



# Using the \$ZF Function Calls for DSM

Version 2007.1

04 June 2007

*Using the \$ZF Function Calls for DSM*  
Caché Version 2007.1 04 June 2007  
Copyright © 2007 InterSystems Corporation  
All rights reserved.

This book was assembled and formatted in Adobe Page Description Format (PDF) using tools and information from the following sources: Sun Microsystems, RenderX, Inc., Adobe Systems, and the World Wide Web Consortium at [www.w3c.org](http://www.w3c.org). The primary document development tools were special-purpose XML-processing applications built by InterSystems using Caché and Java.



Caché WEBLINK, Distributed Cache Protocol, M/SQL, N/NET, and M/PACT are registered trademarks of InterSystems Corporation.



InterSystems Jalapeño Technology, Enterprise Cache Protocol, ECP, and InterSystems Zen are 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 Customer Support**

Tel: +1 617 621-0700  
Fax: +1 617 374-9391  
Email: [support@InterSystems.com](mailto:support@InterSystems.com)

# Table of Contents

<b>Using the \$ZF Function Calls for DSM.....</b>	<b>1</b>
GETJPI Intrinsic Function .....	2
GETDVI Intrinsic Function .....	3
GETSYI Intrinsic Function .....	4
SETSYM Intrinsic Function .....	5
GETSYM Intrinsic Function .....	6
DELSYM Intrinsic Function .....	6
CRELOG Intrinsic Function .....	7
TRNLNM Intrinsic Function .....	8
DELLOG Intrinsic Function .....	9
GETUAI Intrinsic Function .....	9
GETMSG Intrinsic Function .....	10
SETPRN Intrinsic Function .....	10
SETPRI Intrinsic Function .....	11
OPCOM Intrinsic Function .....	11
MOUNT Intrinsic Function .....	12
DISMOUNT Intrinsic Function .....	12
DIRECTORY Intrinsic Function .....	13
PARSE Intrinsic Function .....	13
GETFILE Intrinsic Function .....	14



# Using the \$ZF Function Calls for DSM

This document describes Caché-supplied (intrinsic) functions that are supplied as arguments to the \$ZF function for DSM compatibility and conversion.

This article describes the following intrinsic functions:

- [GETJPI](#) job and process information
- [GETDVI](#) device characteristics
- [GETSYI](#) system information
- [SETSYM](#) sets DCL symbol value
- [GETSYM](#) returns DCL symbol value
- [DELSYM](#) deletes DCL symbol value
- [CRELOG](#) creates logical name
- [TRNLNM](#) translates logical name
- [DELLOG](#) deletes logical name
- [GETUAI](#) returns account authorization parameters
- [GETMSG](#) returns status code message text
- [SETPRN](#) sets name of the calling process
- [SETPRI](#) sets base priority for a process
- [OPCOM](#) sends message to the operator
- [MOUNT](#) mounts a device
- [DISMOUNT](#) dismounts a device
- [DIRECTORY](#) returns the default directory
- [PARSE](#) parses a file name
- [GETFILE](#) returns file information

For more information on ObjectScript functions generally, see the section [Functions](#) in the *Caché ObjectScript Guide*. For more information on the \$ZF function, refer to [\\$ZF](#) in the *Caché ObjectScript Reference*.

The function names are shown here in all uppercase letters, but they are, in fact, case-insensitive.

## GETJPI Intrinsic Function

---

Gets job and process information.

```
SET x = $ZF("GETJPI",pid,item,procname)
```

### Parameters

<i>pid</i>	The process ID (in decimal).
<i>item</i>	An item code for the process information to be returned.
<i>procname</i>	Process name (instead of pid).

### Description

This function returns job and process information.

### Parameters

#### *pid*

The process ID (in decimal).

