Discussion

Dmitry Maslennikov · Jul 10, 2022

Any stability with IRIS Docker images?

I would say it is a post of pain after years of using InterrSystems IRIS Docker images in many projects.

And I hope InterSystems will hear me and do something with it

We have a lot of issues with Docker images, but I see no progress in solving them.

- containers.intersystems.com - any new releases substitute previous versions, makes build useless
    - ARM64 images have separate names, and it makes a pain to use them
- flags in iris-main, appears and disappears from version to version, which may fail the start the container
- healthcheck does not work as expected

containers.intersystems.com

It is quite difficult to follow all the changes on container.intersystems.com, as there are no anounces about any new releases. But the versions there disappears very easily in any moment, and any Contnious Integration process may fail by this reason, very easily. Even versions supposed to be stable can be replaced by some new build.

For security and stability reasons, the best practice is to use exact version for any used libraries, and IRIS is on an exception, but InterSystems makes this rule is useless, by deleting any previous images.

I see, how it can be solved.

- in addition to tags 2022.2.0.270.0, 2021.1.2.338.0 represent them as 2022.2 and 2021.1
- introduce tags such as latest, lts, preview, according to official release cadence, (e.g, 2020-lts, 2021-lts)
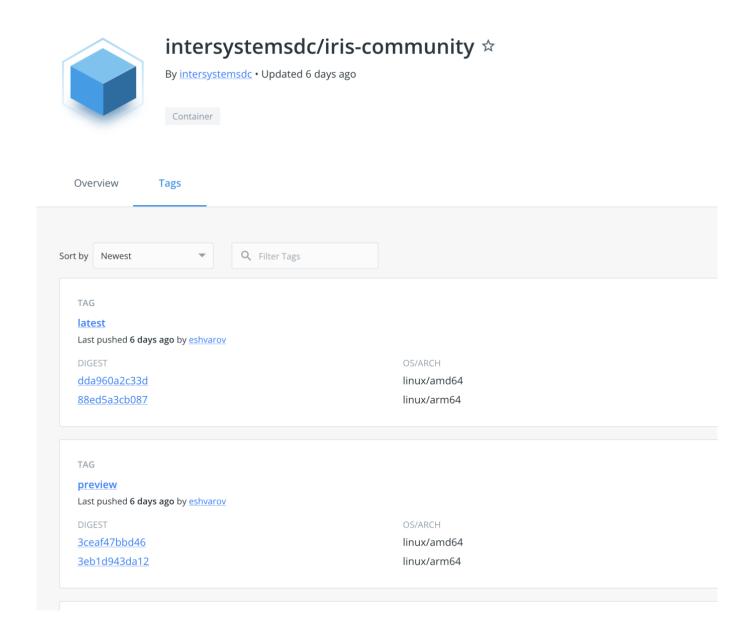
ARM64 images

Developers with ARM64 process, and used vanilla IRIS images, have too fix their Dockerfile, to make it work.

Another issue, is to make multi-platform builds, if any customer would need it, it a lot of pain how to make it with IRIS (even if forget that IRIS does not work in qemu emulation).

Docker registry supports, multi-pllatform images, and I would expect that InterSystems IRIS images would be delivered the same way.

With IRIS Community Edition with ZPM Docker images, we already use this way. Developers on any platform, don't need to bee worried, about new versions from InterSystems or different platform they are running on. Docker will get the right latest version of the image for the running platform.

intersystemsdc/iris-community ☆

By intersystemsdc • Updated 6 days ago

Container

Overview   **Tags**

Sort by | Newest ▾ | 🔍 Filter Tags

TAG

latest
Last pushed **6 days ago** by eshvarov

DIGEST	OS/ARCH
dda960a2c33d	linux/amd64
88ed5a3cb087	linux/arm64

TAG

preview
Last pushed **6 days ago** by eshvarov

DIGEST	OS/ARCH
3ceaf47bbd46	linux/amd64
3eb1d943da12	linux/arm64

---

iris-main flags

Not so far ago with one of the latest version InterSystems introduced a new flag --check-caps, as they said to solve the issue with the latest docker. But, in fact, any previous versions of IRIS versions works well in the same Docker version, so, that issue was added to iris-main, and to solve it was some strange way added this flag. With this new feature in iris-main IRIS does not start in Docker, without that flag. So, instead switching of it by default, as on previous versions, vanilla docker images just will not start without that flag. And than with the newer version 2022.2 they deleted that flag, and any start of IRIS with this flag defined will not start again.

