration > System Configuration > Namespaces**, select **Global Mappings** for the namespace you want to explore. To add a new entry, select **New Global Mapping**. To edit an existing entry, select **Edit** in that entry's row.

# Package

---

Map a package to a specific namespace.

```
[Map.w]      Package_Name=Database
```

## Description

Caché adds entries to the CPF file automatically when you map a package to a specific namespace  $w$  using the Management Portal (see the instructions below). The entries appear in a section called [Map. $w$ ] where  $w$  is the namespace to which the package is being mapped.

Each entry provides a package database location.

Classes in the specified package become available in the  $w$  namespace.

## Examples

Map package TEST to the USER database.

```
Package_Test=USER
```

## Range of Values

A valid package and database name.

## Management Portal

On the page **System Administration > Configuration > System Configuration > Namespaces**, select **Package Mappings** for the namespace you want to explore. To add a new entry, select **New Package Mapping**. To edit an existing entry, select **Edit** in that entry's row.

---

# Routine

---

Map routines to namespaces.

```
[Map.w] Routine_Name Type=Database
```

## Description

Caché adds entries to the CPF file automatically as you define routine mappings using the Management Portal (see the instructions below). The entries appear in a section called [Map.w] where *w* is the namespace name.

Each entry provides one value that maps a routine to a namespace.

Type is normally not specified. Type only needs to be specified if you want to map part of a routine to another database. Valid values for Type are: MAC, INT, INC, or OBJ.

## Examples

Map routine SALE to the SALES database:

```
Routine_SALE=SALES
```

Map all routines starting with ACC to the ACCOUNTS database:

```
Routine_ACC*=ACCOUNTS
```

Map the object code for routine TEST to the TEST database:

```
Routine_TEST_OBJ=TEST
```

## Range of Values

A valid routine, optional type, and database name.

## Management Portal

On the page **System Administration > Configuration > System Configuration > Namespaces**, select **Routine Mappings** for the namespace you want to explore. To add a new entry, select **New Routine Mapping**. To edit an existing entry, select **Edit** in that entry's row.



# [MapMirrors]

This topic describes the [MapMirrors] section of the CPF file.

# MapMirrors

Describes members of each mirror the instance is a member of.

```
[MapMirrors]
Name=,AgentAddress,AgentPort,-,SuperServerAddress,GUID,InstanceDirectory,MemberType,MirrorPrivate,MirrorSSPort,-,EncryptCommunication,
```

## Description

Each entry consists of comma-separated values that define a mirror member. From left to right:

- *Name* – Required. Unique name identifying this member within the mirror. An uppercase alphanumeric string with a maximum length of 15 characters, cannot contain spaces or tabs, commas (,), semicolons (;), or equal signs (=), and is converted to uppercase before saving.

This is the name of this instance in the mirror. Mirror system names must be unique (enforced by the EMS) across all of the mirrors which the EMS manages as an instance has a single mirror system name which may appear in multiple mirror sets (that is, an async member may connect to multiple mirrors). For the most part the *GUID* is used to identify a mirror member, the *Name* is used for display purposes. The name cannot contain a colon (:).

- *AgentAddress* – The network address (IP address preferred to avoid DNS issues) that mirror members which connect to the primary should use to contact the Agent on this failover member. This is omitted on async members as the agent is not used for mirroring on those systems. This is required on failover members. The agent can transfer journal data so a private address may be desirable here to avoid network congestion.
- *AgentPort* – Port # which the agent on this system is configured to listen on.  
`##class(SYS.Agent).GetApplicationPort()` returns the current value if the local agent is active.
- For internal use.
- *SuperServerAddress* – The network address used to connect to the primary by external mirror-aware systems (currently only ECP application servers, although in the future this may extend to other connections). Other mirror members may connect to a member's superserver address for control and monitoring purposes. When a member is primary, an async member attempts to establish its data channel (over which it receives journal data) using this address if the primary's mirror private address (*MirrorPrivate*) is not accessible.
- *GUID* – Required. An internal GUID, unique to this mirror. Uniquely identifies this node in the mirror. Apart from identifying the nodes, primarily used to identify the instance that owns a particular copy of a mirrored database.
- *InstanceDirectory* – The installation directory of the instance (the parent of the mgr directory). Used primarily on failover members to identify the instance to the agent.
- *MemberType* – Numeric value indicating the type of mirror member. One of:
  - 0 - Failover member
  - 2 - Async member
- *MirrorPrivate* – When this instance is primary, other mirror members use this address to establish the mirror data channel, over which they receive journal data from the primary. Async members fall back to the primary's superserver address (*SuperServerAddress*) if they cannot reach it at the mirror private address.
- *MirrorSSPort* – SuperServer port for this instance. Used in conjunction with both the *MirrorAddress* and the *ECPAddress* by clients establishing connections to this instance.
- For internal use.
- For internal use.
- For internal use.

## Examples

Each entry is on one line:

```
[MapMirrors.MIMI]  
MIMI_A=mirrorhostA,2188,,mirrorhostA,C7BA9224-3851-47D4-83BD,c:\intersystems\20142302july10a\,0,mirrorhostA,56776,,0,  
MIMI_B=mirrorhostB,2188,,mirrorhostB,D14611B3-E0F5-4708-A111,c:\intersystems\20142302july10b\,0,mirrorhostB,56777,,0,  
MIMI_D=mirrorhostD,2188,,mirrorhostD,06E1D307-59D9-4500-AA3B,c:\intersystems\20142302jul10d\,2,mirrorhostD,56779,,0,
```

## Management Portal

On the pages **System Administration > Configuration > Mirror Settings > Create a Mirror > Join as Failover** (or **Join as Async**), enter the requested information.





# [MapShadows.Name]

This topic describes the [MapShadows.Name] sections of the CPF file.

# MapShadows.Name

---

Displays shadow directory mappings.

```
[MapShadows.Name] SourceDirectory1=DestinationDirectory1
```

## Description

Each entry describes a shadow server by its name and its source and destination directories.

For more information about defining and managing shadow servers, see the chapter [Shadowing](#) in the *Caché Data Integrity Guide*.

## Example

The following fragment defines a shadow server.

```
[MapShadows.SHADOW1]  
c:\11u\mgr\docbook\=c:\11u3\mgr\docbook\  
c:\11u\mgr\samples\=c:\11u3\mgr\samples\
```

## Management Portal

The page **System Administration > Configuration > Connectivity > Shadow Server Settings** displays shadow settings.

# [Mirrors]

This topic describes the [Mirrors] section of the CPF file.

# Mirrors

Describes mirrors the instance is a member of.

```
[Mirrors]      Name=GUID,2,1,Defined
Primary,QOSTimeout,0,UseSSL,VirtualAddress,0,AgentAddress,AgentPort,ArbiterNode,
AllowParallelDejournaling
```

## Description

Each entry consists of comma-separated values that define a mirror. From left to right:

- *Name* – Required. Unique name by which you can identify this mirror. An uppercase alphanumeric string with a maximum length of 15 characters.
- *GUID* – Required. An internal GUID, unique to this mirror.
- *2* – For internal use.
- *1* – For internal use.
- *Defined Primary* – Used to disable mirror failover, generally for maintenance purposes. Manipulated via the `nofailover` option when shutting down Caché or the **^MIRROR** routine. This contains the mirror name of the member which must be the primary. It is cleared automatically when that node starts up and becomes the primary.
- *QOSTimeout* – Quality of Service Timeout: the maximum time, in milliseconds, that a failover member waits for a response from the other failover member before taking action; also applies to the arbiter's wait for a failover member's response. The default is 8000ms; typically, deployments on physical (non-virtualized) hosts with a dedicated local network can reduce this setting if a faster response to outages is required. See [Configuring the Quality of Service \(QoS\) Timeout Setting](#) in the “Mirroring” chapter of the *Caché High Availability Guide* for more information on the QoS Timeout setting.
- *0* – For internal use.
- *UseSSL* – To provide security within a mirror, you can configure its nodes to use SSL/TLS. This provides for both authentication of one node to another, and for encrypted communication between nodes. To use SSL/TLS with a mirror, each member (failover or async) uses a pair of SSL/TLS configurations, `%Mirror_Client` and `%Mirror_Server`. These configurations must already exist on each member when SSL/TLS is enabled for the mirror. Instructions for setting up SSL are in the section [Creating and Editing SSL/TLS Configurations for a Mirror](#) of the “Using SSL/TLS with Caché” chapter of the *Caché System Security Guide*. Values are 0 (no, default) or 1 (yes).
- *VirtualAddress* – Specifies a virtual IP address. You can configure a mirror virtual IP address (VIP) so that all external clients (language bindings, ODBC/JDBC/SQL clients, direct-connect users, and so on) connect to the mirror through a single address. This virtual IP address is automatically bound to an interface on the current primary member. To use a VIP, which requires that both failover members be on the same subnet. For more information, see the [Configuring a Mirror Virtual IP \(VIP\)](#) section of the chapter “Mirroring” in the *Cache High Availability Guide*.
- *0* – For internal use.
- *ArbiterNode* – The network address of the arbiter configured for this mirror. The arbiter is an independent system hosting an ISCAgent with which the failover members of a mirror maintain continuous contact, providing them with the context needed to safely make failover decisions when they cannot communicate directly. .
- *ArbiterPort* – The port used by the configured arbiter's ISCAgent process (2188 by default).
- *FailoverCompressionMode* – Determines whether journal data is compressed before being transmitted from the primary to the backup. Values are 0 (System Selected, which optimizes for response time between failover members), 1 (Uncompressed) and 2 (Compressed).

- *AsyncCompressionMode* – Determines whether journal data is compressed before being transmitted from the primary to async members. Values are 0 (System Selected, which optimizes for network utilization), 1 (Uncompressed) and 2 (Compressed).
- *AllowParallelDejournaling* – Determines which type of mirror members can run parallel dejournaling updaters. Possible values are 0 (failover and disaster recovery members), 1 (failover members only), and 2 (all members).

## Examples

Each entry is on one line:

```
MIMI=C7BA9224-3851-47D4-83BD-0C01AEEACC76,2,1,,2000,0,0,,0,arblast|2188,0,0
```

## Management Portal

On the page **System Administration > Configuration > Mirror Settings > Create a Mirror > Join as Failover** (or **Join as Async**), enter the requested information.



# [MirrorMember]

This topic describes the parameters found in the [MirrorMember] section of the CPF file.

## AgentAddress

---

Not in use.

## Description

Not in use.



# AsyncMemberGUID

---

Async member GUID

```
[MirrorMember] AsyncMemberGUID=Name
```

## Description

You can create a mirror member called an *async member*, which can be configured to receive updates from one or more mirrors across the enterprise. This allows a single node to act as a comprehensive enterprise-wide data warehouse. Async members do not belong to a mirror and, therefore, are not candidates for failover.

For more information, see the [Async Mirror Members](#) section of the chapter “Mirroring” in the *Cache High Availability Guide*

## Examples

```
AsyncMemberGUID=06E1D307-59D9-4500-AA3B-4FF405E2A44D
```

## Range of Values

An alphanumeric string.

## Management Portal

No equivalent.

# AsyncMemberType

Indicates async type of DR, read-only, or read-write.

```
[MirrorMember] AsyncMemberType=n
```

## Description

AsyncMemberType indicates the async member is a disaster recovery (DR), read-only reporting, or read-write reporting async member.

- 0 - Disaster Recovery (DR). This is a disaster recovery async member. All its mirrored databases are read-only mirrored databases.
- 1 - Read-Only Reporting. This is a reporting async member. All its mirrored databases could be read-only or read-write databases. The default is read-only when the database is created.
- 2 - Read-Write Reporting. This is a reporting async member. All its mirrored databases could be read-only or read-write databases. The default is read-write when the database is created.

For more information, see the [Async Mirror Members](#) section of the chapter “Mirroring” in the *Cache High Availability Guide*

## Examples

```
AsyncMemberType=0
```

## Range of Values

0–2

## Management Portal

On the page **System Administration > Configuration > Mirror Settings > Join as Async**, fill in **Mirror Information** and select **Next**. On the page **Async Member Information**, in the **Async Member System Type** row, select a type from the drop-down list.

# AsyncUseSystemPurgeInterval

Indicates how mirror journal files are purged on the reporting async member.

```
[MirrorMember] AsyncUseSystemPurgeInterval=n
```

## Description

AsyncMemberType indicates how the reporting async member purges mirror journal files received from the primary failover member.

- 0 - Mirror journal files are purged immediately after being de journaled.
- 1 - Mirror journal files are purged according to the instances journal file purge criteria.

