

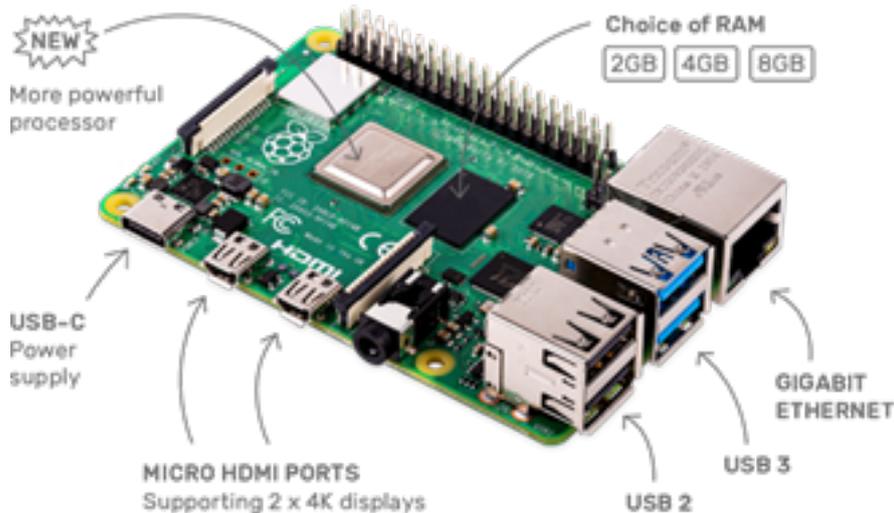
記事

[Toshihiko Minamoto](#) · 2022年4月19日 9m read

## Raspberry Pi Raspberry で InterSystems Iris FHIRserver を実行する

### Raspberry を FHIRserver として実行する

一年ほど前、



Raspberry Pi での HAPI FHIRserver のインストールに関する記事を書きました。当時、FHIR 標準の基本しか知らず、FHIR サーバーの背後にあるテクノロジーや Raspberry についてほとんど知りませんでした。試して、失敗して、諦めて、もう一度試すことで、たくさんのことを学びました。

1 年以上が経ち、いまだに「忍耐力に長けるアマチュア」ではありますが、その間、FHIR のフル環境を Raspberry で構築し、HTML/JavaScript/jQuery/Bootstrap によるアプリケーションや PHP アプリケーションを多数作成してきました。Raspberry の認証サーバーや承認サーバーについてもいくらか経験し、SMART on FHIRを使った実験も行いました。

最後の機能しなかったバージョンと、明らかに私が作成した成功バージョンの違いが判らないまま動作することもありましたが、作業や実験のほとんどは動作しました。今でも驚いています。振り返ると、システム開発について適切に学習しておくべきだったと思います（私は山林学専攻で卒業し、現在はその分野を中心にボランティア活動を行っています。まあ、本題からは外れてしましますが）。

Raspberry Pi は実験には十分ですが、本稼働には適していないことに注意してください。したがって、信頼性、継続性、セキュリティ、プライバシーといった要素が必要である、または期待されているヘルスケアなどの分野で、Raspberry を本稼働システムとして使用しないでください。実験目的では完璧で安価なコンピュータであり、幾度も私の期待を超えていますが、「産業耐久性」に欠けています。

### InterSystems FHIRserver

入手しやすさと、非常に早い段階で結果を得られたことを理由に、当時 HAPI FHIR を選択しました。以前の同僚で、現在では InterSystems に勤めている人から、InterSystems の FHIRserver でも試用することをすでに提案されていました。

InterSystems については非常によく知っており（というより、「i.know」と言うべきでしょうか）、面白そうに思いましたが、当時、一年以上前は、ARM プロセッサ向けに適したバージョンの Iris は存在していませんでした（Raspberry は ARM で動作します）。それとは別に、Iris の Community バージョンは Docker コンテナとして配布されていましたが、私には Docker に関する知識がまったくありませんでした。

## Raspbian から Ubuntu へ

それから 1 年が過ぎました。現在では、さらに強力になった新しい Raspberry Pi が登場しています（在庫のある限り...）。新しい Raspberry 4 は、Raspberry 3 と同様に、64 ビット Unix をサポートしています。IRIS Community バージョンは ARM にも対応しており、コンテナと Docker についてもいくらかさらに学習しました。

