# Caché Release Notes and Upgrade Checklist

Version 2018.1
2019-09-20

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

# Table of Contents

# About This Book

This book provides information on the major features that have been added to Caché in this release, as well as information needed to perform a successful upgrade to this release.

It contains the following sections:

- New and Enhanced Features for Caché 2018.1

- General Upgrade Information

- Upgrade Checklist for Caché 2018.1 — READ THIS INFORMATION BEFORE UPGRADING

The following books provide related information:

- Caché Release Notes and Upgrade Checklist Archive provides information on previous releases — both new features as well as information needed for upgrades.

- The online *InterSystems Supported Platforms* document for this release lists the technologies supported by this release of Caché.

- Caché Installation Guide describes the process of installing Caché on your system.

- Introduction to Caché provides an overview of the features and major components of Caché.

# General Licensing Notes

InterSystems makes its products and features available under license to customers. While InterSystems may or may not enforce the use of said products or features consistent with the purchased license capabilities, customers are expected to comply with terms of their licenses. Moreover, InterSystems may tighten enforcement at any release without notice.

Developers must be aware that certain license types are required in order to use specific product features such as Multi-Server capability, Mirroring, and Web Services features. The specific requirements are noted in the InterSystems Price List and the Terms and Conditions for licensing. These are available from your local InterSystems representative.

# Application Use Of InterSystems Web Server

InterSystems installs an Apache-based web server (often referred to as the "private web server") to assure that the Management Portals for its products are always available. The private web server is built and configured to meet the management needs of InterSystems administrative servers and is configured to only connect to InterSystems products. The options selected are not, in general, suitable for production use - in particular, security is minimal and the options used are generally unsuitable for a high volume of HTTP requests. Testing, by InterSystems, of the private webserver only covers use of the private web server for managing Caché, Ensemble, HealthShare, and other InterSystems products.

Customers are not required to use this web server to manage our products. You may also use a web server of your choice, located on whatever server you elect to use. The private web server is provided as a convenience to simplify installation and installation dependencies. Many developers also find it useful to use the private web server for unit testing.

## UNIX®

The parameters used for the UNIX® build are:

```
--prefix=<installation_location>
--disable-actions
--disable-asis
--disable-auth
--disable-autoindex
--disable-cgi
--disable-cgid
--disable-charset-lite
--disable-dir
--disable-env
--disable-imap
--disable-include
--disable-negotiation
--disable-setenvif
--disable-shared
--disable-status
--disable-userdir
--enable-access
--enable-alias
--enable-log-config
--enable-mime
--enable-so
--without-berkeley-db
--without-gdbm
--without-ndbm
```

The server produced has defaults using the Apache Group's prefork Multi-Processing Module (MPM). This is the non-threaded server model. The number of requests that can be concurrently served is directly related to the number of Apache worker processes in the pool. The private web server is configured to occupy the smallest possible footprint by allowing a maximum of two worker processes to be created for the pool. The following settings will be found in the Apache configuration (httpd.conf) for the server:

- MinSpareServers: 1

- MaxSpareServers: 2

By contrast, the default Apache configuration for a production grade build is usually the following:

- StartServers: 5

- MinSpareServers: 2

- MaxSpareServers: 20

- ServerLimit: 256

- MaxClients: 256

This configuration will allow Apache to create 5 worker processes at start-up time, increasing to a maximum of 256 as the concurrent load increases. Because of these differences in configuration, the performance of the private web server will be noticeably inferior to that of a production grade Apache build as the concurrent load increases.

## Windows

Windows-based Apache installations use the official binary distribution for Windows and a special multithreaded Multi-Processing Module (MPM) which is more suited to the way the operating system is optimized. However, since InterSystems installs and loads only a small subset of modules (mod_alias.so, mod_authz_host.so, mod_log_config.so and mod_mime.so.), their functionality is limited.