For more information, see the [Editing or Removing an Async Members](#) section of the chapter “Mirroring” in the *Cache High Availability Guide*.

## Examples

```
AsyncMemberType=0
```

## Range of Values

0–1

## Management Portal

On the page **System Administration > Configuration > Mirror Settings > Edit Async**, for a reporting async member, use the **Mirror Journal File Retention** drop-down to determine how mirror journal files are purged.

# JoinMirror

---

Determines whether the instance processes its mirror configuration at startup.

```
[MirrorMember]    JoinMirror=n
```

## Description

**JoinMirror** indicates whether the system processes its mirror configuration at startup.

- 1 (true) - The mirror configuration is processed and the system is considered a mirror member according to its configuration.
- 0 (false) - The mirror configuration is ignored and the system is not initialized as a mirror member. Use this if there is a problem in the configuration which prevents the system from starting or if the member must be reconfigured before joining the mirror. For example, if a system was the primary but no longer is, prevents the system from joining the mirror when it restarts and attempting to become the primary again, which could result in dual primaries.

## Examples

```
JoinMirror=0
```

## Range of Values

0–1.

## Management Portal

No equivalent.

---

# SystemName

---

Mirror member name.

```
[MirrorMember]      SystemName=Name
```

## Description

A name for the failover member you are configuring on this system (defaults to a combination of the system host name and the Caché instance name). Mirror member names cannot contain spaces, tabs, or the punctuation characters that follow:

: [ ] # ; / \* = ^ ~ ,

## Examples

SystemName=SystemA

## Range of Values

An alphanumeric string with a maximum length of 32. The name is converted to uppercase before storing.

## Management Portal

On the page **System Administration > Configuration > Mirror Settings > Create a Mirror, Join as Failover, or Join as Async**, enter the **Mirror Member Name**.

## ValidatedMember

---

Determines whether the instance should join the mirror in its previous role or obtain its new role from the current primary before joining the mirror.

```
[MirrorMember]    ValidatedMember=n
```

### Description

**ValidatedMember** indicates how the mirror member determines its role before joining the mirror.

- 1 (true) - The instance joins the mirror in its current role.
- 0 (false) - The instance contacts the primary to obtain its current role before joining the mirror. Use this when there have been role changes within the mirror while the instance and its ISCAgent were down or unreachable. For example, if a DR async has been promoted to backup while the former backup was down, set **ValidatedMember** to 0 before restarting the instance to ensure that the former backup receives its new role of DR async from the primary before restarting the mirror.

### Examples

```
JoinMirror=0
```

### Range of Values

0–1.

### Management Portal

No equivalent.

---

# VirtualAddressInterface

---

Local interface to primary virtual IP address.

```
[MirrorMember]    VirtualAddressInterface=IPaddress/mask
```

## Description

You can configure a mirror virtual IP address (VIP) so that all external clients (language bindings, ODBC/JDBC/SQL clients, direct-connect users, and so on) connect to the mirror through a single address. For more information, see the [Configuring a Mirror Virtual IP \(VIP\)](#) section of the chapter “Mirroring” in the *Cache High Availability Guide*.

## Examples

```
VirtualAddressInterface=192.168.12.0/23
```

## Range of Values

A valid IP address followed by a slash and a Classless Inter-Domain Routing (CIDR) mask.

## Management Portal

On the page **System Administration > Configuration > Mirror Settings > Create a Mirror** with the **Use Virtual IP** box selected, enter an **IP address** and a **Mask (CIDR format)** and select a **Network Interface**.





# [Miscellaneous]

This topic describes the parameters found in the [Miscellaneous] section of the CPF file.

# AsyncDisconnectErr

Enable/disable a process to receive disconnect errors asynchronously.

```
[Miscellaneous] AsyncDisconnectErr=n
```

## Description

This setting modifies the behavior of Caché when [DisconnectErr](#) is enabled. The value *n* may be 1 or 0:

- When 1 (true), the process receives an asynchronous <DSCON> error at the time a disconnect occurs on the device. This error will occur at the next command executed. Hang commands will be interrupted.
- When 0 (false) the process receives a <DSCON> error at the next read or write command.

[AsyncDisconnectErr](#) is only applicable to Telnet connections on Windows. It has no effect on any other device type or operating system. If [DisconnectErr](#) is set to 0 (false), then [AsyncDisconnectErr](#) has no effect.

## Examples

```
AsyncDisconnectErr=0
```

## Range of Values

0 or 1. The default is 0.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Compatibility**, in the [AsyncDisconnectErr](#) row, select **Edit** and choose true (selected) or false (cleared).

# AsynchError

Enable/disable processes to receive asynchronous errors.

```
[Miscellaneous]    AsynchError=n
```

## Description

The value *n* may be 1 or 0:

- When 1 (true) Caché processes can receive asynchronous errors.
- When 0 (false) they cannot.

## Examples

```
AsynchError=1
```

## Range of Values

0 or 1. The default is 1.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Compatibility**, in the [AsynchError](#) row, select **Edit** and choose true (selected) or false (cleared).

# BreakMode

---

Programmer mode response to the BREAK command.

```
[Miscellaneous] BreakMode=n
```

## Description

Caché programs can execute in two modes, depending on how Caché is entered: [application mode](#) and [programmer mode](#).

The BreakMode setting controls how a Caché process in programmer mode responds when it encounters a [BREAK](#) command that has no argument. The value *n* may be 1 or 0:

- When 1 (true) Caché enters the debugger or returns to the direct mode prompt with a <BREAK> error.
- When 0 (false) the BREAK command is ignored.

Application mode jobs always ignore argumentless BREAK commands.

## Examples

```
BreakMode=1
```

## Range of Values

0 or 1. The default is 1.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Compatibility**, in the [BreakMode](#) row, select **Edit** and choose true (selected) or false (cleared).

# CollectResourceStats

Controls whether system resource statistics are collected or not.

```
[Miscellaneous]      CollectResourceStats=n
```

## Description

Control whether system resource statistics are updated or not (seize, nseize, aseize, bseize). The default is that they are not updated.

- When 1 (true), Caché collects system resource statistics.
- When 0 (false), Caché does not collect system resource statistics. The default is false.

## Examples

```
CollectResourceStats=1
```

## Range of Values

0 or 1. The default is 0.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Compatibility**, in the [CollectResourceStats](#) row, select **Edit** and choose true (selected) or false (cleared).

# DisconnectErr

Enable/disable a process to receive disconnect errors.

```
[Miscellaneous]    DisconnectErr=n
```

## Description

This setting conditions how Caché responds to a disconnect of the principal I/O device. The value *n* may be 1 or 0:

- When 1 (true) the process receives a <DSCON> error when a disconnect is detected during a Caché Write or Read command.
- When 0 (false) the process exits without reporting an error to the application when a disconnect is detected.

You should be aware that if `DisconnectErr` is enabled, a process will continue to execute after its principal device has been disconnected. It is the responsibility of the application to detect the disconnect condition and exit gracefully. Use care when enabling `DisconnectErr`. The application must be prepared to recognize the <DSCON> error and handle it appropriately in error traps.

`DisconnectErr` is only applicable to TCP devices and to terminal devices where a disconnect can be recognized. Examples are modem controlled terminals and Windows Telnet, and Windows local cterm (TRM:) connections. `DisconnectErr` is only applicable to the principal device.

## Examples

```
DisconnectErr=0
```

## Range of Values

0 or 1. The default is 0.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Compatibility**, in the [DisconnectErr](#) row, select **Edit**. Choose true (selected) or false (cleared).

# EnableLongStrings

Enable/disable a large stack space for long strings.

```
[Miscellaneous]      EnableLongStrings=n
```

## Description

Caché allocates a fixed amount of space to hold the results of string operations, the string stack. If a string expression exceeds the amount of space allocated, a `<STRINGSTACK>` error results.

The `EnableLongStrings` setting is available, if needed, to significantly increase the size of the string stack. The value may be 1 or 0:

- Default. When 1 (true), Caché does not get a max string error and works with long strings. The memory allocated for actual long string values is not subject to the limit set by the `bbsiz` parameter and does not affect the `$STORAGE` value for the process. For any platform where memory is allocated up front, this is significant at ~7MB and ~14MB (8-bit unicode).
- When 0 (false), this option is disabled.

When you turn on long strings, every process begins with a larger fixed-size string stack. The string stack is for temporary string handling during a command. When you actually store a long string in a variable, the system allocates new memory for it. (There is optimization for two variables storing the same long string to point to the same `malloc()`'d memory (until one of them changes)). When you store a normal string (less than 32k) it comes out of the process buddy block memory (`bbsiz`, aka max mem per process), even when long strings are turned on.

When a process uses a long string, the memory for the string comes from `malloc()` instead of from the partition memory. Thus the `$STORAGE` value is not affected. When the string is killed, Cache releases (with `free()`) the memory that it got from `malloc()` for the long string. (Note that this is different for HP-UX. HP-UX does not release the memory; the process on HP-UX retains the memory after `free()` and that memory becomes available for a subsequent `malloc()`. If that memory were never used again, it is eventually paged out by the HP-UX memory manager (assuming there was some shortage of physical memory).

The value of this setting affects only Caché jobs started after it is changed. Existing jobs are not affected and maintain their current string stack size.

## Examples

```
EnableLongStrings=0
```

## Range of Values

0 or 1. The default is 1.

## Management Portal

On the page **System Administration > Configuration > System Configuration > Memory and Startup**, to enable support for long strings, select the **Enable Long Strings** check box. To disable support, clear the check box.

# FileMode

---

Enable/disable writing to a non-existent file.

```
[Miscellaneous]      FileMode=n
```

## Description

The value *n* may be 1 or 0:

- When 1 (true) and a file is opened for writing that does not exist, a new file is created.
- When 0 (false) and a file is opened for writing that does not exist, a new file is not created unless the N parameter was provided with the OPEN command.

Suppose Caché encounters an OPEN command such as:

```
OPEN "file.x": "WS"
```

When FileMode=1 the new file is created automatically, even though the N parameter is not specified with the OPEN command. The result when FileMode=1 is equivalent to adding the N parameter to each OPEN command, so that:

```
OPEN "file.x": "WS"
```

is equivalent to:

```
OPEN "file.x": "WNS"
```

On the other hand, when Caché encounters an OPEN command *and* no N parameter is provided *and* the file does not already exist, then if FileMode=0 there is no result from the OPEN command except that the process hangs until interrupted.

## Examples

```
FileMode=0
```

## Range of Values

0 or 1. The default value of FileMode is 0.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Compatibility**, in the [FileMode](#) row, select **Edit**. Choose true (selected) or false (cleared).



---

# GlobalKillEnabled

---

Enable/disable KILL of an unsubscripted global.

```
[Miscellaneous]      GlobalKillEnabled=n
```

## Description

Deprecated. The value *n* may be 1 or 0:

- When 1 (true) a KILL of an unsubscripted global is allowed, so you can kill all subscripts of a global with a single kill instead of killing them individually.
- When 0 (false) the KILL of an unsubscripted global results in a <PROTECT> error.

## Examples

```
GlobalKillEnabled=1
```

## Range of Values

0 or 1. The default is 1.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Compatibility**, in the [GlobalKillEnabled](#) row, select **Edit**. Choose true (selected) or false (cleared).

# IEEEError

Enables or disables \$DOUBLE returning INF and NAN values system-wide.

```
[Miscellaneous]      IEEEError=n
```

## Description

This property sets the \$DOUBLE function return-value behavior system-wide.

The property controls the issuing of INF, -INF, and NAN when a \$DOUBLE numeric operation cannot be resolved to a numeric value. It does not control the issuing of INF, -INF, and NAN in all cases. \$DOUBLE always returns INF, -INF, or NAN when you supply one of these strings as the input value, regardless of this property. Mathematical operations on \$DOUBLE numbers that result in an INF, -INF, or NAN are controlled by this property. These include arithmetic operations, exponentiation, and logarithmic and trigonometric functions.

- When 1 (true), \$DOUBLE generates Caché errors for unresolvable IEEE floating point conversions. The default is true.
- When 0 (false), \$DOUBLE returns INF (infinity), -INF, and NAN (Not A Number) for unresolvable IEEE floating point conversions.

## Examples

```
IEEEError=0
```

## Range of Values

0 or 1. The default is 1.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Compatibility**, in the [IEEEError](#) row, select **Edit** and choose true (selected) or false (cleared).

# LineRecall

Enable/disable command line recall for READ commands.

```
[Miscellaneous]      LineRecall=n
```

## Description

The value *n* determines whether the line recall feature is active only for command prompts, or for both command prompts and READ commands:

- When 1 (true) READ commands *and* command prompts can use the line recall feature.
- When 0 (false) only command prompts can use line recall.