#### *item*

Item Codes for GETJPI: "ACCOUNT" "APTCNT" "ASTACT" "ASTCNT" "ASTEN" "ASTLM" "AUTHPRI" "AUTHPRIV" "BIOCNT" "BIOLM" "BUFIO" "BYTCNT" "BYTLM" "CLINAME" "CPU\_ID" "CPULIM" "CPUTIM" "CREPRC\_FLAGS" "CURPRIV" "DFPFC" "DFWSCNT" "DIOCNT" "DIOLM" "DIRIO" "EFCS" "EFCU" "EFWM" "ENQCNT" "ENQLM" "EXCVEC" "FAST\_VP\_SWITCH" "FILCNT" "FILLM" "FINALEXC" "FREPOVA" "FREPIVA" "FREPTCNT" "GPGCNT" "GRP" "IMAGECOUNT" "IMAGNAME" "IMAGPRIV" "JOBPRCCNT" "JOBTYPE" "LOGINTIM" "MASTER\_PID" "MAXDETACH" "MAXJOBS" "MEM" "MODE" "MSGMASK" "NODENAME" "NODE\_CSID" "NOD\_VERSION" "OWNER" "PAGEFLTS" "PAGFILCNT" "PAGFILLOC" "PGFLQUOTA" "PHDFLAGS" "PID" "PPGCNT" "PRCCNT" "PRCLM" "PRCNAM" "PRI" "PRIB" "PROC\_INDEX" "PROCPRIV" "SCHED\_POLICY" "SHRFILLM" "SITESPEC" "SLOW\_VP\_SWITCH" "STATE" "STS" "STS2" "SWPFILLOC" "TABLENAME" "TERMINAL" "TMBU" "TQCNT" "TQLM" "TT\_ACCPORNAM" "TT\_PHYDEVNAM" "UAF\_FLAGS" "UIC"

"USERNAME" "VIRTPEAK" "VOLUMES" "VP\_CONSUMER" "VP\_CPUTIM" "WSAUTH"  
 "WSAUTHEXT" "WSEXTENT" "WSPEAK" "WSQUOTA" "WSSIZE"

### *procname*

The process name.

## GETDVI Intrinsic Function

Gets device information.

```
SET x = $ZF("GETDVI",devname,item,flag)
```

### *Parameters*

<i>devname</i>	A physical device name, or a logical name that equates to a physical device name.
<i>item</i>	An item code for the device information to be returned.
<i>flag</i>	1 = primary characteristics. 2 = secondary characteristics. The default is 2.

### Description

This function returns device characteristics.

### Parameters

#### *devname*

A physical device name, or a logical name that equates to a physical device name.

#### *item*

Item Codes for GETDVI "ACPTYPE" "ALL" "ALLDEVNAM" "ALLOCLASS"  
 "ALT\_HOST\_AVAIL" "ALT\_HOST\_NAME" "ALT\_HOST\_TYPE" "AVL" "CCL" "CLUSTER"  
 "CONCEALED" "CYLINDERS" "DEVBUFSIZ" "DEVCHAR" "DEVCHAR2" "DEVCLASS"  
 "DEVDEPEND" "DEVDEPEND2" "DEVLOCKNAM" "DEVNAM" "DEVSTS" "DEVTYPE" "DIR"  
 "DMT" "DUA" "ELG" "ERRCNT" "EXISTS" "FOD" "FOR" "FREEBLOCKS" "FULLDEVNAM"  
 "GEN" "HOST\_AVAIL" "HOST\_COUNT" "HOST\_NAME" "HOST\_TYPE" "IDV" "LOCKID"  
 "LOGVOLNAM" "MAXBLOCK" "MAXFILES" "MBX" "MEDIA\_ID" "MEDIA\_NAME"  
 "MEDIA\_TYPE" "MNT" "MOUNTCNT" "NET" "NEXTDEVNAM" "ODV" "OPCNT" "OPR"  
 "OWNUIC" "PID" "RCK" "RCT" "REC" "RECSIZ" "REFCNT" "REMOTE\_DEVICE" "RND"  
 "ROOTDEVNAM" "RTM" "SDI" "SECTORS" "SERIALNUM" "SERVED\_DEVICE" "SHR" "SPL"