Version 2020.1, successfully starts

```
$ docker run --rm  containers.intersystems.com/intersystems/iris:2020.1.3.521.0
[WARNING] No init process detected!  This container may accumulate zombie processes i
f run for a long time.  Consider using "docker create --init ..." or equivalent.
[INFO] Starting InterSystems IRIS instance IRIS...
[INFO] This copy of InterSystems IRIS has been licensed for use exclusively by:
Local license key file not found.
Copyright (c) 1986-2022 by InterSystems Corporation
Any other use is a violation of your license agreement 1 alert(s) during startup. See
```

```
 messages.log for details.
Starting IRIS 07/10/22-10:47:21:188 (362) 0 [Generic.Event] Automatically configuring
 buffers
07/10/22-10:47:21:242 (362) 0 [Generic.Event] Allocated 1201MB shared memory: 982MB g
lobal buffers, 35MB routine buffers
07/10/22-10:47:21:243 (362) 0 [Crypto.IntelSandyBridgeAESNI] Intel Sandy Bridge AES-
NI instructions detected.
07/10/22-10:47:21:285 (362) 0 [WriteDaemon.UsingWIJFile] Using WIJ file: /usr/irissys
/mgr/IRIS.WIJ
07/10/22-10:47:21:285 (362) 0 [WriteDaemon.CreatingNewWIJ] Creating a new WIJ file
07/10/22-10:47:21:711 (362) 0 [WriteDaemon.CreatedNewWIJ] New WIJ file created
07/10/22-10:47:21:716 (362) 0 [Generic.Event]
Startup of InterSystems IRIS [IRIS for UNIX (Ubuntu Server LTS for x86-64 Containers)
 2020.1.3 (Build 521U) Wed Mar 16 2022 15:24:10 EDT]
    in /usr/irissys/bin/
    with mgr: /usr/irissys/mgr
    with wij: /usr/irissys/mgr/IRIS.WIJ
    from: /usr/irissys/mgr/
  OS=[Linux], version=[#1 SMP Thu Mar 17 17:08:06 UTC 2022], release=[5.10.104-linuxk
it], machine=[x86_64]
  nodename=[6759571bb2f5].
  numasyncwijbuf: 0, swdwrtmax: 0, wijdirectio: off, synctype: 3
  System Initialized.
07/10/22-10:47:21:728 (363) 0 [WriteDaemon.Started] Write daemon started.
```

Supposed to be stable version 2021.1, fails to start and requires --check-caps false

```
$ docker run --rm containers.intersystems.com/intersystems/iris:2021.1.2.338.0
[INFO] Starting ISC Agent with command "/home/irisowner/irissys/startISCAgent.sh 2188
"
Writing status to file: /home/irisowner/irissys/iscagent.status
Reading configuration from file: /home/irisowner/irissys/iscagent.conf
ISCAgent[15]: Starting
ISCAgent[17]: Starting ApplicationServer on *:2188
[ERROR] Required Linux capability cap_setuid is missing.
[ERROR] Required Linux capability cap_dac_override is missing.
[ERROR] Required Linux capability cap_fowner is missing.
[ERROR] Required Linux capability cap_setgid is missing.
[ERROR] Required Linux capability cap_kill is missing.
[FATAL] Your IRIS container is missing one or more required Linux capabilities.
```

The same version with the flag added

```
$
docker run --rm containers.intersystems.com/intersystems/iris:2021.1.2.338.0 --check-
caps false
[INFO] Starting ISC Agent with command "/home/irisowner/irissys/startISCAgent.sh 2188
"
Writing status to file: /home/irisowner/irissys/iscagent.status
Reading configuration from file: /home/irisowner/irissys/iscagent.conf
ISCAgent[15]: Starting
ISCAgent[17]: Starting ApplicationServer on *:2188
[INFO] Starting InterSystems IRIS instance IRIS...
[INFO] This copy of InterSystems IRIS has been licensed for use exclusively by:
Local license key file not found.
Copyright (c) 1986-2022 by InterSystems Corporation
Any other use is a violation of your license agreement 1 alert(s) during startup. See
```

```
 messages.log for details.
Starting IRIS 07/10/22-11:10:22:596 (389) 0 [Generic.Event] Global buffer setting req
uires attention.  Auto-selected 25% of total memory.
07/10/22-11:10:22:674 (389) 0 [Generic.Event] Allocated 2413MB shared memory: 1965MB
global buffers, 196MB routine buffers
07/10/22-11:10:22:675 (389) 0 [Crypto.IntelSandyBridgeAESNI] Intel Sandy Bridge AES-
NI instructions detected.
07/10/22-11:10:22:859 (389) 0 [WriteDaemon.UsingWIJFile] Using WIJ file: /usr/irissys
/mgr/IRIS.WIJ
07/10/22-11:10:22:859 (389) 0 [WriteDaemon.CreatingNewWIJ] Creating a new WIJ file
07/10/22-11:10:23:247 (389) 0 [WriteDaemon.CreatedNewWIJ] New WIJ file created
07/10/22-11:10:23:250 (389) 0 [Generic.Event]
Startup of InterSystems IRIS [IRIS for UNIX (Ubuntu Server LTS for x86-64 Containers)
 2021.1.2 (Build 338U) Tue Apr 12 2022 12:05:33 EDT]
    in /usr/irissys/bin/
    with mgr: /usr/irissys/mgr
    with wij: /usr/irissys/mgr/IRIS.WIJ
    from: /usr/irissys/mgr/
  OS=[Linux], version=[#1 SMP Thu Mar 17 17:08:06 UTC 2022], release=[5.10.104-linuxk
it], machine=[x86_64]
  nodename=[7f3de2512e2a].
  numasyncwijbuf: 8, wdwrite_asyncio_max: 64, wijdirectio: on, synctype: 3
  System Initialized.
07/10/22-11:10:23:280 (390) 0 [WriteDaemon.Started] Write daemon started.
```