## Examples

```
LineRecall=1
```

## Range of Values

0 or 1. The default is 1.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Compatibility**, in the [LineRecall](#) row, select **Edit**. Choose true (selected) or false (cleared).

# LogRollback

---

Enable/disable logging for transaction rollbacks.

```
[Miscellaneous]      LogRollback=n
```

## Description

The value *n* may be 1 or 0:

- When 1 (true) Caché logs transaction rollbacks to the console log file (that is, cconsole.log in the Caché system management directory, or the alternate *filename.log* named by the second piece of the [console](#) parameter).
- When 0 (false) it does not.

## Examples

```
LogRollback=0
```

## Range of Values

0 or 1. The default is 0.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Compatibility**, in the [LogRollback](#) row, select **Edit**. Choose true (selected) or false (cleared).

# MVDefined

Sets MVBasic handling of undefined variables system-wide.

```
[Miscellaneous]      MVDefined=n
```

## Description

This setting defines MVBasic behavior when it encounters a reference to an undefined variable.

- When 1 (true), if an MVBasic routine references an undefined variable, the system substitutes an empty string for the variable without showing an error.
- When 0 (false), if an MVBasic routine references an undefined variable, the system generates an error. The default is false.

## Examples

```
MVDefined=1
```

## Range of Values

0 or 1. The default is 0.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Compatibility**, in the [MVDefined](#) row, select **Edit** and choose true (selected) or false (cleared).

# NodeNameInPid

---

Behavior when Caché makes a reference to the special variable \$JOB.

```
[Miscellaneous]      NodeNameInPid=n
```

## Description

The value *n* may be 1 or 0:

- When 1 (true) \$JOB returns the process ID number of the current process, concatenated to the nodename.
- When 0 (false) \$JOB returns only the process ID number.

## Examples

NodeNameInPid=0

## Range of Values

0 or 1. The default is 0.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Compatibility**, in the [NodeNameInPid](#) row, select **Edit**. Choose true (selected) or false (cleared).

# NullSubscripts

Enable/disable null subscripts on global references.

```
[Miscellaneous]      NullSubscripts=n
```

## Description

The value  $n$  may be 1 or 0:

- When 1 (true) null subscripts are allowed on global references.
- When 0 (false) a null subscript causes a <SUBSCRIPT> error.

InterSystems recommends that this be set to 0.

## Examples

```
NullSubscripts=0
```

## Range of Values

0 or 1. The default is 0.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Compatibility**, in the [NullSubscripts](#) row, select **Edit**. Choose true (selected) or false (cleared).

# OldZU5

---

Clear global vectors when switching namespace.

[Miscellaneous]      OldZU5=*n*

## Description

When switching namespace to the same namespace via [\\$ZU\(5\)](#) or [ZN](#), clear global vectors.

- 0 - Switching to the same namespace is a NOOP. (default)
- 1 - Switching to the same namespace clears the globals vector cache.

## Examples

OldZU5=0

## Range of Values

0 or 1. The default is 0.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Compatibility**, in the [OldZU5](#) row, select **Edit**. Choose true (selected) or false (cleared).



---

# OpenMode

---

Read/write mode to use when opening sequential files.

```
[Miscellaneous]      OpenMode=n
```

## Description

The value *n* indicates the read/write mode you want used when opening sequential files and no mode is specified in the OPEN command:

- 0 means Read.
- 1 means Read-Write.

## Examples

OpenMode=0

## Range of Values

0 or 1. The default value of OpenMode is 0.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Compatibility**, in the [OpenMode](#) row, select **Edit**. Choose a mode, Read (0) or Read-Write (1).

# PopError

---

When to pop error handlers off the stack.

[Miscellaneous]      PopError=*n*

## Description

The value *n* may be 1 or 0:

- When 1 (true) Caché pops the current error handler off the stack when an error is triggered.
- When 0 (false) the normal behavior prevails: An error handler is popped when returning from the routine or procedure that established it.

## Examples

PopError=0

## Range of Values

0 or 1. The default is 0.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Compatibility**, in the [PopError](#) row, select **Edit**. Choose true (selected) or false (cleared).

---

# RefInKind

---

Result of \$NAME and \$QUERY when an extended global reference is the argument.

```
[Miscellaneous]      RefInKind=n
```

## Description

The value *n* may be 1 or 0:

- When 1 (true) \$NAME and \$QUERY return the global without an extended global reference.
- When 0 (false) the result is also an extended reference.

## Examples

```
RefInKind=0
```

## Range of Values

0 or 1. The default is 0.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Compatibility**, in the [RefInKind](#) row, select **Edit**. Choose true (selected) or false (cleared).

## ScientificNotation

---

Enables or disables lowercase "e" as scientific notation symbol system-wide.

```
[Miscellaneous]      ScientificNotation=n
```

### Description

This setting enables or disables using the lowercase "e" as scientific notation symbol system-wide.

- When 1 (true), Caché uses the lowercase "e" as scientific notation symbol. The default is true.
- When 0 (false), Caché does not use the lowercase "e" as scientific notation symbol.

### Examples

```
ScientificNotation=0
```

### Range of Values

0 or 1. The default is 1.

### Management Portal

On the page **System Administration > Configuration > Additional Settings > Compatibility**, in the [ScientificNotation](#) row, select **Edit** and choose true (selected) or false (cleared).

# SetZEOF

---

Behavior when Caché encounters an unexpected end-of-file when reading a sequential file.

```
[Miscellaneous]    SetZEOF=n
```

## Description

The value *n* may be 1 or 0:

- When 1 (true) Caché sets the special variable \$ZEOF to indicate that you have reached the end of the file.
- When 0 (false) Caché throws an <ENDOFFILE> error instead.

## Examples

```
SetZEOF=0
```

## Range of Values

0 or 1. The default is 0.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Compatibility**, in the [SetZEOF](#) row, select **Edit**. Choose true (selected) or false (cleared).

# ShutDownLogErrors

---

Enable/disable writing of Caché system error log entries (see [Caché System Error Log](#) in the “Monitoring Caché Using the Management Portal” chapter of the *Caché Monitoring Guide*) to the console log file on shutdown.

```
[Miscellaneous]      ShutDownLogErrors=n
```

## Description

This setting controls Caché behavior during execution of its SHUTDOWN procedure. The value *n* may be 1 or 0:

- When 1 (true) during shutdown Caché logs error information from the Caché system error log into the console log file (that is, cconsole.log in the Caché system management directory, or the alternate *filename.log* named by the second piece of the [console](#) parameter).
- When 0 (false) it does not.

## Examples

```
ShutDownLogErrors=0
```

## Range of Values

0 or 1. The default is 0.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Compatibility**, in the [ShutDownLogErrors](#) row, select **Edit**. Choose true (selected) or false (cleared).

# SwitchOSDir

Enable/disable switching current working directories when changing namespaces.

```
[Miscellaneous]      SwitchOSDir=n
```

## Description

This setting specifies what happens to the current working directory (for accessing files by relative pathname, etc.) when you switch to a new namespace. The value  $n$  may be 1 or 0:

- When 1 (true), if you change namespaces, the current working directory remains unaltered no matter what namespace you switch to.
- When 0 (false), if you change namespaces, the current working directory is changed to the directory of the default dataset for non-% globals of the new namespace. However, if this dataset is remote (networked to a different system), the current working directory is left unchanged.

Suppose SwitchOSDir is set to 1, *or* SwitchOSDir is set to 0 and the dataset is remote. In these cases, the current working directory does not change automatically as a result of changing the namespace, but you can always change the current working directory programmatically.

## Examples

```
SwitchOSDir=0
```

## Range of Values

0 or 1. The default is 0.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Compatibility**, in the [SwitchOSDir](#) row, select **Edit**. Choose true (selected) or false (cleared).

# SynchCommit

---

Enable/disable synchronizing TCOMMIT with the corresponding journal write operation.

```
[Miscellaneous]      SynchCommit=n
```

## Description

Every [TCOMMIT](#) command requests a flush of the journal data involved in that transaction to disk. This setting controls what happens at that point. The value *n* may be 1 or 0:

- When 1 (true) TCOMMIT does not complete until the journal data write operation completes.
- When 0 (false) TCOMMIT does not wait for the write operation to complete.

## Examples

```
SynchCommit=0
```

## Range of Values

0 or 1. The default is 0.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Compatibility**, in the [SynchCommit](#) row, select **Edit**. Choose true (selected) or false (cleared).



# TelnetNUL

Suppress/issue Telnet NUL at end-of-line for Telnet transmission. Windows only.

```
[Miscellaneous]      TelnetNUL=n
```

## Description

Suppress/issue Telnet NUL at end-of-line for Telnet transmission. If true (1), the Telnet virtual terminal does not issue a NUL character (ASCII 0) following a CR character (ASCII 13) at end-of-line during Telnet transmission. On output, a Telnet network virtual terminal (NVT) performs the following default end-of-line behavior: either issues a CR followed by an LF (carriage return character followed by a linefeed character), or issues a CR followed by NUL (if no LF is issued). The TelnetNUL setting affects the issuance of the NUL character in the second case. The value *n* may be 1 or 0

This setting applies only to Windows systems.

- When 1 (true) TelnetNUL suppresses (does not issue) a NUL character at end-of-line.
- When 0 (false) TelnetNUL issues a NUL character at end-of-line. Default.

## Examples

```
TelnetNUL=0
```

## Range of Values

0 or 1. The default is 0.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Compatibility**, in the [TelnetNUL](#) row, select **Edit**. Choose true (selected) or false (cleared).

# TruncateOverflow

---

Enable/disable the <MAXNUMBER> error on numeric overflow.

```
[Miscellaneous]      TruncateOverflow=n
```

## Description

Normally, when Caché encounters a number larger than 9223372036854775807 E127 (or smaller than -9223372036854775808 E127) it throws the <MAXNUMBER> error. You can prevent this with the `TruncateOverflow` setting. The value *n* may be 1 or 0:

- When 1 (true) the <MAXNUMBER> error is suppressed.
- When 0 (false) Caché throws the <MAXNUMBER> error as usual.

## Examples

```
TruncateOverflow=0
```

## Range of Values

0 or 1. The default is 0.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Compatibility**, in the [TruncateOverflow](#) row, select **Edit**. Choose true (selected) or false (cleared).

# Undefined

[Miscellaneous]      Undefined=*n*

## Description

Specifies how ObjectScript reacts when it attempts to fetch the value of a variable that has not been defined. The value of Undefined may be 0, 1, or 2:

- 0 - Throw an <UNDEFINED> error. (default)
- 1 - If the undefined variable has subscripts, return a null string, but if the undefined variable is single-valued, throw an <UNDEFINED> error.
- 2 - Always return a null string.

## Examples

Undefined=0

## Range of Values

The value of Undefined may be 0, 1, or 2.

## Management Portal

On the **Compatibility Settings** page (**System Administration** > **Configuration** > **Additional Settings** > **Compatibility**), in the [Undefined](#) row, select **Edit**. Choose true (selected) or false (cleared).

# UseNagleAlgorithm

---

Enable/disable the Nagle algorithm for Telnet.

```
[Miscellaneous]      UseNagleAlgorithm=n
```

## Description

The Nagle algorithm makes Telnet more efficient. It reduces the number of IP packets sent over the network by consolidating messages that are sent within a small time interval into a single IP packet. When the Nagle algorithm is enabled, the operating system waits some interval before actually committing the data from a send command, in the hopes that the application will call send again with more data that can be consolidated with the first. For details see

<http://www.ietf.org/rfc/rfc896.txt>

The value of the UseNagleAlgorithm setting may be 1 or 0:

- When 1 (true), use the Nagle algorithm.
- When 0 (false), do not use it.

## Examples

```
UseNagleAlgorithm=0
```

## Range of Values

0 or 1. The default is 0.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Compatibility**, in the [UseNagleAlgorithm](#) row, select **Edit**. Choose true (selected) or false (cleared).

---

# ViewPastData

---

Enable \$VIEW to examine data outside of Caché memory area.

```
[Miscellaneous]      ViewPastData=n
```

## Description

Enable \$VIEW command to examine data outside of Caché memory area.

- 0 - \$VIEW command throws an error (default)
- 1 - \$VIEW command does not throw an error.

## Examples

```
ViewPastData=0
```

## Range of Values

0 or 1. The default is 0.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Compatibility**, in the [ViewPastData](#) row, select **Edit**. Choose true (selected) or false (cleared).