## Conclusion

If you expect very low volume of HTTP traffic, have limited demands for high availability and secure operation, the private web server may be suitable for your deployment needs. However, if you expect a high amount of HTTP traffic, require high availability in your web server, need to integrate with other sources of web information, or need a high degree of control over your web server, InterSystems recommends installing your own separate copy of Apache, ideally on its own

server, and configuring it to use our CSP gateway to communicate with Cache, Ensemble, or HealthShare. Review the options above to determine if this is so.

# 1

# New and Enhanced Features for Caché 2018.1

This chapter includes:

## 1.1 Key Management Interoperability Protocol (KMIP)

In this release, Caché can be a client to an enterprise key management server and use the Key Management Interoperability Protocol (KMIP) to store and retrieve keys on the server. These keys can be used to encrypt data at rest, such as databases and journal files. KMIP, an OASIS standard, gives you the power of centralized key management.

## 1.2 Microsoft Volume Shadow Copy (VSS) Integration

This release introduces Cache and Ensemble integration with Microsoft's Volume Shadow Copy Service (VSS) framework. This integration allows any backup product which supports VSS integration to identify databases and the volumes they reside on and seamlessly quiesce Cache and Ensemble instances when performing a backup of the system. The copies of the databases included in a VSS shadow are physically, although not logically with regards to transactions, consistent and therefore may be restored individually. To ensure the transaction integrity of these restored databases, one should also replay journal files accordingly.

# 1.3 Support for Microsoft Integrated Windows Authentication for HTTP Connections

This new enhancement allows %Net.HttpRequest to use windows based authentication over HTTP 1.1 to establish a connection to a secure server. Users can provide credentials or, if no credentials are provided, the system will try to authenticate using the current logged in context. Client may initiate a connection to the server with an "Authorization" header or try to establish a connection without that header and process the 401 status code with its associated with WWW-Authenticate header and then respond with the appropriate authentication mechanism. The supported authentication schemes are Negotiate (Kerberos & NTLM), NTLM, and Basic.

# 1.4 SSH Enhancements

Establishing SSH connections for Caché and Ensemble uses a new SSH library with multiple enhancements to functionality and performance.

# 1.5 New and Enhanced Features for Caché 2018.1.2

This release includes substantial improvements to journaling performance. This enables Caché to perform updates more efficiently and to handle larger throughput of data. For a detailed list of the changes in this maintenance release, see the Caché and Ensemble Maintenance Release Notes (2018.1.2).

# 1.6 New and Enhanced Features for Caché 2018.1.1

This release includes the substantial scalability and performance improvements, particularly for large-scale Non-Uniform Memory Access (NUMA) systems. This includes changes to improve scalability in statistics tracking and global buffer management, performance improvements in use of subscript level mapping, and more effective optimizations to avoid traversing global pointer blocks. To enable these improvements there are minor changes to memory utilization and system statistics described in the Upgrade Checklist.

These enhancements increase the amount of memory allocated for global buffer metadata by 64 bytes per buffer on Intel systems and by 128 bytes per buffer on IBM Power systems. For example, with 8K buffer sizes, the shared memory allocated for a global buffer increases by 0.75% on Intel systems and by 1.5% on IBM Power systems. These enhancements also cause minor changes is in statistics displayed by utilities and the Management Portal.

In addition, this maintenance release adds Cross-Origin Resource Sharing (CORS) support for %CSP.Page subclasses. In previous releases CORS support was only available in %CSP.REST subclasses.

For a detailed list of the changes in this maintenance release, see the Caché and Ensemble Maintenance Release Notes (2018.1.1).

# 2

# General Upgrade Information

InterSystems' ultimate goal is to have a release which can be installed with no, or little, effect on the applications it supports. This release provides an easier upgrade path if you are upgrading from version 2013.1 or later. If you are upgrading from a release prior to 2013.1, you must still follow the prior recommendations; for details, see "Pre-2014.1 Upgrade Information" in the *Caché Release Notes and Upgrade Checklist Archive*.

