

Hail-jp 「The 2nd」 Hands on seminars

第1回 Hail をインストールしてみよう！

日時
2022年 1月 27 日

会場
オンライン開催

mac でも windows でも Linux でも OK
全3回開催のオンラインセミナーで、インタラクティブでスケーラブルな遺伝統計研究のためのライブラリ Hail をお手元の環境で試せるようになります

第1回 Hail をインストールしてみよう！(1/27 開催)
第2回 GWAS tutorial を動かしてみよう！(2月下旬開催予定)
第3回 Hail を使ってポリジェニックスコアを計算してみよう！(3月下旬開催予定)

お問い合わせは hail-jp-staff@googlegroups.com まで

会場等協力：
GAJ
Genomic Analytics Japan
NABE International
株式会社 ナベインターナショナル



Hail-jp ハンズオンセミナー

「Hailをインストールしてみよう！」

14:00- m1 mac

14:30- intel mac

15:00- Windows slot#1

15:30- Windows slot#2

16:00- Linux

解説：内田智博



今日の目標

自分の環境で動くhailを手に入れる！

できるだけプレーンな状態のマシンにインストールをデモします。

この組み合わせなら「動くはず」と「手順」を手に入れてください。



今日の流れ

1. (DEMO) 実機にインストール

並行してStep by Stepで実行していただきます

(適宜使用するソフトウェアのバージョンやポイントを解説します。)

2. Dockerを使ったHailを紹介

3. 質疑応答



1. (DEMO) 実機にインストール

並行してStep by Stepで実行していただきます

注意! まだほとんど使用していないmacosへのインストールを前提としています。

使い込んでいる場合はすでにjavaやpythonがあるかもしれません。
適宜読み替えていただくな、後ほどslackでご相談ください。



1. DEMO 実機にインストール

Hailの動作要件

Java 8 or 11

(最近11がサポートされました。今回は8で行きます)

Azulu ?

Install Hail on Mac OS X

- Install Java 8 or Java 11. We recommend using a [packaged installation from Azul](#) (make sure the OS version and architecture match your system) or using [Homebrew](#):

```
brew cask install adoptopenjdk8  
brew install --cask adoptopenjdk8
```

Homebrew

- Install Python 3.7 or later. We recommend [Miniconda](#).
- Open Terminal.app and execute `pip install hail`.
- [Run your first Hail query!](#)

Miniconda

Python

Hail

Jupyterlabも

インストールは簡単ですが、使用するソフトウェアやバージョンの選択をしていく必要があります。

<https://hail.is/docs/0.2/install/macosx.html>



先に結論: 今日構築する内容

1. DEMO 実機にインストール

JAVA8をインストール

パッケージ管理ツール miniforgeを使って
Python3.9をインストール
そのPythonに、
HailとJupyterlabをインストール

Azule zulu Java 8
(8u322)

最新のHail

最新のJupyterlab

Python 3.9以上

Miniforge(conda)

MacOS

Apple Mac ハードウェア (Intel)

hail.jp



先に結論: 今日構築する内容

1. DEMO 実機にインストール

Azule zule Java 8
(8u322)

<https://www.azul.com>よりインストーラーを取得
GUIでインストールする

Miniforge(conda)

```
#インストーラーを取得  
curl https://github.com/conda-forge/miniforge/releases/latest/download/Miniforge3-MacOSX-x86_64.sh -OL  
#インストーラーを実行  
bash Miniforge3-MacOSX-x86_64.sh  
  
#miniforgeで、python-3.9.7がインストールされる
```

Python 3.9以上

最新のHail

```
 #(terminalを開き直すなどしてconda環境を有効にしてから)  
pip3 install hail jupyterlab
```

最新のJupyterlab

動作確認

```
# jupyterlabの起動  
jupyter-lab --ip=0.0.0.0 --no-browser --NotebookApp.token="
```

ブラウザにhttp://localhost:8888 と打ち込めばjupyterlabを開けるようになります



1. DEMO 実機にインストール

1. 実機に直接インストールをデモしつつ、 Step by Stepで実行していただきます

それでは始めてみましょう！

今日打つコマンド一覧です。先にチャット欄に貼り付けます。

```
#java8の動作確認
```

```
java -version
```

```
#インストーラーを取得
```

```
curl https://github.com/conda-forge/miniforge/releases/latest/download/Miniforge3-MacOSX-x86_64.sh -OL
```

```
#インストーラーを実行
```

```
bash Miniforge3-MacOSX-x86_64.sh
```

```
#{terminalを開き直すなどしてconda環境を有効にしてから)
```

```
#miniforgeでインストールされたpythonのバージョンを確認します。
```

```
python --version
```

```
pip3 install hail jupyterlab
```

```
# jupyterlabの起動
```