# ZaMode

---

ZALLOCATE (ZA) and ZDEALLOCATE (ZD) behavior.

[Miscellaneous]      ZaMode=*n*

## Description

The value *n* determines how the ZALLOCATE (ZA) and ZDEALLOCATE (ZD) commands behave, according to Caché rules or DSM-11 rules:

- 0 chooses Caché rules, which means that ZA and ZD behave exactly like LOCK + and LOCK -.
- 1 chooses DSM-11 rules, which means that ZA locks can only be unlocked by ZD and LOCK + locks can only be unlocked by LOCK.

For details, see the section “[DSM-11 Language Compatibility](#)” in the “Open M Language Compatibility” chapter of *Using Caché ObjectScript*.

## Examples

ZaMode=0

## Range of Values

0 or 1. The default is 0.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Compatibility**, in the [ZaMode](#) row, select **Edit**. Choose true (selected) or false (cleared).

# ZDateNull

\$ZDATE response to an invalid value.

```
[Miscellaneous]      ZDateNull=n
```

## Description

The value *n* determines how a [\\$ZDATE](#) call responds when triggered by an invalid value:

- When 1 (true) \$ZDATE returns a null value.
- When 0 (false) \$ZDATE returns an error.

## Examples

```
ZDateNull=0
```

## Range of Values

0 or 1. The default is 0.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Compatibility**, in the [ZDateNull](#) row, select **Edit**. Choose true (selected) or false (cleared).





# [Monitor]

This topic describes the parameters found in the [Monitor] section of the CPF file.

# PatrolCollectionInterval

---

Number of seconds between when Caché collects data and makes it available to Patrol.

```
[Monitor]    PatrolCollectionInterval=n
```

## Description

Number of seconds between each time Caché collects data and makes it available to Patrol. The default is 30 seconds. Min=1, max = 900

## Examples

```
PatrolCollectionInterval=120
```

## Range of Values

1 to 900.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Monitor**, for the **Patrol Collection Interval Seconds** setting, enter number of seconds.

---

# PatrolDisplayMode

---

Controls how the monitoring data is displayed in the Patrol console.

```
[Monitor]    PatrolDisplayMode=n
```

## Description

0, 1, or 2 controls how the monitoring data is displayed in the Patrol console. Options are as follows: F

- 0 means Total: display the total counts since the collection was started. This is the default.
- 1 means Delta: display the count for the last collection period.
- 2 means Rate: display a calculated count per second.

## Examples

```
PatrolDisplayMode=1
```

## Range of Values

0, 1, or 2. Default is 0.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Monitor**, for the **Patrol Display Mode** setting, choose Total, Delta, or Rate.

## PatrolEnabled

---

Sets whether the connection to Patrol starts automatically when Caché starts.

```
[Monitor]    PatrolEnabled=n
```

### Description

0 or 1. When 1 (true) the connection to Patrol starts automatically when Caché starts up. When 0 (false) it does not. The default is 0.

### Examples

```
PatrolEnabled=1
```

### Range of Values

0 or 1.

### Management Portal

On the page **System Administration > Configuration > Additional Settings > Monitor**, for the **Start Patrol at System Startup** setting, choose Yes or No.

---

# PatrolTopProcesses

---

Number of processes to be displayed on the Patrol console.

```
[Monitor]    PatrolTopProcesses=n
```

## Description

Any non-zero value sets the number of processes to be displayed in the Process Status window on the Patrol console. This window shows the “top” processes as sorted by global or routine activity. The default number of processes is 20. A value of 0 tells the Patrol utility to stop calculating the top processes, potentially saving significant work on systems with a lot of processes.

## Examples

```
PatrolTopProcesses=10
```

## Range of Values

Default is 20.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Monitor**, for the **Patrol Top Processes to Monitor** setting, enter a number of processes.

## SNMPEnabled

---

Enable/disable automatic SNMP startup.

```
[Monitor]      SNMPEnabled=n
```

### Description

The value *n* may be 1 or 0:

- When 1 (true) the SNMP agent automatically starts when Caché starts up.
- When 0 (false) it does not.

### Examples

SNMP=0

### Range of Values

0 or 1. The default is 0.

### Management Portal

On the page **System Administration > Configuration > Additional Settings > Monitor**, for the **Start SNMP Agent at System Startup** setting, choose Yes (true) or No (false).

---

# WMIEnabled

---

Enable/disable Windows Management Instrumentation (WMI) monitoring.

```
[Monitor]    WMIEnabled=n
```

## Description

The value *n* may be 1 or 0:

- When 1 (true) Windows Management Instrumentation (WMI) monitoring is enabled.
- When 0 (false) it is disabled.

## Examples

WMI=0

## Range of Values

0 or 1. The default is 0.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Monitor**, for the **WMI Enabled** setting, choose Yes (true) or No (false).





# [Namespaces]

This topic describes the parameters found in the [Namespaces] section of the CPF file.

# Namespace

Define Caché namespaces.

```
[Namespaces]      Name=globals,routines,temporary
```

## Description

Each entry provides three comma-separated values that define a namespace. Only the first field is required. If the other fields are not specified, they are set to the system default. From left to right:

- Default database name for globals. Required.
- Default database name for routines and classes. If the database is not specified, it defaults to the globals database.
- Default database name for temporary storage. If the database is not specified, it defaults to CACHETEMP.

Globals starting with a % are mapped to CACHESYS unless mapped to another database by a user defined mapping.

Temporary globals are by default mapped to CACHETEMP. These include globals starting with CacheTemp\* and mtemp\*.

In all namespaces, routine and classes that start with a % come from the CACHELIB database with the following exceptions:

- Routines and classes starting with %SYS.\* come from CACHESYS (supplied by Caché).
- Routines and classes starting with %Z\* and %z\* come from CACHESYS (user defined routines and classes).
- Routines and classes that are explicitly mapped from another database by the user using routine or package mapping.

Note that routines that reside in the CACHESYS database have special security privileges including the ability to modify the roles and other security attributes of the process executing them.

Caché adds entries to the CPF file automatically as you add and configure namespaces using the Management Portal (see the instructions below).

## Examples

In the [Namespaces] section, each entry appears all on one line:

```
[Namespaces]
%SYS=CACHESYS
DOCBOOK=DOCBOOK
SAMPLES=SAMPLES
USER=USER

SALES=SALESGBL,SALESRTN
; Globals and routines/classes split into separate databases.
BILLING=BILLING,,TEMPDATA
; Globals and routines/classes in the same database,
; temporary globals are mapped to the databases TEMPDATA
```

## Range of Values

A valid namespace or database name.

## Management Portal

On the page **System Administration > Configuration > System Configuration > Namespaces**, to add a new entry, select **Create New Namespace**. To edit an existing entry, select **Edit** in that entry's row.

# [Shadows]

This topic describes the parameters found in the [Shadows] section of the CPF file.

# Shadow

Define shadow servers.

```
[ Shadows ]
name=Address,Enabled,JournalDirectory,DaysBeforePurge,SSLConfig,FilterRoutine,MaxErrors
```

## Description

Each entry consists of a shadow server has seven comma-separated properties. (Note that shadow names cannot contain the tilde (~) character.) From left to right:

- *Address* — Combined IP address and port in the form *address|port*.
- *Enabled* — Starts (1) a defined shadow server or does not start the shadow server (0). The default is to not start (0).
- *JournalDirectory* — Full path name of the journal file directory. The default is the shadow subdirectory of the manager directory
- *DaysBeforePurge* — Number of days to keep the shadow copies of the source journal files. Default is 0. By default, the Caché destination purges its copy of a journal file as soon as it finishes dejournaling as long as it does not contain open transactions. Enter a value in this field to keep the shadow copies of the journal files on the destination longer.
- *SSLConfig* — Name an existing client configuration. Default is blank to not use SSL for the shadow connection.
- *FilterRoutine* — Name (omit the leading ^) of an optional filter routine the shadow uses to filter journal records before dejournaling them on the shadow. The routine should be in the %SYS namespace.
- *MaxErrors* — Number of shadowing errors from 0 to 200 which Caché should retain. The default is 10
- *DisableJournalUpdates=n* If 1, updates that the shadow applies to the shadow databases are not journaled, regardless of the journal settings on the databases. If 0, shadow updates are journaled. Default is 0.

For more information about defining and managing shadow servers, see the chapter [Shadowing](#) in the *Caché Data Integrity Guide*. For configuring shadows using APIs, see documentation for the %SYS.SYS.Shadowing.Shadow class.

## Examples

The shadow is named SHADOW1 at the shown address and port. The shadow is running. The rest of the properties are left as the default values.

```
SHADOW1=ShadowMachineIP|56775,1
```

The same shadow as the first example, but not running.

```
SHADOW1=ShadowMachineIP|56775,0
```

With *DaysBeforePurge* set to 4:

```
SHADOW1=ShadowMachineIP|56775,0,,4
```

## Range of Values

As shown above for each property

## Management Portal

On the page **System Administration > Configuration > Connectivity > Shadow Server Settings**, select **Add New Shadow**.

# [SQL]

This topic describes the parameters found in the [SQL] section of the CPF file.

# AllowRowIDUpdate

---

Enable/disable SQL row ID update.

```
[SQL]      AllowRowIDUpdate=n
```

## Description

The value *n* may be 1 or 0:

- When 1 (true) the table compiler omits the checks to see if the RowID field(s) are being updated.
- When 0 (false) the checks are performed.

Set to 1 only if you are doing your own filing in a BEFORE trigger and using the %SkipFiling flag. Otherwise, use the default of 0.

## Examples

```
AllowRowIDUpdate=0
```

## Range of Values

0 or 1. The default is 0.

## Management Portal

No equivalent.

---

# BiasQueriesAsOutlier

---

Enable/disable query optimization biased toward outliers.

```
[SQL]      BiasQueriesAsOutlier=n
```

## Description

The value *n* may be 1 or 0:

- When 1 (true), InterSystems SQL optimizes for queries that primarily return [outlier values](#).
- When 0 (false), it does not.

## Examples

```
BiasQueriesAsOutlier=0
```

## Range of Values

0 or 1. The default is 0.

## Management Portal

On the page **System Administration > Configuration > SQL and Object Settings > General SQL Settings**, for the **Bias queries as outlier** setting, choose Yes (true) or No (false).

## Comment

---

Enable/disable retaining embedded SQL statements as comments in source code.

```
[SQL]      Comment=n
```

### Description

The value *n* may be 1 or 0:

- When 1 (true) embedded SQL statements are retained as comments in the source code (.INT) version of the routine.
- When 0 (false) they are not.

### Examples

```
Comment=1
```

### Range of Values

0 or 1. The default is 1.

### Management Portal

On the page **System Administration > Configuration > SQL and Object Settings > General SQL Settings**, for the **Retains SQL Statements as Comments in .INT Code** setting, choose Yes (true) or No (false).



---

# DBMSSecurity

---

Enable/disable SQL security.

```
[SQL]      DBMSSecurity=n
```

## Description

The value *n* may be 1 or 0:

- When 1 (true) Caché SQL security is enabled. This means privilege-based table/view/procedure security is active. A user can only perform actions on a table or view for which that user has been granted privilege.
- When 0 (false) Caché SQL Security is disabled. This means privilege-based table/view/procedure security is suppressed. A user can perform actions on a table or view even if that user has no privileges to do so.

## Examples

```
DBMSSecurity=1
```

## Range of Values

0 or 1. The default is 1.

## Management Portal

On the page **System Administration > Configuration > SQL and Object Settings > General SQL Settings**, for the **SQL Security Enabled** setting, choose Yes (true) or No (false).

# DDLDefineBitmapExtent

Determines whether a table created by a DDL statement defines a bitmap extent index.

```
[SQL]      DDLDefineBitmapExtent=n
```

## Description

The value *n* may be 1 or 0:

- When 1 (true), a table created by a DDL CREATE TABLE statement defines a bitmap extent index. The index improves the performance of [COUNT\(\\*\)](#), a function that returns the number of rows in the table.
- When 0 (false), the table does not define a bitmap extent index.

## Examples

```
DDLDefineBitmapExtent=1
```

## Range of Values

0 or 1. The default is 1.

## Management Portal

On the page **System Administration > Configuration > SQL and Object Settings > General SQL Settings**, for the **Do classes created by a DDL CREATE TABLE statement define a bitmap extent index for the class?** setting, choose Yes (true) or No (false).

# DDLFinal

Determines whether a class created by a DDL statement is final.

```
[SQL]      DDLFinal=n
```

## Description

The value *n* may be 1 or 0:

- When 1 (true), a class created by a DDL CREATE TABLE statement is [final](#), meaning it cannot have subclasses.
- When 0 (false), the class is not final.