こういった改善すべてを基に、昨年の年末にオランダで施行されたわけのわからない「ロックダウン」が、IRIS Community バージョンを Raspberry Pi で実行させる新しい実験を開始するチャンスとなりました。

振り返ると、これは非常に簡単なことでした。実際、注意すべき点は 3 つしかありません。

1. 64 ビットをサポートする Raspberry が必要です。つまり、Raspberry 3 または 4 が必要となります。
2. (64 ビット) Raspbian ではうまくいきませんでした（ちなみに非常に新しいものです）。しかし、64 ビット Ubuntu では成功しました。
3. InterSystems FHIRserver は、HAPI FHIRserver よりも FHIR 呼び出しに関して厳格さが増しているようです。  
これについては別の機会に説明しますが、これを悪いことだとは思っていません。（ヘルス）ケア環境における中央リポジトリであるため、標準について厳格である必要があるためです。そうでなければ、「ガベージイン・ガベージアウト」になってしまうでしょう。

それでも、HAPI FHIRserver でうまく実行していたアプリケーションが IRIS では失敗することを理解するには、いくらか時間がかかりました。ネタバレ注意：問題は私のアプリケーションにありました。

それにしても、Raspberry Pi のような、100 ユーロもかからない小さなコンピュータで、InterSystems IRIS フルプラットフォームが実行し、管理ポータル、Ensemble、Caché、そして FHIRserver が備わっているのを目の当たりにするのは、楽しく本当に価値以上の価値があります。

以降では、そこに至るまでに私が行ったことを、順を追って説明します。Ubuntu の歴史、Docker の背後の意ある哲学など、長々とした話にはなりません。そちらの方に関心のある方は、インターネットをご覧ください。ここでは「楽しく進められるフロー」に焦点を当て、どこで初めて間違ったのかについて言及する場合にのみ、注釈したいと思います。

## InterSystems FHIRserver のインストール

### フェーズ 1: Ubuntu 64 ビット

1. SD カードをフォーマットし、そこに ARM プロセッサ用 64 ビット Ubuntu を配置します。私の場合、圧縮バージョンの名前は「ubuntu-20.04.3-preinstalled-server-arm64+raspi.img.xz」でした。これは <http://cdimage.ubuntu.com> にあります。（もちろん、SD カードに置く前に、.img ファイルを解凍する必要があります！）

RaspberryPi-imager は、誤ったバージョンを提供するため、使用しないでください。私は Win32DiskImager を使用しました。

2. 新しいイメージで Raspberry を開始します。ubuntu-account のパスワードを変更します（そして記憶しましょう！）。あくまでも ubuntu-account のパスワードです。
3. この時点で Raspberry は Ubuntu で実行していますが、まだ準備はできていません。

以下を見てください。

この Ubuntu-version はまだ 32 ビットになっています！

今すぐこれを直しましょう。

1.  
Pi にログインします。
2.  
`sudo rpi-update`
3.  
`sudo reboot now`
4.  
`sudo copy /boot/config.txt /boot/config-ok.txt` ( 検出したエラーを逃してしまった場合に備えます )
5.  
`sudo nano /boot/config.txt` [pi4] の下に次のテキストを追加します: `arm64bit=1`
6.  
`boot/config.txt` を保存します。 ( nano で: `<ctrl>O` )
7.  
nano を終了します。 ( `<ctrl>X` )
8.  
`sudo reboot now`

これで、64 ビットで実行するようになりました。自分で確認してみましょう:

Pi のIP アドレスをメモし ( `sudo hostname -I`。I は大文字のアイです )、Ubuntu では SSH はデフォルトで有効になっているため、別の場所に保管しておきます。

## フェーズ 2: Docker

1.  
`sudo curl -fsSL https://get.docker.com -o get-docker.sh`
2.  
`sudo sh get-docker.sh`

しばらくすると、これは、「non-privileged user」として使用することに関する Docker の詳細な通知が得られます。

3.  
`sudo usermod -aG docker $USER`
4.  
ログオフしてから、もう一度ログインします。
5.  
`docker container run hello-world`

Docker はシステム上の「hello-world」イメージを探すようになりました。見つからない場合は、Docker リポジトリからダウンロードします。その後、Docker はイメージを Pi 上のコンテナに配置し、「hello-world」イメージを開始します。