## 2.1 Important Considerations

### 2.1.1 Compatibility

The goal of each release is a smooth upgrade to new and improved functionality with no changes required to existing applications. However, because of error corrections, significant enhancements, or changes to applicable standards, this goal cannot always be met. In this case, InterSystems discloses all identified instances where changes to applications are necessary so that customers can gauge effort required to move to the new release.

This book describes the details changes made in that release that may affect applications upgrading from the prior release; the information for older releases is in the *Caché Release Notes and Upgrade Checklist Archive*. Among the items listed are: changes to the way the system operates; enhancements to the class and routine compilers; detailed changes made to system classes in terms of parameters, properties, and methods removed; changes to method call signatures; and differences in method returns.

You may, after reading the release-specific changes, conclude that none of them affect your application. Even in this case, regardless of how robustly designed and how well implemented your application is, there is no substitute for quality assurance test results that confirm your judgement and demonstrate the application remains unaffected by the upgrade.

**Important:**     InterSystems recommends that each application be thoroughly tested in the upgraded environment **before** it is deployed to customers and begins processing live data.

### 2.1.2 Technology Previews

In recent releases, InterSystems made available software labeled as "experimental", or "preview". The purpose of these offerings is to allow customers to participate more closely in the development process by giving them access to software during a period where their feedback can influence the course of future evolution.

As long as its status remains "experimental", this software will change during the course of a release and, depending on how long it remains in that state, even from one release to another. Some of these changes may be significant enough to make the latest software version incompatible with prior versions. Therefore, customers who wish to take advantage of

such software in their application should consult with InterSystems beforehand. InterSystems will be happy to provide details on the implementation and its probable future path toward becoming a standard part of a release.

## 2.1.3 Field Test

Toward the end of development for each release, InterSystems makes available successively more complete copies of the final product to its customers. The notifications are published on the website and on public blogs. The purpose of this is two-fold:

- It provides an early opportunity for customers to determine how the changes and enhancement in the release affect existing applications, to report issues found, and verify those issues have been resolved.

- It also provides exposure of important proposed features (the experimental software) which may become part of future releases. Customer have a chance to try out the proposed ideas and give feedback on the usefulness of this feature for their business area.

InterSystems strongly encourages customers to plan on obtaining one or more instances of the field test and to test their application against it.

**Important:**     InterSystems does not support upgrading from a field test version.

### Unresponsive Systems

One of the goals for field test is to expose the new release to real-world operating challenges to assure its reliability. Therefore, it is possible, although unlikely, that an unanticipated sequence of events may render Caché unresponsive. In this situation, it is extremely important to gather system diagnostic information while in the hung state for InterSystems to analyze. Should an instance of Caché become unresponsive,

- Log in as an administrator

- In a terminal window, run the CacheHung script in the directory, <install-dir>/bin.

    The scripts corresponding to supported systems are:

    – Windows: CacheHung.cmd

    – UNIX®, Linux, AIX, and so on: CacheHung.sh

- Send the resulting output file to the InterSystems Worldwide Response Center. You can email the file to support@inter-systems.com, open a new problem using the WRC Online, or call the Center directly for additional assistance.

# 2.2 Upgrade Specifics

This section contains specific instructions applicable to this transition.

## 2.2.1 Classes

InterSystems recommends that customers recompile all their classes contained in each namespace. This will assure that:

- Subclasses derived from the InterSystems product library will see improved product behavior if a method call results in executing code in its superclass(es).

- All embedded SQL will use the latest versions of the SQL infrastructure.

- All projections involved in language bindings will be updated.

- All generated routines and classes will be updated.