## Examples

```
DDLFinal=0
```

## Range of Values

0 or 1. The default is 1.

## Management Portal

On the page **System Administration > Configuration > SQL and Object Settings > General SQL Settings**, for the **Are classes created via DDL CREATE TABLE statement defined as Final?** setting, choose Yes (true) or No (false).

## DDLNo201

Suppress/return error upon CREATE of a previously existing table.

```
[SQL]      No201=n
```

### Description

The value *n* may be 1 or 0:

- If 1 (true) when an attempt is made to CREATE a previously existing table, Caché suppresses the SQLCODE -201 error.
- If 0 (false) Caché returns the error.

### Examples

```
DDLNo201=0
```

### Range of Values

0 or 1. The default is 0.

### Management Portal

On the page **System Administration > Configuration > SQL and Object Settings > General SQL Settings**, for the **Allow DDL CREATE TABLE for Existing Table** setting, choose Yes (true) or No (false).

# DDLNo30

Suppress/return error upon DROP of a nonexistent table.

```
[ SQL ]      DDLNo30=n
```

## Description

The value *n* may be 1 or 0:

- If 1 (true) when an attempt is made to DROP a nonexistent table, Caché suppresses the SQLCODE -30 error.
- If 0 (false) Caché returns the error.

## Examples

```
DDLNo30=0
```

## Range of Values

0 or 1. The default is 0.

## Management Portal

On the page **System Administration > Configuration > SQL and Object Settings > General SQL Settings**, for the **Allow DDL DROP of Non-existent Table** setting, choose Yes (true) or No (false).

## DDLNo307

Suppress/return error upon CREATE of a primary key constraint when one exists.

```
[SQL]      DDLNo307=n
```

### Description

The value *n* may be 1 or 0:

- If 1 (true) when an attempt is made to CREATE a primary key constraint to a table through DDL, and a primary key constraint already exists for that table, Caché suppresses the SQLCODE -307 error.
- If 0 (false) Caché returns the error.

### Examples

```
DDLNo307=0
```

### Range of Values

0 or 1. The default is 0.

### Management Portal

On the page **System Administration > Configuration > SQL and Object Settings > General SQL Settings**, for the **Allow Create Primary Key Through DDL When Key Exists** setting, choose Yes (true) or No (false).

# DDLNo311

Suppress/return error upon ADD a foreign key, when a key of that name already exists.

```
[SQL]      DDLNo311=n
```

## Description

The value *n* may be 1 or 0:

- If 1 (true) when an attempt is made to ADD a foreign key, even if a key of that name already exists, Caché suppresses the SQLCODE -311 error.
- If 0 (false) Caché returns the error.

## Examples

```
DDLNo311=0
```

## Range of Values

0 or 1. The default is 0.

## Management Portal

On the page **System Administration > Configuration > SQL and Object Settings > General SQL Settings**, for the **Allow DDL ADD Foreign Key Constraint when Foreign Key Exists** setting, choose Yes (true) or No (false).

## DDLNo315

---

Suppress/return error upon DROP of a nonexistent constraint.

```
[SQL]      DDLNo315=n
```

### Description

The value *n* may be 1 or 0:

- If 1 (true) when an attempt is made to DROP a nonexistent constraint, Caché suppresses the SQLCODE -315 error.
- If 0 (false) Caché returns the error.

### Examples

```
DDLNo315=0
```

### Range of Values

0 or 1. The default is 0.

### Management Portal

On the page **System Administration > Configuration > SQL and Object Settings > General SQL Settings**, for the **Allow DDL DROP of Non-constraint** setting, choose Yes (true) or No (false).



# DDLNo324

Suppress/return error upon CREATE of a previously existing index.

```
[ SQL ]      DDLNo324=n
```

## Description

The value *n* may be 1 or 0:

- If 1 (true) when an attempt is made to CREATE a previously existing index, Caché suppresses the SQLCODE -324 error.
- If 0 (false) Caché returns the error.

## Examples

```
DDLNo324=0
```

## Range of Values

0 or 1. The default is 0.

## Management Portal

On the page **System Administration > Configuration > SQL and Object Settings > General SQL Settings**, for the **Allow DDL CREATE INDEX for Existing Index** setting, choose Yes (true) or No (false).

## DDLNo333

Suppress/return error upon DROP of a nonexistent index.

```
[SQL]      DDLNo333=n
```

### Description

The value *n* may be 1 or 0:

- If 1 (true) when an attempt is made to DROP a nonexistent index, Caché suppresses the SQLCODE -333 error.
- If 0 (false) Caché returns the error.

### Examples

```
DDLNo333=0
```

### Range of Values

0 or 1. The default is 0.

### Management Portal

On the page **System Administration > Configuration > SQL and Object Settings > General SQL Settings**, for the **Allow DDL DROP of Non-existent Index** setting, choose Yes (true) or No (false).

---

# DDLSQLOnlyCompile

---

Enable/disable an SQL-only compile.

```
[SQL]      DDLSQLOnlyCompile=n
```

## Description

The value *n* may be 1 or 0:

- When 1 (true), any class compilation performed as a result of executing a DDL statement compiles the class with the *q* (sqlonly) flag.
- When 0 (false), the *q* flag is not used.

## Examples

```
DDLSQLOnlyCompile=0
```

## Range of Values

0 or 1. The default is 0.

## Management Portal

No equivalent.

# DDLUseSequence

Determines which function a table created by a DDL statement uses for ID assignment.

```
[SQL]      DDLUseSequence=n
```

## Description

The value *n* may be 1 or 0:

- When 1 (true), a table created by DDL CREATE TABLE uses **\$SEQUENCE** for ID assignment.
- When 0 (false), the table uses **\$INCREMENT**.

**\$SEQUENCE** is the default function, and is better suited for ID assignment. For a comparison of the two functions, see the [\\$INCREMENT](#) or [\\$SEQUENCE](#) section in the chapter “\$INCREMENT” in *Caché ObjectScript Reference*.

## Examples

```
DDLUseSequence=0
```

## Range of Values

0 or 1. The default is 1.

## Management Portal

On the page **System Administration > Configuration > SQL and Object Settings > General SQL Settings**, for the **Do classes created by a DDL CREATE TABLE statement use \$Sequence for ID assignment?** setting, choose Yes (true) or No (false).

# DefaultSchema

Default SQL schema name.

```
[SQL]      DefaultSchema=n
```

## Description

The value *n* is a string defining the default SQL schema name. The maximum length of the string is 128 characters. The default string is:

`SQLUser`

The default schema name comes into play when an unqualified table name is encountered in an SQL statement and there is no `#import` statement specified. This setting has nothing to do with the mappings between SQL schema names and the class package name; it only specifies the default schema name.

If you specify `_CURRENT_USER` as the default schema name, the default schema name becomes the username of the currently logged-in process or, if the process has not logged in, `SQLUser` becomes the default schema name.

If you specify `_CURRENT_USER/name` as the default schema name, where *name* is any string of your choice, then the default schema name becomes the username of the currently logged-in process or, if the process has not logged in, *name* is used as the default schema name. For example, `_CURRENT_USER/HMO` uses HMO as the default schema name if the process has not logged in.

## Examples

```
DefaultSchema=SQLUser
```

## Range of Values

As described above.

## Management Portal

On the page **System Administration > Configuration > SQL and Object Settings > General SQL Settings**, for the **Default SQL Schema Name** setting, enter a string of characters.

## DelimitedIds

Enable/disable interpreting double-quoted strings as delimited identifiers.

```
[SQL]    DelimitedIds=n
```

### Description

The value *n* may be 1 or 0:

- When 1 (true) a double-quoted string ("My String") is considered a delimited identifier within an SQL statement.
- When 0 (false) a double-quoted string ("My String") is considered a string constant or literal string.

### Examples

```
DelimitedIds=0
```

### Range of Values

0 or 1. The default is 1.

### Management Portal

On the page **System Administration > Configuration > SQL and Object Settings > General SQL Settings**, for the **Support Delimited Identifiers** setting, choose Yes (true) or No (false).

---

# DropDelete

---

DROP TABLE behavior.

```
[SQL]      DropDelete=n
```

## Description

The value *n* may be 1 or 0:

- When 1 (true) a DROP TABLE statement deletes the table *and* the table's data.
- When 0 (false) it deletes the table, but does not delete the data.

## Examples

DropDelete=1

## Range of Values

0 or 1. The default is 1.

## Management Portal

On the page **System Administration > Configuration > SQL and Object Settings > General SQL Settings**, for the **Does DDL DROP TABLE Delete the Table's Data** setting, choose Yes (true) or No (false).

## ECPSync

---

Ensures that the server and client cache are in sync.

```
[SQL]      ECPSync=n
```

### Description

The value *n* may be 1 or 0:

- When 1 (true): when an SQL Select statement is executed, forces all pending ECP requests to the DB-server. On completion it guarantees that the client cache is in sync.
- When 0 (false): Does not guarantee that the client cache is in sync. Default.

### Examples

```
ECPSync=1
```

### Range of Values

0 or 1. The default is 0.

### Management Portal

On the page **System Administration > Configuration > SQL and Object Settings > General SQL Settings**, for the **SQL SELECT Synchronizes ECP Cache** setting, choose Yes (true) or No (false).



# ExtrinsicFunctions

Enable/disable extrinsic functions in SQL statements.

```
[SQL]      ExtrinsicFunctions=n
```

## Description

The value *n* may be 1 or 0:

- When 1 (true) extrinsic functions can be used in SQL statements through ODBC, JDBC, and Dynamic Query.
- When 0 (false) this feature is disabled.

## Examples

```
ExtrinsicFunctions=0
```

## Range of Values

0 or 1. The default is 0.

## Management Portal

On the page **System Administration > Configuration > SQL and Object Settings > General SQL Settings**, for the **Allow Extrinsic Functions in SQL Statements** setting, choose Yes (true) or No (false).

# FastDistinct

Enable/disable SQL DISTINCT optimization.

```
[SQL]      FastDistinct=n
```

## Description

The value *n* may be 1 or 0:

- When 1 (true) SQL queries involving DISTINCT and GROUP BY run more efficiently by making better use of indices (if available).

**CAUTION:** The values returned by such queries are collated in the same way they are stored within the index. This means the results of such queries may be uppercase. This may have an effect on case-sensitive applications.

- When 0 (false) optimization of DISTINCT queries is disabled.

## Examples

```
FastDistinct=1
```

## Range of Values

0 or 1. The default is 1.

## Management Portal

On the page **System Administration > Configuration > SQL and Object Settings > General SQL Settings**, for the **DISTINCT optimization turned ON** setting, choose Yes (true) or No (false).

# IdKey

Primary key constraint behavior.

```
[SQL]      IdKey=n
```

## Description

The value *n* may be 1 or 0:

- 1 (true) - When a Primary Key constraint is specified, it does *not* also become the IDKey index in the class definition.
- 0 (false) - A Primary Key constraint specified also becomes the IDKey index in the class definition; this generally gives better performance, but means that the Primary Key fields cannot be updated.

## Examples

IdKey=1

## Range of Values

0 or 1. The default is 1.

## Management Portal

On the page **System Administration > Configuration > SQL and Object Settings > General SQL Settings**, for the **Are Primary Keys Created through DDL not ID Keys** setting, choose Yes (true) or No (false).

# IdTrxFrom

The “From” list of characters for Identifier Translation.

```
[SQL]      IdTrxFrom=n
```

## Description

The value *n* is a string of characters that provides the “From” list for Identifier Translation mappings. The maximum length of the string is 256 characters. The default string is:

```
~ `@#$$%^&*()_+-=[ ]\{| ; ' : " , . / < > ? " .
```

These mappings filter/modify valid SQL identifier characters when translating SQL identifiers into Objects identifiers. When converting an SQL identifier to an Objects identifier at runtime, the characters in the “From” string are converted to the characters in the “To” string. Also see [IdTrxTo](#).

## Examples

```
IdTrxFrom=~ `@#$$%^&*()_+-=[ ]\{| ; ' : " , . / < > ?
```

## Range of Values

As described above.

## Management Portal

On the page **System Administration > Configuration > SQL and Object Settings > General SQL Settings**, in the **Identifier translation - From** field, enter a string of characters.

# IdTrxTo

The “To” list of characters for Identifier Translation.

```
[SQL]      IdTrxTo=n
```

## Description

The value *n* is a string of characters that provides the “To” list for Identifier Translation mappings. Also see [IdTrxFrom](#).