"SPLDEVNAM" "SQD" "STS" "SWL" "TRACKS" "TRANSCNT" "TRM" "TT\_ACCPORNAM"  
 "TT\_ALTYPEAHD" "TT\_ANSICRT" "TT\_APP\_KEYPAD" "TT\_AUTOBAUD" "TT\_AVO"  
 "TT\_BLOCK" "TT\_BRDCSTMBX" "TT\_CRFILL" "TT\_DECCRT" "TT\_DECCRT2" "TT\_DIALUP"  
 "TT\_DISCONNECT" "TT\_DMA" "TT\_DRCS" "TT\_EDIT" "TT\_EDITING" "TT\_EIGHTBIT"  
 "TT\_ESCAPE" "TT\_FALLBACK" "TT\_HALFDUP" "TT\_HANGUP" "TT\_HOSTSYNC"  
 "TT\_INSERT" "TT\_LFFILL" "TT\_LOCALECHO" "TT\_LOWER" "TT\_MBXDSABL"  
 "TT\_MECHFORM" "TT\_MECHTAB" "TT\_MODEM" "TT\_MODHANGUP" "TT\_NOBRDCST"  
 "TT\_NOECHO" "TT\_NOTYPEAHD" "TT\_OPER" "TT\_PAGE" "TT\_PASTHRU" "TT\_PHYDEV-  
 NAM" "TT\_PRINTER" "TT\_READSYNC" "TT\_REGIS" "TT\_REMOTE" "TT\_SCOPE"  
 "TT\_SECURE" "TT\_SETSPEED" "TT\_SIXEL" "TT\_SYSPWD" "TT\_TTSYNC" "TT\_WRAP"  
 "UNIT" "VOLCOUNT" "VOLNAM" "VOLNUMBER" "VOLSETMEM" "VPROT" "WCK"

**flag**

1 = primary characteristics. 2 = secondary characteristics. The default is 2.

## GETSYI Intrinsic Function

Gets system information.

```
SET x = $ZF("GETSYI",item,csid,nodename)
```

**Parameters**

<i>item</i>	An item keyword code for the system parameter value to be returned, specified as a quoted string.
<i>csid</i>	Node CSID (cluster node ID) parameter for the cluster member from which to obtain information. Specify 0 (as a placeholder) when specifying the optional <i>nodename</i> parameter. Specify -1 to search all cluster nodes.
<i>nodename</i>	<i>Optional</i> — The name of the cluster node, specified as a quoted string. You can specify the cluster node by <i>nodename</i> or by <i>csid</i> ; when specifying the cluster node by <i>nodename</i> , specify a value of 0 for <i>csid</i> .

**Description**

This function returns system information.

## Parameters

### *item*

Item Codes for GETSYI: "ACTIVECPU\_CNT" "ARCHFLAG" "ARCH\_NAME" "ARCH\_TYPE" "AVAILCPU\_CNT" "BALSETCNT" "BJOBLIM" "BOOTTIME" "CHARACTER\_EMULATED" "CLUSTER\_EVOTES" "CLUSTER\_FSYSID" "CLUSTER\_FTIME" "CLUSTER\_MEMBER" "CLUSTER\_NODES" "CLUSTER\_QUORUM" "CPU" "CONTIG\_GBLPAGES" "CPUTYPE" "DECIMAL\_EMULATED" "D\_FLOAT\_EMULATED" "HW\_MODEL" "HW\_NAME" "FREE\_GBLPAGES" "FREE\_GBLSECTS" "GBLPAGES" "GBLPAGFIL" "GBLSECTIONS" "LOCKIDTBL" "LOCKIDTBL\_MAX" "MAXPROCESSCNT" "MAXSYSGROUP" "NODENAME" "NODE\_CSID" "PAGEFILE\_FREE" "PAGEFILE\_PAGE" "PQL\_DWSDEFAULT" "PQL\_DWSEXTENT" "PQL\_DWSQUOTA" "PQL\_MWSDEFAULT" "PQL\_MWSEXTENT" "PQL\_MWSQUOTA" "PQL\_MASTLM" "PQL\_MBYTLM" "PQL\_MDIOLM" "PQL\_MFILLM" "PQL\_MENQLM" "SCSNODE" "SPTREQ" "SRPCOUNTV" "SWAPFILE\_FREE" "SWAPFILE\_PAGE" "VERSION" "VIRTUALPAGECNT" "WSMAX"

### *csid*

The cluster node ID. You can specify a specific cluster node, or search through all cluster nodes.

To search all cluster nodes, specify a *csid* value of -1. Each subsequent call using a *csid* of -1 returns the *item* keyword information for the next node in the cluster. Once all nodes have been sequentially gone through, a call with a *csid* of -1 returns an empty string (""). Specifying any *csid* value other than -1 terminates this sequential node access.