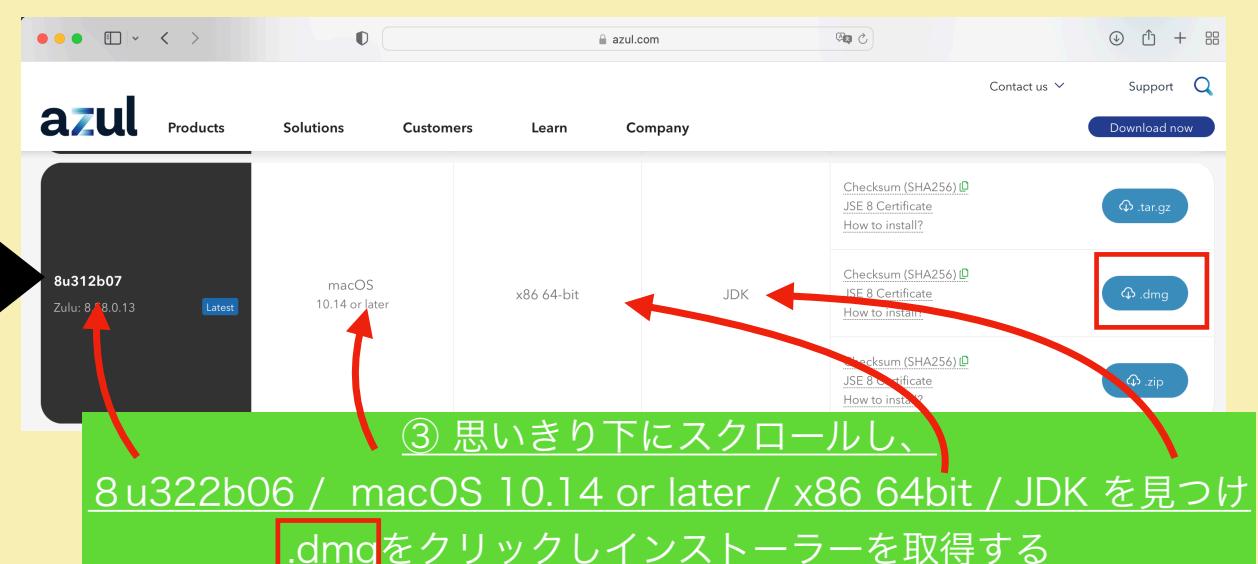
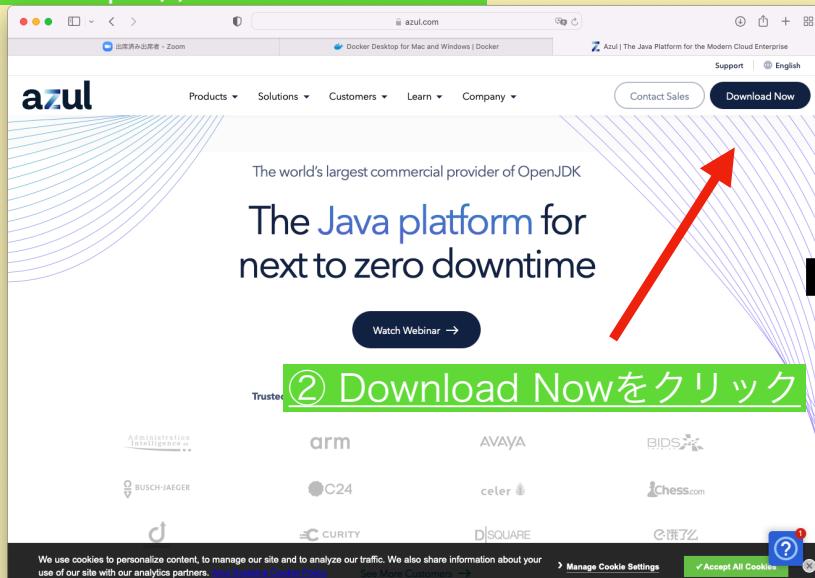
```
jupyter-lab --ip=0.0.0.0 --no-browser --NotebookApp.token="
```