## Examples

```
IdTrxTo=
```

## Range of Values

The maximum length of the string is 256 characters. The default is an empty string.

## Management Portal

On the page **System Administration > Configuration > SQL and Object Settings > General SQL Settings**, in the **Identifier translation - To** field, enter a string of characters.

# JavaClassPath

---

List of additional .jar files to load when the JDBC service is started.

```
[SQL]      JavaClassPath=jar file, jar file...
```

## Description

Default .jar files that are loaded when JDBC starts are specified in the cachegateway.jar file. If you want additional .jar files to load when JDBC gateway starts, specify them with their full path in the JavaClassPath setting.

## Examples

```
JavaClassPath=C:/JarFolder/example.jar
```

## Range of Values

One or more valid .jar files.

## Management Portal

No equivalent

# JavaHome

---

Specifies the Java version to use with the JDBC Gateway.

```
[SQL]      JavaHome=directory
```

## Description

This setting specifies the full path of the directory that contains the non-default version of Java that you want to run when you start the JDBC Gateway. If this setting is empty, then your system's default version of Java is used.

## Examples

```
JavaHome=C:/Javaversion
```

## Range of Values

A valid directory.

## Management Portal

On the page **System Administration > Configuration > Connectivity > JDBC Gateway Server**, in the **Java Home Directory** field, enter the Java directory.

# JDBCGatewayAddress

---

Address of JDBC Gateway.

```
[SQL]      JDBCGatewayAddress=nnn.n.n.n
```

## Description

The value *n* is the Internet address for the JDBC Gateway.

## Examples

```
JDBCGatewayAddress=127.0.0.1
```

## Range of Values

A valid IP address.

## Management Portal

No equivalent.



# JDBCGatewayJVMArgs

---

Optional JVM arguments to use when starting the JDBC Gateway.

```
[SQL]      JDBCGatewayJVMArgs=args
```

## Description

Optional JVM arguments to include in the command line when starting the JDBC Gateway.

## Examples

This entry defines the minimum (-Xms) and maximum (-Xmx) JVM heap size:

```
JDBCGatewayJVMArgs=-Xms64m -Xmx512m
```

## Range of Values

A string of 0–1024 characters.

## Management Portal

On the page **System Administration > Configuration > Connectivity > JDBC Gateway Server**, in the **JVM Arguments** field, enter arguments.

---

# JDBCGatewayLog

---

Name of the JDBC Gateway log file.

```
[SQL]      JDBCGatewayLog=n
```

## Description

The value *n* is a string giving the name of the JDBC Gateway log file. There is no default name. You must specify a name if you wish logging to occur. The maximum string length is 256 characters.

## Examples

JDBCGatewayLog=

## Range of Values

As described above.

## Management Portal

On the page **System Administration > Configuration > Connectivity > JDBC Gateway Server**, in the **Log File** field, enter the name of the JDBC Gateway log file.

---

# JDBCGatewayPort

---

Port number for the JDBC Gateway.

```
[SQL]      JDBCGatewayPort=n
```

## Description

The value *n* is the port number for the JDBC Gateway; a five-digit integer.

## Examples

```
JDBCGatewayPort=31486
```

## Range of Values

1–99999. The default is 62972. If you have multiple Caché instances on your system, this port number is incremented for each successive Caché instance.

## Management Portal

On the page **System Administration > Configuration > Connectivity > JDBC Gateway Server**, in the **Port** field, enter the number of the JDBC Gateway port.

# JDBCGatewayUsePassphrase

---

Enable/disable the passphrase requirement for JDBC connections.

```
[SQL]      JDBCGatewayUsePassphrase=n
```

## Description

The value *n* may be 1 or 0:

- When 1 (true), a passphrase is required for JDBC connection.
- When 0 (false), no passphrase is required.

## Examples

JDBCGatewayUsePassphrase=0

## Range of Values

1 or 0. The default is 0.

## Management Portal

On the page **System Administration** > **Configuration** > **Connectivity** > **JDBC Gateway Server**, select the **Use Passphrase** check box to require a passphrase.

---

# LockThreshold

---

SQL lock threshold.

```
[SQL]      LockThreshold=n
```

## Description

The value *n* is the lock threshold. This is the number of inserts, updates, or deletes for a single table within a single transaction that will trigger a table-level lock when reached. For example, if the lock threshold is 1000 and a process starts a transaction and then inserts 2000 rows, after the 1001<sup>st</sup> row is inserted the process will attempt to acquire a table-level lock instead of continue to lock individual rows. This is to help keep the lock table from becoming too full.

## Examples

```
LockThreshold=1000
```

## Range of Values

Any integer. The default is 1000.

## Management Portal

On the page **System Administration > Configuration > SQL and Object Settings > General SQL Settings**, in the **Lock Threshold** field, enter a number.

# LockTimeout

---

SQL lock timeout.

```
[SQL]      LockTimeout=n
```

## Description

The value *n* is the lock timeout (in seconds) for Caché locks made during execution of SQL statements.

## Examples

```
LockTimeout=10
```

## Range of Values

0–32767 seconds (up to 9 hours). The default is 10 seconds.

## Management Portal

On the page **System Administration > Configuration > SQL and Object Settings > General SQL Settings**, for the **LockTimeout (in seconds)** setting, enter a number.

---

# ODBCVarcharMaxlen

---

Define the MaxLen for ODBC fields of type VarChar.

```
[SQL]      ODBCVarcharMaxlen=n
```

## Description

The value *n* is the MaxLen (maximum length) that Caché will report to ODBC for fields with the data type VarChar.

## Examples

```
ODBCVarcharMaxlen=4096
```

## Range of Values

Any integer. The default is 4096.

## Management Portal

No equivalent.

---

# QueryProcedures

---

Defines whether all class queries project as SQL Stored Procedures.

```
[SQL]      QueryProcedures=n
```

## Description

Defines whether or not all class queries project as SQL Stored Procedures regardless of the query's `SqlProc` setting. Default is 0. Only class queries defined with `SqlProc=1` project as Stored Procedures. When set to 1, all class queries project as stored procedures. When changing this setting, you must recompile the classes with the class queries in order for this change to have an affect. Modifying this setting in the CPF file does not require a system restart to make it active.

## Examples

```
QueryProcedures=1
```

## Range of Values

0 or 1. The default is 0.

## Management Portal

On the page **System Administration > Configuration > SQL and Object Settings > General SQL Settings**, in the **All Class Queries Project as Stored Procedures** setting, select **Yes** (to project queries as stored procedures) or **No** (the default).



# ReferentialChecks

Enable/disable validating the foreign key constraint.

```
[SQL]    ReferentialChecks=n
```

## Description

The value *n* may be 1 or 0:

- When 1 (true) Caché validates the foreign key constraint for INSERT, UPDATE, and DELETE operations.
- When 0 (false) Caché bypasses validation of foreign key constraints.

## Examples

```
ReferentialChecks=0
```

## Range of Values

0 or 1. The default is 1.

## Management Portal

On the page **System Administration > Configuration > SQL and Object Settings > General SQL Settings**, for the **Perform Referential Integrity Checks on Foreign Keys for INSERT, UPDATE, and DELETE** setting, choose Yes (true) or No (false).

# SaveMAC

Enable/disable saving the source code for cached query routines.

[SQL]      SaveMAC=*n*

## Description

The value *n* may be 1 or 0:

- When 1 (true) the source code (.MAC and .INT) for cached query routines is saved.
- When 0 (false) it is not.

## Examples

SaveMAC=0

## Range of Values

0 or 1. The default is 0.

## Management Portal

On the page **System Administration > Configuration > SQL and Object Settings > General SQL Settings**, for the **Cached query - Save Source** setting, choose Yes (true) or No (false).

# TCPKeepAlive

(Windows and Linux) Number of seconds between keep-alive messages.

```
[SQL]      TCPKeepAlive=n
```

## Description

TCP Keep-Alive Interval; the number of seconds between keep-alive messages. The default is 300 seconds (5 minutes). The setting applies only to Caché running on Windows and Linux.

You can set this system wide setting from the Caché command prompt by calling:

```
Set statusCode = $SYSTEM.SQL.SetTCPKeepAlive(<seconds>, .oldVal)
```

For a TCP device, you can also specify this interval with the OPEN or USE commands, as the eighth parameter (for example: `O tcp: ("SERVER":port:mode:::::keepalive)`, or the keyword `/KEEPALIVE=n`. The OPEN command parameters for a TCP device are:

```
hostname{:port{:mode{:terminators{:ibfsz{:obfsz{:queuesize{:keepalivetime}}}}}}}
```

For details about the [OPEN](#) and [USE](#) commands and arguments, including examples, see the [Caché ObjectScript Reference](#).

## Examples

```
TCPKeepAlive=300
```

## Range of Values

30–432,000 (5 days). The default is 300. If the value is less than 30, the setting uses 30. If the value is 0, the setting treats this as if it were unset and uses the operating system default.

## Management Portal

On the page **System Administration > Configuration > SQL and Object Settings > General SQL Settings**, enter a value for the **TCP Keep Alive Interval (in seconds)** setting.

# TimePrecision

Default time precision for SQL scalar time functions.

```
[SQL]      TimePrecision=n
```

## Description

The value *n* is the default time precision for the Time component of the value returned by the SQL Scalar functions GETDATE(), CURRENT\_TIME, and CURRENT\_TIMESTAMP. The precision is expressed as the number of decimal places allowed for the millisecond portion of the time value.

## Examples

```
TimePrecision=0
```

## Range of Values

0–9. The default is 0, which means milliseconds are not returned in the time value.

## Management Portal

On the page **System Administration > Configuration > SQL and Object Settings > General SQL Settings**, for the **Default Time Precision for GETDATE, CURRENT\_TIME, and CURRENT\_TIMESTAMP** setting, enter a number of decimal places.

---

# TODATEDefaultFormat

---

Default date format for the SQL TO\_DATE() function.

```
[SQL]      TODATEDefaultFormat=n
```

## Description

The value *n* provides the format string that the SQL TO\_DATE() function uses when TO\_DATE() is called without a format specified.

## Examples

```
TODATEDefaultFormat=DD MON YYYY
```

## Range of Values

Any string in a format appropriate for the TO\_DATE() function. The default string is:

```
DD MON YYYY
```

## Management Portal

No equivalent.



# [SqlSysDatatypes]

This topic describes the parameters found in the [SqlSysDatatypes] section of the CPF file.

# System-Defined SQL Datatypes

Map SQL datatypes to their Caché equivalents.

```
[SqlSysDatatypes]      x=a
```

## Description

The [SqlSysDatatypes] section contains system-defined datatype descriptions. Each description maps an SQL datatype to its Caché equivalent in the format  $x=a$ , as follows:

- Each keyword  $x$  is the name of the SQL datatype plus any allowed arguments.
- The value  $a$  is the Caché equivalent, including any constraints on the arguments.

