Microsoft Internet Information Services Web Server Considerations (Windows)

Version 2024.1
2024-05-16

InterSystems Corporation 1 Memorial Drive Cambridge MA 02142 www.intersystems.com

Microsoft Internet Information Services Web Server Considerations (Windows)
InterSystems IRIS Data Platform Version 2024.1 2024-05-16
Copyright © 2024 InterSystems Corporation
All rights reserved.

InterSystems®, HealthShare Care Community®, HealthShare Unified Care Record®, IntegratedML®, InterSystems Caché®, InterSystems Ensemble® InterSystems HealthShare®, InterSystems IRIS®, and TrakCare are registered trademarks of InterSystems Corporation. HealthShare® CMS Solution Pack™ HealthShare® Health Connect Cloud™, InterSystems IRIS for Health™, InterSystems Supply Chain Orchestrator™, and InterSystems TotalView™ For Asset Management are trademarks of InterSystems Corporation. TrakCare is a registered trademark in Australia and the European Union.

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

Table of Contents

Microsoft Internet Information Services Web Server Considerations (Windows) 1

1 Restarting IIS 1

2 IIS Worker Processes and the Web Gateway 1

2.1 Web Gateway System Status 1

2.2 Minimum and Maximum Connections 2

2.3 Worker Process Timeout and Worker Process Recycling 2

3 Bitness: Serving a 32-bit Application on a 64-bit Server 2

Microsoft Internet Information Services Web Server Considerations (Windows)

This page provides additional technical details which you may wish to consider as you deploy an InterSystems Web Gateway alongside a Microsoft Internet Information Services (IIS) web server. Refer to IIS documentation for further details about IIS administration: https://learn.microsoft.com/en-us/iis/get-started/introduction-to-iis/iis-web-server-overview.

1 Restarting IIS

IIS must be restarted in order for changes to the Web Gateway's configuration to take effect. This must be done by completely restarting the World Wide Web Publishing service from the main Windows Services control panel or through the Windows command line, not through the Internet Services Manager control panel. See Restarting IIS.

2 IIS Worker Processes and the Web Gateway

The recommended Web Gateway deployment method (using the Native Modules CSPms.dll and CSPmsSys.dll) manages persistent resources—such as connections to an InterSystems IRIS instance—within the web server extension itself. (By contrast, NSD-based Web Gateway deployments manage resources independently of IIS, within the NSD itself.)

Therefore, the performance of the Web Gateway Native Module extension is influenced by the following IIS configuration items, which determine how IIS worker processes serve requests for your applications:

Application pool

Within IIS, a designated set of one or more worker processes which can be assigned to serve requests for one or more IIS applications.

Web garden

An IIS application pool which contains more than one worker process. Requests are distributed among the worker processes in a web garden.

Each worker process manages its own instance of the Web Gateway extension. For application pools that are configured to use no more than one worker process, this has no visible impact on the way the Web Gateway operates within the context of a single web application path (for example, /csp). However, within a web garden, the requests are evenly distributed amongst the Web Gateway instances managed by participating worker processes.

Because of this, several restrictions must be borne in mind.

2.1 Web Gateway System Status

The Web Gateway's System Status management page cannot accurately monitor the connections used by web applications across an entire web garden. At any given time, the Systems Status page reflects the status for the instance of the Web Gateway that happens to be attached to the current worker process (that is, the worker process that happens to service the Web Gateway's request).

2.2 Minimum and Maximum Connections

Each worker process manages its own instance of the Web Gateway application, and therefore maintains its own pool of persistent connections to InterSystems IRIS application servers. The Web Gateway configuration parameters which specify the minimum and maximum number of InterSystems IRIS connections do not specify those limits in aggregate across an entire application pool. Rather, these parameters define the minimum and maximum number of connections allowed for each worker process's Web Gateway instance.

2.3 Worker Process Timeout and Worker Process Recycling

You may wish to configure worker processes within an application pool to recycle periodically or to terminate after a specified period of idle time.

When a worker process within an application pool terminates, the instance of the Web Gateway that the process manages terminates as well, closing the pool of connections that the Web Gateway instance maintained with InterSystems IRIS. Under such circumstances, stateless connections can be replaced in a way that is transparent to users of a web application. However, state-aware sessions (preserve mode 1) terminate when their hosting connection is closed.

3 Bitness: Serving a 32-bit Application on a 64-bit Server

Note: This section applies to modules that are loaded into the address space of the hosting web server: the Native Modules (CSPms[Sys].dll and CSPcms.dll) and the ISAPI extensions. CGI modules are not affected since they run as a process which is detached from IIS.

IIS allows you to set the bitness (64-bit or 32-bit) for a particular application pool. A single IIS installation can feature application pools which serve native 64-bit applications alongside other application pools which serve 32-bit applications. The **Enable 32-Bit Applications** setting is available by accessing the **Advanced Settings** for an application pool in the IIS Manager.

The Native Modules or ISAPI extensions which an application pool loads must match the bitness of the application pool. For example, if the hosting application pool is 64-bit, then the 64-bit Gateway modules (such as CSPms[Sys].dll) must be used. If the hosting application pool is 32-bit, then the 32-bit Gateway modules must be used instead.

The bitness check for individual modules is done via a *preCondition* in the module's web.config file. For the Web Gateway, this file typically looks something like the following:

```
<?xml version="1.0" encoding="UTF-8"?>
<configuration>
   <system.webServer>
      <handlers>
         <add name="WebGateway_All" path="*" verb="*" modules="CSPms" resourceType="Unspecified" \
          preCondition="bitness64" />
      </handlers>
      <security>
         <requestFiltering>
            <hiddenSegments>
               <remove segment="bin" />
            </hiddenSegments>
         </requestFiltering>
      </security>
   </system.webServer>
</configuration>
```

Note the bitness setting in the *precondition* clause. In this case bitness is set to `bitness64` which means that IIS checks for 64-bit Gateway modules operating in a 64-bit Application Pool.

If a 32-bit Application Pool is used, then the 32-bit Gateway modules must be used and the *preCondition* set to `bitness32`.

If there is an inconsistency between the modules installed, the precondition clause, or the expectations of the hosting Application Pool, IIS returns an error.