

記事

[Toshihiko Minamoto](#) · 2020年12月25日 8m read

Docker for Windows で InterSystems IRIS コンテナを使用する

InterSystems は自社が提供する InterSystems IRIS の Docker イメージについて、Linux 環境での使用のみをサポートしています。Docker for Windows は Linux プラットフォームのようにコンテナをネイティブプロセスとして実行するのではなく、Windows の仮想機能である Hyper-V で実行される Linux VM を作成してコンテナをホストします。このような追加イヤーによって複雑度が増しているため、InterSystems は現時点で Docker for Windows をサポートすることができません。

ただし、テストやその他の特殊な目的のために InterSystems の IRIS ベースのコンテナを Docker for Windows 環境で実行する必要があることは私たちが理解しています。この記事では、InterSystems が提供するコンテナイメージの使用に関して InterSystems が認識している Docker for Windows と Docker for Linux の違いについて説明します。その他にも予期しない問題が発生する可能性があります。Windows プラットフォームで InterSystems IRIS のイメージコンテナを使用する場合は、参考のために [Docker ドキュメント](#) (特に「[Getting started with Docker for Windows](#)」) にアクセスできることを確認してください。

Docker for Windows 環境のコンテナで外部永続ストレージを操作する場合は、Windows および Linux のファイルシステムとファイル操作の両方が関係してくるため、ここでは主にストレージに関する違いを説明しています。

InterSystems が提供するイメージを使用して InterSystems IRIS を Docker コンテナで実行する場合の一般的な情報については、「[Running InterSystems IRIS in Docker Containers](#)」と「[First Look: InterSystems IRIS in Docker Containers](#)」を参照してください。

ディスクドライブを共有する

Windows では Docker が存在するディスクドライブを共有し、Docker に操作対象となるすべてのストレージへのアクセスを許可する必要があります。1 台以上のドライブを共有するには、次の手順に従ってください(この手順は前の項目の手順と組み合わせることができます)。

1. システムトレイの Docker アイコンを右クリックし、[[Settings ...](#)] を選択します。
2. [Shared Drives] タブを選択し、ストレージが配置されているドライブを選択してから [[Apply](#)] をクリックします。ドライブがすでに選択されている場合は(デフォルトでは C ドライブが選択されています)、チェックボックスをオフにして [[Apply](#)] をクリックし、ドライブを選択してから [[Apply](#)] をもう一度クリックします。
3. プロンプトが表示されたら、ログイン資格情報を入力します。
4. 変更を適用すると、Docker が自動的に再起動します。再起動しなかった場合はシステムトレイの Docker アイコンを右クリックし、[[Restart](#)] を選択します。

コンテナ内で外部ファイルをコピーする

Docker を使用する場合は、一般的には外部ファイルシステム内のディレクトリをコンテナ内のボリュームとしてマウントし、それをコンテナ内のソフトウェアが必要とするすべての外部ファイル用の場所として使用するのが便利です。例えば、あるボリュームをマウントし、`iris-main` プログラムの `--key` オプションにパスワード変更スクリプトがそれぞれアクセスする外部ディレクトリにある InterSystems IRIS のライセンスキー(`iris.key`) や所定の InterSystems IRIS のパスワードを含むファイルを配置できます(「[Running InterSystems IRIS in Containers](#)」の「[The iris-main Program](#)」と「[Changing the InterSystems IRIS Password](#)」を参照してください)。ただし、Docker for Windows ではファイル操作アクセス許可の違いにより、マウントされた外部ボリューム上のファイルがコンテナ内のプログラムによって適切に使用されない場合があります。多くの場合はプログラムでコンテナ内のファイルをコピー

してからこのコマンドを使用することで、アクセス許可の問題を解決できます。

例えば、iris-main --before オプションはコンテナ内の InterSystems IRIS インスタンスのパスワードを変更するためによく使用されます。以下に例を示します。

```
--before "changePassword.sh /external/password.txt"
```

この方法を使って Windows で意図した通りにパスワードを変更できない場合は、以下を試してください。

```
--before "cp /external/password.txt /external/password_copied.txt && \
changePassword.sh /external/password_copied.txt"
```

名前付きボリュームを使用する

動的に変更されるファイルが存在する場合、個別にマウントしてすべてのファイルをコピーできなくても、コンテナ内に Windows ファイルシステムを直接マウントする問題が発生する可能性があります。InterSystems IRIS の場合、これは特にインスタンス固有データを永続的に保存するための Durable %SYS 機能(「Running InterSystems IRIS in Containers」の「[Durable %SYS for Persistent Instance Data](#)」を参照してください)とジャーナルファイルの両方に当てはまります。この問題は名前付きボリュームをマウントすることで解決できます。名前付きボリュームは、システム上のコンテナをホストしている Linux VM のファイルシステムにマウントポイントを持つストレージボリュームです。VM はファイルシステムベースであるため、このようなボリュームの内容は Docker やシステムがダウンした場合で VM の他の部分と一緒に Windows ディスクに保存されます。

例えば InterSystems IRIS コンテナを実行中に Durable %SYS を有効にする場合、一般的には外部ボリュームをマウントし、--env オプションを使用して ISC_DATA_DIRECTORY 環境変数をそのボリュームの場所に設定します。以下に例を示します。

```
docker run ... \
--volume /nethome/pmartinez/iris_external:/external \
--env ISC_DATA_DIRECTORY=/external/durable/
```

????? Docker for Windows ??????????????docker volume create ??????????????????????Durable %SYS ?????????????????????? ?????Linux ??????????ISC_DATA_DIRECTORY ??????? Durable %SYS ??????????? /irissys ?????????????????? ???????Windows ??????????????????????

```
docker volume create durable
docker run ... \
--volume durable:/durable \
--env ISC_DATA_DIRECTORY=/durable/irissys/
```

インスタンスのジャーナルファイルにこの方法を使用するには、上記の Durable %SYS の例のように名前付きボリュームを作成してマウントし、任意の構成法(^JOURNAL オプション、管理ポータル)の [Journal Settings] ページ、または iris.cpf ファイルを使用して現在および代替のジャーナルディレクトリを名前付きボリュームの場所に設定します。現在のジャーナルディレクトリと代替のジャーナルディレクトリを分離するには、それぞれに対応する名前付きボリュームを作成してマウントする必要があります (この方法は **十分に検証されておらず**、ジャーナルファイルが危険にさらされる可能性がありますのでご注意ください)。

VM 内の名前付きボリュームにある Linux ファイルシステムのマウントポイントを検出するには、次のように docker volume inspect コマンドを使用します。

```
docker volume inspect durable_data
[
  {
    "CreatedAt": "2018-05-04T12:11:54Z",
```