## Examples

Within a Caché parameter file, a line break indicates the end of a parameter definition. The following example of a [SqlSysDatatypes] section contains line breaks that would not be valid in a Caché parameter file. You should interpret every character that you see between an equals sign (=) and the next parameter name as being on the same line in the file.

```
[SqlSysDatatypes]
BIGINT=%Library.BigInt
BIGINT(%1)=%Library.BigInt
BINARY=%Library.Binary(MAXLEN=1)
BINARY VARYING=%Library.Binary(MAXLEN=1)
BINARY VARYING(%1)=%Library.Binary(MAXLEN=1)
BINARY(%1)=%Library.Binary(MAXLEN=1)
BIT=%Library.Boolean
CHAR=%Library.String(MAXLEN=1)
CHAR VARYING=%Library.String(MAXLEN=1)
CHAR VARYING(%1)=%Library.String(MAXLEN=1)
CHAR(%1)=%Library.String(MAXLEN=1)
CHARACTER=%Library.String(MAXLEN=1)
CHARACTER VARYING=%Library.String(MAXLEN=1)
CHARACTER VARYING(%1)=%Library.String(MAXLEN=1)
CHARACTER(%1)=%Library.String(MAXLEN=1)
DATE=%Library.Date
DATETIME=%Library.TimeStamp
DEC=%Library.Numeric(MAXVAL=9999999999999999,MINVAL=-9999999999999999,SCALE=0)
DEC(%1)=%Library.Numeric(MAXVAL=<| '$$maxval^apiSQL(%1,0)' |>,MINVAL=<| '$$minval^apiSQL(%1,0)' |>,SCALE=0)
DEC(%1,%2)=%Library.Numeric(MAXVAL=<| '$$maxval^apiSQL(%1,%2)' |>,MINVAL=<| '$$minval^apiSQL(%1,%2)' |>,SCALE=%2)
DECIMAL=%Library.Numeric(MAXVAL=9999999999999999,MINVAL=-9999999999999999,SCALE=0)
DECIMAL(%1)=%Library.Numeric(MAXVAL=<| '$$maxval^apiSQL(%1,0)' |>,MINVAL=<| '$$minval^apiSQL(%1,0)' |>,SCALE=0)
DECIMAL(%1,%2)=%Library.Numeric(MAXVAL=<| '$$maxval^apiSQL(%1,%2)' |>,MINVAL=<| '$$minval^apiSQL(%1,%2)' |>,SCALE=%2)
DOUBLE=%Library.Double
DOUBLE PRECISION=%Library.Double
FLOAT=%Library.Float
FLOAT(%1)=%Library.Float(MAXVAL=<| '$$maxval^apiSQL(%1)' |>,MINVAL=<| '$$minval^apiSQL(%1)' |>)
IMAGE=%Stream.GlobalBinary
INT=%Library.Integer(MAXVAL=2147483647,MINVAL=-2147483648)
INT(%1)=%Library.Integer(MAXVAL=2147483647,MINVAL=-2147483648)
INTEGER=%Library.Integer(MAXVAL=2147483647,MINVAL=-2147483648)
LONG=%Stream.GlobalCharacter
LONG BINARY=%Stream.GlobalBinary
LONG RAW=%Stream.GlobalBinary
LONG VARCHAR=%Stream.GlobalCharacter
LONG VARCHAR(%1)=%Stream.GlobalCharacter
LONGTEXT=%Stream.GlobalCharacter
LONGVARBINARY=%Stream.GlobalBinary
LONGVARBINARY(%1)=%Stream.GlobalBinary
LONGVARCHAR=%Stream.GlobalCharacter
LONGVARCHAR(%1)=%Stream.GlobalCharacter
MEDIUMINT=%Library.Integer(MAXVAL=8388607,MINVAL=-8388608)
MEDIUMINT(%1)=%Library.Integer(MAXVAL=8388607,MINVAL=-8388608)
MONEY=%Library.Currency
NATIONAL CHAR=%Library.String(MAXLEN=1)
NATIONAL CHAR VARYING=%Library.String(MAXLEN=1)
NATIONAL CHAR VARYING(%1)=%Library.String(MAXLEN=1)
NATIONAL CHAR(%1)=%Library.String(MAXLEN=1)
NATIONAL CHARACTER=%Library.String(MAXLEN=1)
NATIONAL CHARACTER VARYING=%Library.String(MAXLEN=1)
NATIONAL CHARACTER VARYING(%1)=%Library.String(MAXLEN=1)
NATIONAL CHARACTER(%1)=%Library.String(MAXLEN=1)
NATIONAL VARCHAR=%Library.String(MAXLEN=1)
```



As described above.

On the page **System Administration > Configuration > SQL and Object Settings > System-defined DDL Mappings**, select **Edit** to modify a datatype definition.



# [SqlUserDatatypes]

This topic describes the parameters found in the [SqlUserDatatypes] section of the CPF file.

# User-Defined SQL Datatypes

---

Map SQL datatypes to their Caché equivalents.

```
[SqlUserDatatypes]      x=a
```

## Description

The [SqlUserDatatypes] section contains a user-defined datatype descriptions. Each description maps an SQL datatype to its Caché equivalent in the format  $x=a$ , as follows:

- Each keyword  $x$  is the name of the SQL datatype plus any allowed arguments.
- The value  $a$  is the Caché equivalent, including any constraints on the arguments.

## Examples

With the following line in the CPF file, when MYVARCHAR(10) is seen in a statement, the property is created with type %Library.String(MAXLEN=10,TRUNCATE=0).

```
MYVARCHAR(%1)=%Library.String(MAXLEN=%1,TRUNCATE=0)
```

## Range of Values

As described above.

## Management Portal

On the page **System Administration > Configuration > SQL and Object Settings > User-defined DDL Mappings**, select **Create New User-defined DDL Mapping** or **Edit** to modify list of mappings.

# [Startup]

This topic describes the parameters found in the [Startup] section of the CPF file.

# CallinHalt

---

Callin close.

```
[Compatibility]      CallinHalt=n
```

## Description

Indicates whether or not Caché executes the CALLIN^%ZSTOP routine entry at background process (started with the \$JOB command) logout. 1 (true) enables execution of CALLIN^%ZSTOP at Callin close; 0 (false) disables it. The default is 1.

See also [ProcessStart](#) and [ProcessHalt](#), [JobStart](#) and [JobHalt](#), and [SystemStart](#) and [SystemHalt](#).

## Examples

```
CallinStart=1
```

## Range of Values

0 or 1

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Startup**, in the [CallinHalt](#) row, select **Edit**. Choose true or false.

---

# CallinStart

---

Callin initialization.

```
[Compatibility]      CallinStart=n
```

## Description

Indicates whether or not Caché executes the CALLIN^%ZSTART routine entry at background process (started with the \$JOB command) login. 1 (true) enables execution of CALLIN^%ZSTART at Callin initialization; 0 (false) disables it. The default is 1

See also [ProcessStart](#) and [ProcessHalt](#), [JobStart](#) and [JobHalt](#), and [SystemStart](#) and [SystemHalt](#).

## Examples

```
CallinStart=1
```

## Range of Values

0 or 1

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Startup**, in the [CallinStart](#) row, select **Edit**. Choose true or false.

# CliSysName

---

Node name for the local system.

```
[Startup]      Name=n
```

## Description

The value *n* is the node name for this computer, and has many uses:

- The node name to be sent to the ECP network server, so that the server can identify the client
- The node name for a unique [\\$JOB](#) value. This is useful when using \$JOB to index globals accessed by more than one networked system.
- Certain forms of the [\\$SYSTEM](#) function return this node name concatenated with the Caché instance name, as *nodename:instancename*. This concatenated string is recorded in Audit files.

## Examples

Name=

## Range of Values

The default is blank. If no name is provided, Caché reads the computer settings and uses the computer “host name” as the client node name. This can be up to 64 characters long.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Startup**, in the [CliSysName](#) row, select **Edit**. Enter a node name.



---

# DBSizesAllowed

---

Database block sizes allowed in the database creation screen.

```
[Startup]      DBSizesAllowed=n[n,n...]
```

## Description

Comma separated list of allowed database block sizes – 4096, 8192, 16384, 32768, 65536 – that you can enter in the database creation screen.

## Examples

```
DBSizesAllowed=8192
```

```
DBSizesAllowed=8192,16384
```

## Range of Values

4096, 8192, 16384, 32768, 65536. Default is 8192.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Startup**, in the **DBSizesAllowed** row, select **Edit**.

## DefaultPort

---

Port number for the Caché Superserver.

```
[Startup]      DefaultPort=n
```

### Description

The value *n* is the port number for the Caché superserver (the Caché server that listens on a specified port (1972 by default) for incoming connections to Caché and dispatches them to the appropriate subsystem).

### Examples

```
DefaultPort=1972
```

```
DefaultPort=56773
```

### Range of Values

Valid IP port number. A standard Caché installation sets the superserver port number to 1972, or if that port is in use by another Caché instance on the same system, the next unused port number greater than or equal to 56773.

### Management Portal

On the page **System Administration > Configuration > System Configuration > Memory and Startup**, enter a number in the **Superserver Port Number** field.

# DefaultPortBindAddress

IP address for the Caché Superserver to bind to.

```
[Startup]      DefaultPortBindAddress=nnn.nnn.nn.nn (ip address)
```

## Description

Specify an IP address for the SuperServer to bind to. Set to one of the host system's IP addresses. The SuperServer binds to that address. Requests to the SuperServer port on other IP addresses on the host are not accepted. This makes it possible to limit connections to the SuperServer to a single address on a multihomed host. The SuperServer is the process that accepts client connections for Cache Direct, ODBC, JDBC, Cache Studio, and other connection technologies. If this property is not set, the SuperServer accepts requests on all IP addresses on the host. The default is to accept on all addresses.

## Examples

```
DefaultPortBindAddress=196.200.34.56
```

## Range of Values

Valid IP address.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Startup**, in the **DefaultPortBindAddress** row, select **Edit**.

# EnsembleAutoStart

---

Start the Ensemble productions when Caché starts.

```
[Startup]      EnsembleAutoStart=n
```

## Description

Start the Ensemble productions when Caché starts. Default = 1. This entry is ignored if this not an Ensemble system.

## Examples

```
EnsembleAutoStart=1
```

## Range of Values

0 or 1.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Startup**, in the **EnsembleAutoStart** row, select **Edit**.

# ErrorPurge

Number of days to keep error globals.

```
[Startup]      ErrorPurge=n
```

## Description

The value  $n$  is the number of days of errors which were logged by the error handler %ETN are retained Errors older than this are deleted on the next Caché restart.

## Examples

```
ErrorPurge=30
```

## Range of Values

1–1000 days. The default is 30 days.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Startup**, in the [ErrorPurge](#) row, select **Edit**. Enter a number of days.

# FIPSMoDe

---

Enable/disable FIPS 140–2 compliant library for database encryption on Red Hat Linux.

```
[Startup]      FIPSMoDe=n
```

## Description

Enable/disable FIPS 140–2 compliant library for database encryption on Red Hat Enterprise Linux 6.6 (or later minor version) and Red Hat Enterprise Linux 7.1 (or later minor version) for x86-64.

The value *n* is either 1 or 0.

- When FIPSMoDe=1 (true), Caché uses the FIPS 140–2 compliant library for database encryption.
- When FIPSMoDe=0 (false), Caché does not use the FIPS 140–2 compliant library for database encryption. Default is 0.

See the article [FIPS 140–2 Compliance for Caché Database Encryption](#) for details.

## Example

```
FIPSMoDe=0
```

## Range of Values

0 or 1. The default is 0 (false, disabled).

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Startup**, in the [FIPSMoDe](#) row, select **Edit**. Select the checkbox to use the FIPS 140-2 compliant library for database encryption.

# IPv6

Enable/disable IPv6 network with IPv6 addresses.

```
[Startup]      IPv6=n
```

## Description

This entry controls whether your system is operating in an IPv6 (Internet Protocol Version 6) network with IPv6 addresses. Default=0.

## Examples

```
IPv6=1
```

## Range of Values

0 or 1; the default is 0.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Startup**, in the [IPv6](#) row, select **Edit**. Enter 1 to enable or 0 to disable.

# JobHalt

---

Background process (job) shutdown.

```
[Compatibility]      JobHalt=n
```

## Description

Indicates whether or not Caché executes the `JOB^%ZSTOP` routine entry at background process (started with the `$JOB` command) logout. 1 (true) enables execution of `JOB^%ZSTOP` at background process logout; 0 (false) disables it. The default is 1.

[ProcessStart](#) and [ProcessHalt](#) govern *foreground* processes. These are processes that are started via a terminal session or are specifically set to run in the foreground.

By contrast, [JobStart](#) and `JobHalt` govern *background* processes include any processes that are started via the `JOB` command, plus any background server processes including CSP, Cache Direct, ODBC, or any of the objects bindings.

Also see [SystemStart](#) and [SystemHalt](#).

## Examples

```
JobHalt=1
```

## Range of Values

0 or 1.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Startup**, in the [JobHalt](#) row, select **Edit**. Choose true or false.



# JobServers

Number of job servers.

```
[Startup]      JobServers=n
```

## Description

The value *n* is the number of job servers you want Caché to start up.

Having a large number of job servers running will use more memory and processes, but allows for much faster jobbing of processes because Caché doesn't have to start the processes at the system level and then initialize them.

Job servers are best used when the application creates a significant number of short-lived processes via the Job command. For this type of process where operating system process creation overhead dominates the total cost of running the process, using job servers can be beneficial. If background processes tend to perform extended tasks then there is very little benefit from using job servers.

## Examples

```
JobServers=0
```

## Range of Values

0–2000 job servers. The default is 0.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Startup**, in the [JobServers](#) row, select **Edit**. Enter a number of job servers.

# JobStart

---

Background process (job) startup.

```
[Compatibility]      JobStart=n
```

## Description

Indicates whether or not Caché executes the JOB^%ZSTART routine entry at background process (started with the \$JOB command) login. 1 (true) enables execution of JOB^%ZSTART at background process login; 0 (false) disables it. The default is 1.

[ProcessStart](#) and [ProcessHalt](#) govern *foreground* processes. These are processes that are started via a terminal session or are specifically set to run in the foreground.