### 2.2.1.1 Class compiler version utility

To assist customers in determining which class compiler version a class or classes in a namespace have been compiled with, InterSystems provides two assists

- Method – $System.OBJ.CompileInfoClass(<classname>)

  This method returns the version of the class compiler used to compile this <classname> and the datetime the class was compiled

- Query – $System.OBJ.CompileInfo(<sortby>)

  This query generates a report for the current namespace that includes all classes, the version of the compiler used to compile each one, and the datetime each was compiled. The first argument <sortby> may have the following values:

  - 0 – the time the class was compiled
  - 1 – the class name
  - 2 – the version of Caché the class was compiled in

## 2.2.2 Routines

ObjectScript routines, and MultiValue and Caché Basic programs do not need to be recompiled after upgrade with the following exceptions:

- Routines containing embedded SQL must be recompiled.
- MultiValue paragraphs containing queries must be recompiled.

## 2.2.3 Cached Queries

Cached queries are always purged during upgrade. They are recompiled and cached as needed.

## 2.2.4 Web Services and SOAP

It is not necessary to re-import any Web Service Definition (WSDL) files.

# 3

# Upgrade Checklist for Caché 2018.1

**Purpose**

The purpose of this chapter is to highlight those features of Caché 2018.1 that, because of their difference in this version, affect the administration, operation, or development activities of existing systems.

Customers upgrading their applications from earlier releases are strongly urged to read the upgrade checklist for the intervening versions as well. This document addresses only the differences between 2017.2 and 2018.1.

**Important:**    Before upgrading Caché, read "Upgrading Caché" in the *Caché Installation Guide* for instructions. If you are upgrading from a version earlier than Caché 2012.1, pay special attention to Supported Upgrade Paths in the *Caché Installation Guide*.

The maintenance release Caché 2018.1.1 introduced the following upgrade checklist items:

- Increase in Memory Usage

- DeepSee Cubes Require Recompile

- Complex MDX Queries Are Rejected

- ID Notation in XEP

- System Statistics Changes

## 3.1 Administrators

This section contains information of interest to those who are familiar with administering prior versions of Caché and wish to learn what is new or different in this area for version 2018.1. The items listed here are brief descriptions. In most cases, more complete descriptions are available elsewhere in the documentation.

### 3.1.1 Operational Changes

#### 3.1.1.1 Increase in Memory Usage (Version 2018.1.1 and later)

Maintenance Release 2018.1.1 provides performance and scalability enhancements for large-scale systems. These enhancements increase the amount of memory allocated for global buffer metadata by 64 bytes per buffer on Intel systems and by 128 bytes per buffer on IBM Power systems. For example, with 8K buffer sizes, the shared memory allocated for a global buffer increases by 0.75% on Intel systems and by 1.5% on IBM Power systems. There is also an increase in per-process memory usage for systems using large numbers of subscript level mappings.

### 3.1.1.2 Allocating Space for Write Image Journaling (WIJ) Files

This release contains a new parameter, Target size for the WIJ, in the system configuration journal settings. Setting this target size ensures that disk space is allocated for the WIJ early in the start-up process. If sufficient space is not allocated early and there is not enough available space for the WIJ, the system may encounter problems. Allocating space for WIJ is an advanced configuration setting. If you encounter issues with this, contact the InterSystems Worldwide Resource Center.

### 3.1.1.3 Changes to JRNDUMP Utility Navigation

The JRNDUMP utility has improved navigation. If you have a script that responds to JRNDUMP prompts, you should check the script to ensure that it handles the changed navigation.

## 3.1.2 Platform-Specific Items

This section holds items of interest to users of specific platforms.

### 3.1.2.1 Windows Upgrade Installation Does Not Create USER Namespace if Previously Deleted

In previous releases, if, on a Windows system, you upgraded Caché where the USER namespace had been deleted, the installation would create a new USER namespace. In this release, the installation will not create a USER namespace if it was previously deleted from the instance. This is typically the desired behavior, but, if you have a script that automatically deletes the USER namespace after an upgrade, you may need to modify the script. The Windows install now matches the behavior of Caché installations on other platforms, that is, if the instance does not have a USER namespace, the upgrade does not create one.

# 3.2 Developers

This section contains information of interest to those who have designed, developed and maintained applications running on prior versions of Caché.