Docker の別のテキストブロックで、「Hello from Docker」のウェルカムメッセージと共に確認できます。

## フェーズ 3: InterSystems の IRIS FHIRserver の Docker イメージ

1. 以下のコマンドを一度に発行します。

```
sudo docker run --name my-iris -d --publish 9091:1972 --publish 9092:52773  
containers.intersystems.com/intersystems/irishealth-community-arm64
```

多数のイベントが順に発生します。

- Docker は Irishealth から「community edition」を取得し、それをローカルの Docker リポジトリに配置します。
- Docker は「my-iris」という Docker コンテナで Community エディションを起動します。
- Docker はコンテナで実行している IRIS のポート 1972 を Raspberry のポート 9091 にマッピングします。
- 同じパターンに従って、IRIS のポート 52773 (コンテナ内) は Raspberry のポート 9092 になります。

以下のようにすると、何が起きたのかを確認できます。

2. `sudo docker container ls` ただし、これを確認するには別のはるかに印象的な方法があります。

## フェーズ 4: InterSystems IRIS for Health Fhirserver の起動

ネットワークで Web ブラウザを起動します。英語版 PC の場合は次の場所に移動してください。

<Raspberry の IP アドレス>:9092/csp/sys/UtilHome.csp\_

IRIShealth の管理ポータルが表示されます。このポータルは、自動的に PC の言語設定に調整されます。

標準アカウントは SYSTEM (先頭のアンダースコアを含む) で、初回パスワードは SYS (アンダースコアはありません) です。IRIS からすぐにパスワードを変更するように求められます。

これで、次に記載されている指示に従って IRIS for Health FHIRserver を構成し、起動できるようになりました。

<https://docs.intersystems.com/irisforhealthlatest/csp/docbook/DocBook.UI...>

「Foundation」を構成して「Endpoint」(FHIR 3 または 4) を定義する必要があります。

「Endpoint」は、FHIRserver にアクセスできる URI となります。

私の場合は、<http://192.168.1.29:9092/csp/healthshare/fhironpi/fhir/r4/Patient> にある「Patient」リソースです (外部からはアクセスできません! )。

## まとめ

手間を取る価値があったでしょうか? そのとおり! それ以上の価値があります。

まず、それほどの手間はありませんでした。Raspbian

ではうまくいきませんでした。それは簡単に解決しました。そして、IRIS-for-health プラットフォームを Raspberry Pi のようなデバイスで実行できるのは素晴らしいことです。

ダッシュボード (DeepSee)、データ変換ツール、Ensemble (Enterprise Service Bus とその他多数のツール)、CSP ページ、および

Caché (多次元データベース) すべてを実験できるようですし、自然言語処理の I.know も管理ポータルに表示されています。IRIS-for-health プラットフォームには、FHIRserver 単体よりもはるかに多くの機能が備わっており、1つの記事ではとても説明しきれません。

私個人としては、ほぼリリース直後から Ensemble

を使用してきましたし(アーキテクトとして)、それとは別に、IRIS-for-health の起源は 20 世紀後半だったため、思い入れがあります。ヘルスケアアプリケーション向け MUMPS 開発環境 (MUMPS/マンブス:

マサチューセッツ総合病院ユーティリティマルチプログラミングシステム) は、ほとんどが Digital Corp の Vax コンピュータシステムで実行されていたのですが、私の IT キャリアが始まったのも同時期なのです。

MUMPS、Caché、Ensemble、そして今日には IRIS-for-Health

プラットフォームの安定性は非常に高く、必要なサポートとメンテナンスは比較的控え目です。InterSystems 製品のドキュメントは広範に用意されており、簡単に見つけることができます。

IRIS と FHIRserver での実験をさらに続けていく予定です。最低でも後 1 つ、FHIR 標準への厳格な準拠に対する私の「奮闘」に関する記事を絶対に公開いたします!

健康を大切に、思考を繰り返して、実験し続けましょう!

Bob

[#FHIR](#) [#コンテナ化](#) [#初心者](#) [#InterSystems IRIS](#)

---

ソースURL:<https://jp.community.intersystems.com/post/raspberry-pi-raspberry-%E3%81%A7-intersystems-iris-fhirserver-%E3%82%92%E5%AE%9F%E8%A1%8C%E3%81%99%E3%82%8B>