## SETSYM Intrinsic Function

Sets DCL symbol.

```
SET x = $ZF("SETSYM",symbol,value,flag)
```

### Parameters

<i>symbol</i>	The DCL symbol to set.
<i>value</i>	The value for the DCL symbol.
<i>flag</i>	1 = local. 2 = global. The default is 1.

### Description

This function sets a DCL symbol to a value. This DCL symbol can be local or global. SETSYM returns the null string.

# GETSYM Intrinsic Function

---

Gets a DCL symbol value.

```
SET x = $ZF("GETSYM",symbol)
```

## Parameters

<i>symbol</i>	The DCL symbol to return the value of.
---------------	--

## Description

This function returns the value of a specified DCL symbol. This DCL symbol can be local or global. DCL values are set using the SETSYM function.

# DELSYM Intrinsic Function

---

Deletes a DCL symbol.

```
SET x = $ZF("DELSYM",symbol,flag)
```

## Parameters

<i>symbol</i>	The DCL symbol to delete.
<i>flag</i>	1 = local. 2 = global. The default is 1.

## Description

This function deletes the value of a specified DCL symbol. This DCL symbol can be local or global. DCL values are set using the SETSYM function.

# CRELOG Intrinsic Function

Creates a logical name.

```
SET x = $ZF("CRELOG", logname, value, type, table)
```

## Parameters

<i>logname</i>	The logical name to create.
<i>value</i>	An equivalence string for the logical name.
<i>type</i>	<i>Optional</i> — Table/mode to define. Available values are described below. The default is USER mode.
<i>table</i>	<i>Optional</i> — The name of a user-defined logical name table. A <i>table</i> value is required if <i>type</i> =TABLE. The default is the process logical name table.

## Description

This function creates a logical name in the specified logical name table with the specified mode. Defaults are *type*=USER mode and *table*=process logical name table. CRELOG returns the null string.

The available *type* values are:

- USER = process table / user mode.
- SUPERVISOR or PROCESS = process table / supervisor mode.
- JOB = job table — LNM\$JOB.
- GROUP = group table — LNM\$GROUP.
- SYSTEM = system table — LNM\$SYSTEM.
- TABLE = user-defined logical name table.

# TRNLNM Intrinsic Function

Translates a logical name.

```
SET x = $ZF("TRNLNM", logname, table, index, acmode, trans, item)
```

## Parameters

logname	The logical name to translate.
table	The name of a user-defined logical name table. The default is the process logical name table.
index	Index of logical name if more than one translation.
acmode	Access mode. Available values are USER, SUPERVISOR, EXECUTIVE, and KERNEL.
trans	Type of case translation. Available values are CASE_BLIND and CASE_SENSITIVE.
item	The type of information about the logical name to return.

## Description

This function translates a logical name and returns the equivalence name string or the requested attributes. TRNLNM returns the null string if no match is found.

## Parameters

### *item*

Item codes for TRNLNM: "ACCESS\_MODE" "CONCEALED" "CONFINE" "CRELOG" "LENGTH" "MAX\_INDEX" "NO\_ALIAS" "TABLE" "TABLE\_NAME" "TERMINAL" "VALUE"

# DELLOG Intrinsic Function

Deletes logical name.

```
SET x = $ZF("DELLOG", logname, type, table)
```

## Parameters

<i>logname</i>	The logical name to delete.
<i>type</i>	Type of logical name.
<i>table</i>	The name of a user-defined logical name table. The default is the process logical name table.

## Description

This function deletes a logical name in the specified logical name table.

# GETUAI Intrinsic Function

Get user account information.

```
SET x = $ZF("GETUAI", account, item)
```

## Parameters

<i>account</i>	An account name.
<i>item</i>	An item code.

## Description

This function returns values for authorization parameters of a specific OpenVMS account.

## Parameters

### *item*

Item Codes for GETUAI: "ACCOUNT" "ASTLM" "BIOLM" "BYTLM" "DEFDEV" "DEFDIR" "DEF\_PRIV" "DFWSCNT" "DIOLM" "ENQLM" "FILLM" "FLAGS" "JTQUOTA" "LASTLOGIN\_I" "LASTLOGIN\_N" "LGICMD" "MAXACCTJOBS" "MAXDETACH" "MAXJOBS" "OWNER" "PBYTLM" "PGFLQUOTA" "PRCCNT" "PRI" "PRIV" "QUEPRI" "SHRFILLM" "TQCNT" "UIC" "USERNAME" "WSEXTENT" "WSQUOTA"

## Examples

The following example shows **GETUAI** used as a function:

```
SET x = $ZF( "GETUAI" , "FRED" , "USERNAME" )
```

## GETMSG Intrinsic Function

---

Returns error message text.

```
SET x = $ZF( "GETMSG" , code )
```

### Parameters

<i>code</i>	A status code for an error.
-------------	-----------------------------

### Description

This function returns the message text that corresponds to the specified status code.