The items listed here are brief descriptions. In most cases, more complete descriptions are available elsewhere in the documentation.

## 3.2.1 Class Changes

### 3.2.1.1 Removed Classes

No classes have been removed in this release that were present in the previous version.

### 3.2.1.2 Removed Methods

The following methods were present in the previous version and have been removed in this release:

- %Net.SSH.SFTP

    - genPattern

- Config.GenericMapMethods

    - Create

### 3.2.1.3 Removed Properties

No properties have been removed in this release that were present in the previous version.

### 3.2.1.4 Removed Parameters

No parameters have been removed in this release that were present in the previous version.

### 3.2.1.5 Removed Indices

No indices have been removed in this release that were present in the previous version.

### 3.2.1.6 Modified Methods

The following methods have different signatures in this version of Caché:

- %CSP.UI.Portal.Mirror.Utils

    - 2017.2: `modifiedAddress () As (none)`

    - 2018.1: `modifiedAddress (NewVal, Type, useZen) As (none)`

    - 2017.2: `modifiedAgentAddress () As (none)`

    - 2018.1: `modifiedAgentAddress (NewVal, Type, useZen) As (none)`

- %DeepSee.Component.Widget.pivot

    - 2017.2: `exportPDF (printMultiple, preserveTempFiles) As (none)`

    - 2018.1: `exportPDF (printMultiple, preserveTempFiles, filename) As (none)`

- %DeepSee.Component.Widget.widget

    - 2017.2: `printSVGContent (svgFrameId, parms, svgContent) As (none)`

    - 2018.1: `printSVGContent (svgFrameId, parms, svgContent, filename) As (none)`

- %Net.FtpSession

    - 2017.2: `setupPASV (&Device:%String) As %Boolean`

    - 2018.1: `setupPASV (&Device:%String, *ServerName:%String) As %Boolean`

    - 2017.2: `setupPORT (&Device:%String) As %Boolean`

    - 2018.1: `setupPORT (&Device:%String, *ServerName:%String) As %Boolean`

- %Net.HttpRequest

    - 2017.2: `OutputHeaders () As (none)`

    - 2018.1: `OutputHeaders () As %Status`

- %Net.SSH.SFTP

    - 2017.2: `Test (host:%String, username:%String, password:%String, dir:%String="/etc", spec:%String="", dotFiles:%Boolean=1) As %Status`

    - 2018.1: `Test (host:%String, username:%String, password:%String, dir:%String="/etc", spec:%String="", dotFiles:%Boolean=1, &t) As %Status`

- %SYSTEM.License

- 2017.2: `GetUserId () As %String`

- 2018.1: `GetUserId (PID:%String, JobNumber:%String) As %String`

## 3.2.2 Debugging Changes

### 3.2.2.1 Debugger on AIX Breaks After True Post-Conditional

When you are stepping through a routine on an AIX system and a command post-conditional is true, the debugger generates an additional <BREAK> before executing the command.

## 3.2.3 Terminal Changes

The cterm application now accepts terminal input directly as Unicode instead of ANSI characters. This change should not impact terminal usage and behavior, but it is possible that there may be differences in some locales. This change allows Caché to support additional languages.

## 3.2.4 DeepSee Changes

### 3.2.4.1 DeepSee Cubes Require Recompile (Version 2018.1.1 and later)

After upgrading to Caché 2018.1.1, you must recompile all DeepSee cubes. The 2018.1.1 Maintenance Release contains a fix to the %DeepSee.Generator that requires this recompile. If you build a cube without recompiling, it may cause an error.

### 3.2.4.2 Complex MDX Queries Are Rejected (Version 2018.1.1 and later)

In previous releases, a complex MDX query with nested %OR setfunctions could produce erroneous results. This release detects the constructs, blocks them, and issues an error. It is possible to produce the intended logical result with a simpler construct. You should rewrite these queries so that any %OR contains only AND terms.

