

---

Article

[Erik Hemdal](#) · Jul 22, 2019 3m read

## Unexpected Write Daemon hang in an InterSystems IRIS container / Macintosh

One of my colleagues at InterSystems encountered an unexpected issue when running InterSystems IRIS on a Macintosh in a container using Docker for Mac. I'd like to share what we found, so you might avoid running into similar issues.

### The Problem

The task at hand was running a Java application with XEP to do a large data load into IRIS. When running the data load, the write daemon hung soon after starting the job, with messages like these in messages.log:

```
05/21/19-14:57:50:625 (757) 2 Process terminated abnormally (pid 973, jobid 0x00050016)
(was a global updater)
```

```
05/21/19-14:58:52:990 (743) 2 CP: Pausing users because the Write Daemon has not shown
signs of activity for 301 seconds. Users will resume if Write Daemon completes a pass or writes
to disk (wdpass=98).
```

This problem was completely reproducible and was very mysterious, so Support got involved.

### What we found

We were able to start the SystemPerformance utility while reproducing the problem and discovered the issue readily.

In the iris.cpf file, the cache for 8KB databases was set to 4GB:

```
globals=0,0,4096,0,0,0
```

That looked reasonable for an instance running on a machine with 8GB of memory. Since this was a test, the Mac was otherwise not heavily loaded. However, not all of that system memory was actually available to IRIS, as we saw in the output of the Linux free command inside the container:

```
Memtotal,  used,  free,  shared,buf/cache,available,swaptotal, swapused, swapfree,
 1998,    331,   322,   513, 1344,  1003,  1023,    11,  1012,
 1998,    340,   312,   513, 1345,   994,  1023,    11,  1012,
...
```

```
1998, 272, 72, 1563, 1653, 44, 1023, 105, 918,  
...  
1998, 123, 67, 1770, 1807, 12, 1023, 870, 153,  
...  
1998, 135, 54, 1777, 1809, 14, 1023, 1023, 0,
```

Only about 2GB was actually available. During the heavy data load, IRIS rapidly consumed the database cache until all memory and swap space available was exhausted; at which point the instance hung.

### The Cause

Docker relies heavily on some key Linux technologies, particularly cgroups and namespaces, that aren't available natively on platforms like Macintosh and Windows. On these platforms, Docker uses a Linux virtual machine internally: in the case of the Macintosh, this is provided by HyperKit. And as we found, it is possible to overallocate memory on this platform and configure IRIS with more memory than is actually available. If you are using Docker for Mac as your development platform, keep this internal VM in mind and size memory appropriately.

[#Containerization](#) [#Docker](#) [#Tips & Tricks](#) [#InterSystems IRIS](#)

---

#### Source

URL: <https://community.intersystems.com/post/unexpected-write-daemon-hang-intersystems-iris-container-macintosh>