



Caché Switches

Version 5.1
15 June 2006

Caché Switches

Caché Version 5.1 15 June 2006

Copyright © 2006 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. The primary document development tools were special-purpose XML-processing applications built by InterSystems using Caché and Java.



The Caché product and its logos are trademarks of InterSystems Corporation.



The Ensemble product and its logos are trademarks of InterSystems Corporation.



The InterSystems name and logo are trademarks of InterSystems Corporation.

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.

Caché, InterSystems Caché, Caché SQL, Caché ObjectScript, Caché Object, Ensemble, InterSystems Ensemble, Ensemble Object, and Ensemble Production are trademarks of InterSystems Corporation. All other brand or product names used herein are trademarks or registered trademarks of their respective companies or organizations.

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

Table of Contents

- Caché Switches..... 1**
 - 1 Currently-Defined Switches 1
 - 2 Manipulating Switches 2
 - 3 Other Considerations 5
 - 3.1 Locality 5
 - 3.2 Failure Modes 6

Caché Switches

Background

Switches in Caché have their genesis in the physical contacts once part of computer operator consoles or included in the front panel of microcomputers. By setting one of these switches, an operator could convey a single bit of information to the programs running on the machine at that time. Since Caché implements a “virtual machine”, the concept of the switch for this machine has been similarly abstracted.

Today, switches in Caché are represented as individual bit settings in the shared, common memory of a Caché instance; they are visible to all Caché processes. While several have been set aside for users, most influence the operation of Caché itself.

1 Currently-Defined Switches

All switches are identified by number. They are initialized to zero (off) when Caché starts. The following table gives the switch number(s) and their effect:

Switch	Meaning / Use
0 — 7	Reserved for use by applications programs.
8	Inhibits existing Caché daemons from responding to network requests.
9	Inhibits the creation of new daemons to process network logins.
10	Inhibit all global access except by the process that sets this switch. Also inhibit routine accesses that causes disk IO except for this process.
11	Inhibit all global access except for the system job that sets this switch. This overrides switch 10 and is reserved for use by the system. This switch is set, for example, by the backup process to quiesce system activity before copying.
12	Inhibits the ability to login to Caché. Users who attempting to login will receive a message: "Sign-on and JOB inhibited: Switch 12 is set".
13	Inhibits all global SETs, KILLs and ZSAVE commands; only read access is allowed to globals and routines.

Switch	Meaning / Use
14	Inhibits all access to all globals and all routines.
15	Allow network references from peers, even if switch 10,13, or 14 would normally prevent the access.
16	Used internally by Caché to coordinate shutdown activity.
17	Bypass wait for completion of journal flush on clusters.
18	Inhibits pausing added processes if the queue for a block gets too long.
19	Inhibit the start of new transactions.
20	Inhibits dead job cleanup. Cleaning up dead jobs may result in transactions not being rolled back and locks remaining unreleased. (InterSystems recommends putting a call in your ZSTU to set switch 20 so that no dead job cleanup can be done.)
21 — 31	Undefined and reserved for InterSystems use.

CAUTION: Unless specifically directed to do so by InterSystems personnel or its documented procedures, customers should confine any switch activity by their applications to the set reserved for their use, 0 — 7.

2 Manipulating Switches

The `^SWSET` routine is used to directly manipulate the values of the switches. In addition, other Caché facilities, such as those that work with journals on clustered systems and system backup, also set them on behalf of their callers.

SWSET

SWSET

Parameters

rtm	The name of the target routine.
extent	A string holding the parsed extension.
version	A string giving the version number.
namespace	A string giving the namespace, if any, contained in the routine name.

Description

This provides an interactive way to set the value of the switches from, for example, a terminal session.

Remarks

When invoked as in the example below, the routine will prompt for the switch number and then prompt for the value to be set in the switch (0 or 1).

Examples

The following example demonstrates the use of **SWSET** . After executing

```
DO ^SWSET
```

the user will successively see the following:

```
Set/Clear switch #:
```

```
Set/Clear switch #: 2
```

```
Set/Clear switch #: 2 to value (0 or 1):
```

```
Set/Clear switch #: 2 to value (0 or 1): 1
```

```
Set/Clear switch #: 2 to value (0 or 1): 1...done
```

%swstat^SWSET

```
%swstat^SWSET
```

Parameters

switch	The number of the switch.
--------	---------------------------

Description

This function returns the current setting for the switch.

Parameters

switch

The number of a valid switch.

Remarks

If the switch is a valid number, this function returns the value of the switch as one of the following:

- 0 — the switch is reset (off)
- 1 — the switch is set (on)

otherwise it returns a value of -1 indicating that an error has occurred.

Examples

The following example prints the value of switch number 1.

```
Write $$%swstat^SWSET(1)
```

%swset^SWSET

```
%swset^SWSET
```

Parameters

switch	The number of the switch.
value	The value it should have, 0 or 1.

Description

This function sets the switch to the specified value.

Parameters

switch

The number of a valid switch.

value

The value 0 or 1.

Remarks

If the switch is a valid number and value is either a 0 or 1, this function sets the switch to that value and returns:

- 0 — the switch is now reset (off)
- 1 — the switch is now set (on)

otherwise it returns a value of -1 indicating that an error has occurred.

Examples

The following example sets the value of switch number 1 to off.

```
Write $$%swstat^SWSET(1, 0)
```

3 Other Considerations

3.1 Locality

As noted in the introduction, users should view the switches as being local to a Caché instance. Although Caché itself provides mechanisms to propagate the meaning of certain settings to other members of a cluster or ECP configuration, these are for InterSystems internal use only. The values of the user switches cannot be moved to other systems.

3.2 Failure Modes

A Caché process which sets one of the system-reserved switches and terminate without properly cleaning up its work can leave the system in a restricted operating mode. For example, a process that sets switch 12 and then suffers a catastrophic failure (or even merely HALTs) will leave Caché in a state where no further users can login. If this situation occurs, the administrator or operator is urged to call the [InterSystems Worldwide Response Center](#).

Note: The only situation for which Caché implements an automatic recovery is for switch 10. If a process sets this switch and then HALTs, Caché will automatically reset the switch to zero.