### 3.2.4.3 Recompile Measure-Specific Listings

This release contains a correction to handle listing filters within compound cubes. You must recompile measure-specific listings to get this fix.

### 3.2.4.4 GetCubeMeasures Changes

The %DeepSee.Utils:%GetCubeMeasures() method should return the caption as the second item in the output list. For calculated measures it was returning the name. In this release, it correctly returns the caption. If your code depended on the previous incorrect behavior, you should modify it.

## 3.2.5 Language Binding Changes

### 3.2.5.1 ID Notation in XEP (Version 2018.1.1 and later)

In C# classes used for XEP, developers can specify an id annotation. In previous releases, this could cause bulk updates to fail. To avoid this problem, Caché now generates id_property. In some cases, schemas with the generated id annotation may encounter a problem. If you encounter this problem, remove the annotation and the behavior will be unchanged from the previous version.

### 3.2.5.2 Issue Using C++ Binding with Output Redirection

In order to perform output redirection, the C++ binding stores and resets the $ZR value. If $ZR points to a database that the user does not have write access, this encounters a protect error.

## 3.2.6 Security Changes

### 3.2.6.1 Changes to HttpRequest Security

In this release, %Net.HttpRequest is enhanced to support SPNEGO, Kerberos and NTLM authentication, as well as Basic authentication, for HTTP clients. This new authentication process for %Net.HttpRequest changes the existing %Net.HttpRequest support for Basic authentication. In previous releases, setting the Username property will as a side-effect return a Basic Authorization header. This behavior will only continue for HTTP 1.0 responses. For HTTP 1.1 responses an unsolicited Authorization header will never be sent unless the InitiateAuthentication property is set. If the current Basic authentication support is used, it will result in a 301 status response with a WWW-Authenticate header indicating Basic and possibly other schemes. HttpRequest then sends another request with the proper Authorization header. Existing code will continue to work but there will be an extra round-trip

## 3.2.7 System Changes

### 3.2.7.1 System Statistics Changes (Version 2018.1.1 and Later)

To improve performance of large-scale systems, the statistics reported by the system utilities have changed. In most cases this will not cause any compatibility issues; however, if your code explicitly examines the statistics output for specific data, you may have to update your code to adjust for these changes. The following minor changes are visible:

- The ^GLOSTAT utility no longer offers detailed block type statistics and no longer asks the question "Should detailed statistics be displayed for each block type? No =>". Additionally a bug is corrected in ^GLOSTAT which was present since 2016.1.0 and caused remote reads and remote cache efficiency to be incorrect (usually zero).

- The Management Portal **System Dashboard** is enhanced to show private global references and updates and WIJ writes, in the same way that ^GLOSTAT does. The detailed block type statistics page, 'Disk and Buffer Statistics', is no longer available.

- The ability to zero statistics (from ^GLOSTAT and the Management Portal) is not available. This capability is not compatible with the more efficient statistics collection. Most counters are now 64-bit and timed collection is available, so there is no need to zero the counters.

- The class SYS.Stats.Disk, which accesses the detailed block type statistics, reports all as zero; the class reference is updated to reflect this. The class SYS.Stats.Global, which accesses the main global statistics, continues to function as before and is enhanced to include write image journal writes as a property called 'WIJWrites'. Additionally, the internal class SYS.Metrics was previously exposed (unintentionally) as a public API; it is now properly marked as internal-only and it no longer reports values for detailed block type statistics.

- The WS-Monitoring and BMC Patrol facilities that report the detailed block type statistics, report them zero or null. BMC Patrol, WMI, and SNMP monitoring facilities lock counts (total, success, and fail) also report as zero.

- The cstat -gl is updated to include more fields that were previously only available in ^GLOSTAT displays. Some statistics that were available via the cstat -c option are no longer available.

**Note:** %Monitor.Process, MONLBL, PERFMON and similar performance monitoring tools are unaffected by this change and maintain their own counters.