## SETPRN Intrinsic Function

---

Sets name of the calling process.

```
SET x = $ZF( "SETPRN" , pname )
```

### Parameters

<i>pname</i>	A valid OpenVMS process name.
--------------	-------------------------------

### Description

This function sets the name of the calling process to the specified OpenVMS process name.

## SETPRI Intrinsic Function

Set the base priority for the process.

```
SET x = $ZF("SETPRI",pri,pid)
```

### Parameters

<i>pri</i>	The base priority of a process, specified as an integer from 0 to 15 (inclusive).
<i>pid</i>	The process ID (in decimal).

### Description

This function sets the base priority of the specified process. The process must be running CACHE image.

## OPCOM Intrinsic Function

Sends a message to the operator.

```
SET x = $ZF("OPCOM",msg)
```

### Parameters

<i>msg</i>	A message string to send to the operator terminal(s).
------------	---

### Description

This function sends the specified message to the operator terminal(s). It returns a null string.

## MOUNT Intrinsic Function

---

Mounts the specified device.

```
SET x = $ZF("MOUNT",devnam,label,options,lognam)
```

### Parameters

<i>devnam</i>	A device name.
<i>label</i>	A device label.
<i>options</i>	One or more mount options, as specified in the DCL Dictionary.
<i>lognam</i>	A logical name to assign to the device.

### Description

This function mounts a device. It calls LIB\$SPAWN to issue the DCL command MOUNT.

## DISMOUNT Intrinsic Function

---

Dismounts the specified device.

```
SET x = $ZF("DISMOUNT",devnam,options)
```

### Parameters

<i>devnam</i>	A device name.
<i>options</i>	One or more dismount options, as specified in the DCL Dictionary.

### Description

This function dismounts a device. It calls LIB\$SPAWN to issue the DCL command DISMOUNT.

# DIRECTORY Intrinsic Function

Returns the default directory.

```
SET x = $ZF("DIRECTORY",dir)
```

## Parameters

<i>dir</i>	A new directory name.
------------	-----------------------

## Description

This function returns the current OpenVMS default directory, and optionally sets the OpenVMS default directory to the specified directory.

# PARSE Intrinsic Function

Parses a file name.

```
SET x = $ZF("PARSE",filename,defname,relname,keyword,flag)
```

## Parameters

<i>filename</i>	The name of the file to be parsed.
<i>defname</i>	The default file name.
<i>relname</i>	A related file name.
<i>keyword</i>	Specifies which field value to return from the parsed file name. Available values are: NODE, DEVICE, DIRECTORY, NAME, TYPE, VERSION.
<i>flag</i>	A flag to indicate whether or not to perform an I/O check. 1 = no I/O check is performed. 0 = I/O check is performed. The default is 0.

## Description

This function parses the specified file name and returns a parsed file name. PARSE calls the RMS \$PARSE service.

---

# GETFILE Intrinsic Function

---

Gets file information.

```
SET x = $ZF("GETFILE",filename,keyword)
```

## Parameters

<i>filename</i>	The name of a file.
<i>keyword</i>	Specifies which attributes to return from the file.

## Description

This function returns information about the specified file. It emulates the F\$FILE\_ATTRIBUTES lexical function, using the same keywords. GETFILE returns the UIC in standard OpenVMS format, and returns TRUE or FALSE as 1 or 0.

## Parameters

### *keyword*

Keywords for GETFILE: "ALQ", "BDT", "BKS", "BLS", "CBT", "CDT", "CTG", "DBS", "DEQ", "DID", "DVI", "EDT", "EOF", "FID", "FSZ", "GRP", "GBC", "IBS", "LRL", "LVL", "MBM", "MRN", "MRS", "NOA", "NOK", "ORG", "PRO", "PVN", "RAT", "RCK", "RDT", "RFM", "RVN", "SIZ", "UIC", "WCK"