By contrast, [JobStart](#) and [JobHalt](#) govern *background* processes include any processes that are started via the JOB command, plus any background server processes including CSP, Cache Direct, ODBC, or any of the objects bindings.

Also see [SystemStart](#) and [SystemHalt](#).

## Examples

```
JobStart=1
```

## Range of Values

0 or 1.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Startup**, in the [JobStart](#) row, select **Edit**. Choose true or false.

---

# MaxCacheTempSizeAtStart

---

Maximum size of database on restart.

```
[Startup]      MaxCacheTempSizeAtStart=n
```

## Description

Maximum size in megabytes of the CacheTemp database when the system is restarted. When the system restarts, the CacheTemp database is truncated to this size. If 0, the CacheTemp database is not truncated.

## Examples

```
MaxCacheTempSizeAtStart=500000
```

## Range of Values

1–1000000 (1 million) megabytes. The default is 0 megabytes (the CacheTemp database is not truncated).

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Startup**, in the [MaxCacheTempSizeAtStart](#) row, select **Edit**. Enter a number of megabytes.

# MaxConsoleLogSize

---

Maximum size of the console log file.

```
[Startup]      MaxConsoleLogSize=n
```

## Description

The value *n* is the maximum size of the Caché console file, in megabytes. The console file is `cconsole.log` in the Caché system management directory, or the alternate `filename.log` named by the second piece of the `console` parameter in the `[config]` section.

After the console file reaches the size limit of *n* megabytes, the next Caché startup renames the existing `filename.log` to `filename.old_Date` and creates a new, empty Caché console file called `filename.log`.

## Examples

```
MaxConsoleLogSize=5
```

```
MaxConsoleLogSize=8
```

## Range of Values

1–500 megabytes. The default is 5 megabytes.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Startup**, in the [MaxConsoleLogSize](#) row, select **Edit**. Enter a number of megabytes.

# ProcessHalt

Foreground shutdown.

```
[Compatibility]      ProcessHalt=n
```

## Description

Indicates whether or not Caché executes the LOGIN^%ZSTOP routine entry at foreground process logout. 1 (true) enables execution of LOGIN^%ZSTOP at foreground process logout; 0 (false) disables it. The default is 1.

[ProcessStart](#) and [ProcessHalt](#) govern *foreground* processes. These are processes that are started via a terminal session or are specifically set to run in the foreground.

By contrast, [JobStart](#) and [JobHalt](#) govern *background* processes include any processes that are started via the JOB command, plus any background server processes including CSP, Cache Direct, ODBC, or any of the objects bindings.

Also see [SystemStart](#) and [SystemHalt](#).

## Examples

```
ProcessHalt=1
```

## Range of Values

0 or 1.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Startup**, in the [ProcessHalt](#) row, select **Edit**. Choose true or false.

# ProcessStart

---

Foreground startup.

```
[Compatibility]    ProcessStart=n
```

## Description

Indicates whether or not Caché executes the LOGIN^%ZSTART routine entry at foreground process login. 1 (true) enables execution of LOGIN^%ZSTART at foreground process login; 0 (false) disables it. The default is 1.

ProcessStart and [ProcessHalt](#) govern *foreground* processes. These are processes that are started via a terminal session or are specifically set to run in the foreground.

By contrast, [JobStart](#) and [JobHalt](#) govern *background* processes include any processes that are started via the JOB command, plus any background server processes including CSP, Cache Direct, ODBC, or any of the objects bindings.

Also see [SystemStart](#) and [SystemHalt](#).

## Examples

```
ProcessStart=1
```

## Range of Values

0 or 1.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Startup**, in the [ProcessStart](#) row, select **Edit**. Choose true or false.

---

# ShutdownTimeout

---

Number of seconds Caché should wait for shutdown to complete.

```
[Startup] ShutdownTimeout=n
```

## Description

The value  $n$  is the number of seconds Caché should wait for shutdown to complete normally before timing out and forcing a shutdown.

## Examples

```
ShutdownTimeout=300
```

## Range of Values

From 120 to a maximum of 100,000 seconds. The default is 300 seconds (5 minutes).

## Management Portal

On the page **System Administration** > **Configuration** > **Additional Settings** > **Startup**, in the [ShutdownTimeout](#) row, select **Edit**. Enter a number of seconds.

# SystemHalt

---

System shutdown.

```
[Compatibility]    SystemHalt=n
```

## Description

Indicates whether or not Caché executes the SYSTEM^%ZSTOP routine at system shutdown. 1 (true) enables execution of SYSTEM^%ZSTOP at system shutdown; 0 (false) disables it. The default is 1.

## Examples

```
SystemHalt=1
```

## Range of Values

0 or 1

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Startup**, in the [SystemHalt](#) row, select **Edit**. Choose true or false.



---

# SystemStart

---

System startup.

```
[Compatibility]    SystemStart=n
```

## Description

Indicates whether or not Caché executes the SYSTEM^%ZSTART routine at system startup. 1 (true) enables execution of SYSTEM^%ZSTART at system startup; 0 (false) disables it. The default is 1.

## Examples

```
SystemStart=1
```

## Range of Values

0 or 1

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Startup**, in the [SystemStart](#) row, select **Edit**. Choose true or false.

# TempDirectory

---

Subdirectory for temporary files.

```
[Startup]    TempDirectory=n
```

## Description

The value *n* is the name of a subdirectory in which to store temporary files for Caché. The string can be a relative pathname which would be created under the Caché manager directory, or a full directory specification.

The default value, Temp, identifies the following location under *install-dir*, the Caché installation directory:

*install-dir*\Mgr\Temp

When you set a new TempDirectory value, the system creates a subdirectory of this name under the Mgr subdirectory in the Caché installation directory, as shown for Temp in the previous example. This new directory becomes the Caché temporary directory.

## Examples

To create c:\intersystems\cache\mgr\temp\ on windows

```
TempDirectory=Temp
```

## Range of Values

The default string is Temp

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Startup**, in the [TempDirectory](#) row, select **Edit**. Enter a subdirectory name.

# TerminalPrompt

Sets the format of the terminal prompt.

```
[Startup]      TerminalPrompt=n
```

## Description

This is a comma separated string of values which set the default terminal prompt for the system. The default is 8, 2

Values: The order of the values in the string determines the order the values appear in the prompt. For example

```
TerminalPrompt="2,1"
```

gives you a terminal prompt of %SYS:HostName>

- 0 - Use only ">" for the prompt.
- 1 - Host name, also known as the current system name. The name assigned to your computer. For example, LABLAPTOP>. This is the same for all of your terminal processes.
- 2 - Namespace name. For example, %SYS>. The current namespace name is contained in the \$NAMESPACE special variable. It can be an explicit namespace name or an implied namespace name.
- 3 - Config name. The name of your system installation. For example, CACHE2>. This is the same for all of your terminal processes.
- 4 - Current time, expressed as local time in 24-hour format with whole seconds. For example, 15:59:36>. This is the static time value for when the prompt was returned. This value changes for each prompt.
- 5 - pid. The Process ID for your terminal. For example, 2336>. This is different for each terminal process. This value can also be returned from the \$JOB special variable.
- 6 - Username. For example, fred>. This is the same for all of your terminal processes.
- 7 - Elapsed time executing the last command, in seconds.milliseconds. For example, .000495>. Leading and trailing zeros are suppressed. This changes for each prompt.
- 8 - Transaction Level. For example, TL1>.

## Examples

```
TerminalPrompt=0
```

This gives you a right-angle bracket as a prompt.

## Range of Values

0–8, as described above.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Startup**, in the [TerminalPrompt](#) row, select **Edit**. Enter a comma-separated string of values.

# WebServer

---

Start or don't start the Web server.

```
[Compatibility]    WebServer=n
```

## Description

This parameter indicates whether to start the Web server. The Web server is necessary to enable Caché Server Pages (CSP) applications such as the Management Portal.

Also see [WebServerURLPrefix](#), [WebServerPort](#), and [WebServerName](#).

## Examples

WebServer=1

## Range of Values

0 or 1

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Startup**, in the [WebServer](#) row, select **Edit**. Choose true (selected) or false (cleared).

---

# WebServerName

---

Sets an IP address or a DNS name on the Web server.

```
[Compatibility]      WebServerName=nnn.nnn.nn.nn(ip address) or www.DNS name.com
```

## Description

The DNS name or IP address of the Web Server that is configured for use with Caché tools. For example if Studio is connected to this Caché system, and if the user requests to run a CSP template from Studio, WebServerName identifies the DNS name or IP address to which this request will be sent. The Web server is necessary to enable Caché Server Pages (CSP) applications such as the Management Portal.

Also see [WebServer](#) , [WebServerPort](#),[WebServerURLPrefix](#).

## Examples

```
WebServer=196.200.34.56
```

## Range of Values

A valid IP address.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Startup**, in the [WebServerName](#) row, select **Edit**. Enter a name.

# WebServerPort

---

Set the Web server port.

```
[Compatibility]      WebServerPort=nnnnn
```

## Description

The port number to use for the Web server. A standard Caché installation sets the Web server port number to the first unused port number greater than or equal to 57772.

Also see [WebServer](#), [WebServerName](#), [WebServerURLPrefix](#).

## Examples

```
WebServerPort=57772
```

## Range of Values

57772 and above.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Startup**, in the [WebServerPort](#) row, select **Edit**. Enter a port number.

# WebServerURLPrefix

Identify the Caché instance name so that a remote Web server can access it.

```
[Compatibility]      WebServerURLPrefix=n
```

## Description

A Caché installation is called an instance. Each instance has a name — the name you assigned to Caché when you installed it.

If you are using a remote Web server to access one or more Caché instances, then for each instance you must provide a `WebServerURLPrefix` parameter in the [Startup] section of the Caché parameter file, as shown above. The value *n* must be the name of the Caché instance.

This is only one of the steps required to set up a remote Web server to access one or more Caché instances. For details, see the “[Connecting to Remote Servers](#)” chapter in the *Caché System Administration Guide*.

## Examples

```
WebServerURLPrefix=CACHE3
```

## Range of Values

The default is blank (no prefix).

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Startup**, in the [WebServerURLPrefix](#) row, select **Edit**. Enter a Caché instance name.

# ZSTU

---

User-defined startup.

```
[Startup]      ZSTU=n
```

## Description

The value *n* may be 1 or 0:

- When 1 (true) Caché runs the ^ZSTU user-defined startup routine.
- When 0 (false) it does not.

## Examples

ZSTU=1

## Range of Values

0 or 1. The default is 1.

## Management Portal

On the page **System Administration > Configuration > Additional Settings > Startup**, in the [ZSTU](#) row, select **Edit**. Choose true or false.



# [Telnet]

This topic describes the parameters found in the [Telnet] section of the Caché parameter file. This section is valid for Windows systems only.

# DNSLookup

(Windows) Enable/disable DNS lookup of the Telnet client address.

```
[Telnet]    DNSLookup=n
```

## Description

This setting enables or disables DNS lookup of the client address in the telnet daemon before passing the address to the Caché process that was created to service the connection. This determines the format of the client address returned by \$IO and \$ZIO in the Cache process.

The value *n* may be ON or OFF:

- When ON (true) a DNS lookup of the client address is performed, and the client name is passed to Caché.
- When OFF (false), no DNS lookup is performed, and the client address is provided in either dotted decimal format (if the connection was via IPV4) or in the colon separated hexadecimal format (if the connection was via IPV6).

DNS lookup should be disabled if a DNS server is not available to do the lookup, because a long delay will occur during login if the DNS server is not available.

Caché Telnet settings apply only to Windows configurations in which InterSystems supplies the Telnet servers.

## Examples

```
DNSLookup=ON
```

## Range of Values

ON or OFF. The default is ON.

## Management Portal

On the page **System Administration > Configuration > Device Settings > Telnet Settings**, select the **Telnet** category, In the [DNSLookup](#) row, select **Edit**. Choose **ON** or **OFF**.

---

# Port

---

Telnet port number.

```
[Telnet]      Port=n
```

## Description

The value  $n$  is the TCP/IP port number for Telnet connections. If multiple Caché configurations are to run on the same host, a different Telnet port number must be specified for each configuration. Clients can attach to configurations using the non-default port number by specifying the port number when they invoke Telnet on the client system.

Caché Telnet settings apply only to Windows configurations in which InterSystems supplies the Telnet servers.

## Examples

```
Port=23
```

## Range of Values

Valid TCP/IP port number. The default is 23.

## Management Portal

On the page **System Administration > Configuration > Device Settings > Telnet Settings**, select the **Telnet** category. In the [Telnet Port Number](#) box, enter a TCP/IP port number.