So, let's check the next general availability release, the latest one, 2022.1, fails again

```
$ docker run containers.intersystems.com/intersystems/iris:2022.1.0.209.0
[INFO] Executing command /home/irisowner/irissys/startISCAgent.sh 2188...
[INFO] Writing status to file: /home/irisowner/irissys/iscagent.status
Reading configuration from file: /home/irisowner/irissys/iscagent.conf
ISCAgent[17]: Starting
ISCAgent[19]: Starting ApplicationServer on *:2188
[INFO] ...executed command /home/irisowner/irissys/startISCAgent.sh 2188
[ERROR] Required Linux capability cap_setuid is missing.
[ERROR] Required Linux capability cap_dac_override is missing.
[ERROR] Required Linux capability cap_fowner is missing.
[ERROR] Required Linux capability cap_setgid is missing.
[ERROR] Required Linux capability cap_kill is missing.
[FATAL] Your IRIS container is missing one or more required Linux capabilities.
```

And recently was introduced the new version for Developer Preview, 2022.2, let's start it with the flag --check-caps, and it does not know this flag anymore and fails to start.

```
$ docker run containers.intersystems.com/intersystems/iris:2022.2.0.270.0 --check-
caps false
PARSE ERROR: Argument: --check-caps
            Couldn't find match for argument Brief USAGE:
   /iris-main  [--ISCAgentPort <integer>] [--ISCAgent <bool>] [-k <license
               key>] [-L <<licenseID> <host1>,<port1>[,<dir1>] [<host2>
               ,<port2>[,<dir2>]]>] [-p <password file>] [-t <command>] [-c
               <command>] [-e <command>] [-a <command>] [-b <command>] [-l
               <log file>] [-s <bool>] [-u <bool>] [-d <bool>] [-i
               <instance>] [--] [--version] [-h] For complete USAGE and HELP type:
   /iris-main --help
```

It requires too much attention, for developers, to check all the versions.

Healthcheck

Docker offers a way, to check if container really alive and doing it's job or not. So, in production environment, it will be possible to replace the container with in right time, to make the downtime as less as possible.

And IRIS images is used this feature

```
$ docker inspect containers.intersystems.com/intersystems/iris:2022.1.0.209.0 -f '{{
.Config.Healthcheck }}'

{[CMD /irisHealth.sh] 1m0s 10s 1m0s 3}
```

But is it really helpful?

Let's start some IRIS image, healthcheck says it starting, yet. But it already started in fact and can be accessible.

```
$ docker ps -l
CONTAINER ID    IMAGE                                                              COMMAND
                   CREATED               STATUS
                         92343114736a    containers.intersystems.com/intersystems/iris
:2022.2.0.2
70.0   "/tini -- /iris-main…"   22 seconds ago   Up 21 seconds (health: starting)
$ docker exec -it iris iris session iris

Node: 92343114736a, Instance: IRIS

USER>
```

Few moments later, it shows healthy, ok, that's good, even that it came a bit later than expected

```
$ docker ps -l
CONTAINER ID    IMAGE                                                              COMMAND
                   CREATED               STATUS
92343114736a    containers.intersystems.com/intersystems/iris:2022.2.0.270.0    "/tini
-- /iris-main…"   3 minutes ago   Up 3 minutes (healthy)
```

Let's just restart it, wait a minute and check the health again

```
$ docker restart iris

$ docker ps -l
CONTAINER ID    IMAGE                                                              COMMAND
                   CREATED               STATUS

92343114736a    containers.intersystems.com/intersystems/iris:2022.2.0.270.0    "/tini
-- /iris-main…"   6 minutes ago   Up About a minute (health: starting)    a

$ docker ps -l
CONTAINER ID    IMAGE                                                              COMMAND
                   CREATED               STATUS
92343114736a    containers.intersystems.com/intersystems/iris:2022.2.0.270.0    "/tini
-- /iris-main…"   7 minutes ago   Up 3 minutes (health: starting)
Still starting? even when IRIS already accessable
```

Finally, after 5 minutes, we got changes, but, what, it says unhealthy, how come?

```
$ docker ps -l
CONTAINER ID   IMAGE                                                          COMMAND
                    CREATED           STATUS
92343114736a   containers.intersystems.com/intersystems/iris:2022.2.0.270.0   "/tini
-- /iris-main…"   11 minutes ago   Up 5 minutes (unhealthy)
```

How to deal with it in production? Is not it just a waste of resources? From what I see, that healthcheck makes no sense at all.

And that's it for now

I'd like to hear notes from InterSystems customers, which may have struggles like me as well, and from InterSystems and hope to see positive changes in future.

#Containerization #Continuous Delivery #Continuous Integration #Docker #InterSystems IRIS

Source URL:https://community.intersystems.com/post/any-stability-iris-docker-images