java8のインストール

1. DEMO 実機にインストール

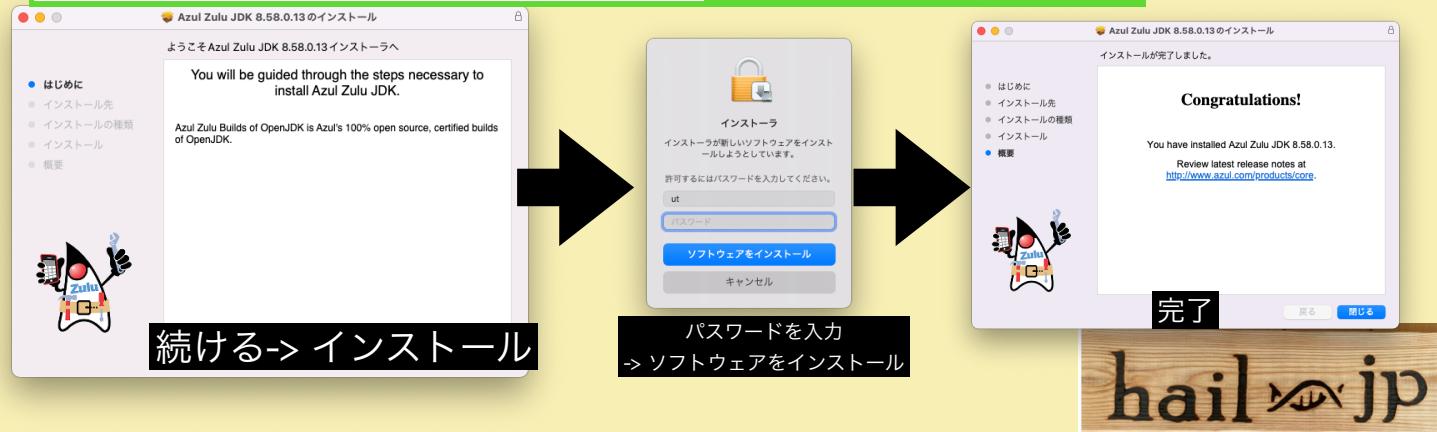
① https://azul.com を開く



④ダウンロードしたインストーラーを実行



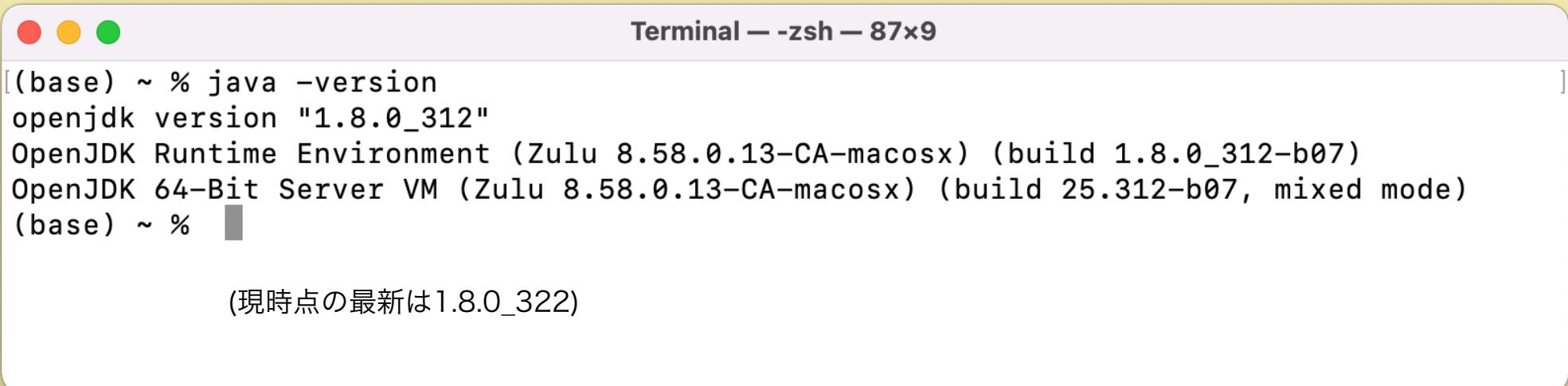
⑤指示に従いインストールをすすめます



ターミナルを開いて

java -version

を実行。次のような出力が得られればOK!



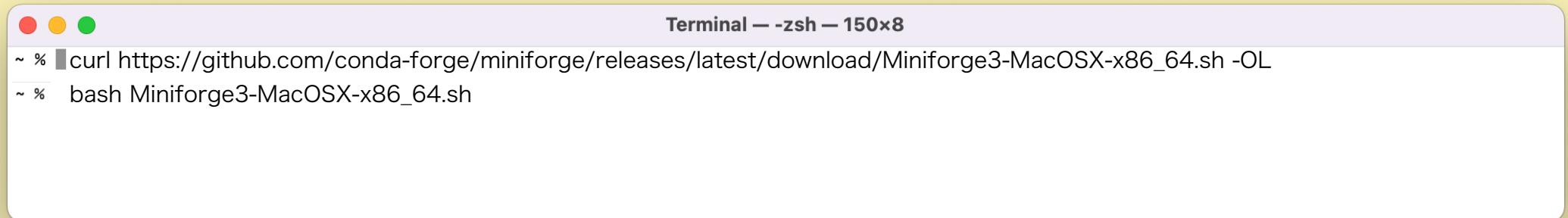
The screenshot shows a macOS Terminal window titled "Terminal — -zsh — 87x9". The window contains the following text:

```
(base) ~ % java -version
openjdk version "1.8.0_312"
OpenJDK Runtime Environment (Zulu 8.58.0.13-CA-macosx) (build 1.8.0_312-b07)
OpenJDK 64-Bit Server VM (Zulu 8.58.0.13-CA-macosx) (build 25.312-b07, mixed mode)
(base) ~ %
```

Below the terminal window, the text "(現時点の最新は1.8.0_322)" is displayed.

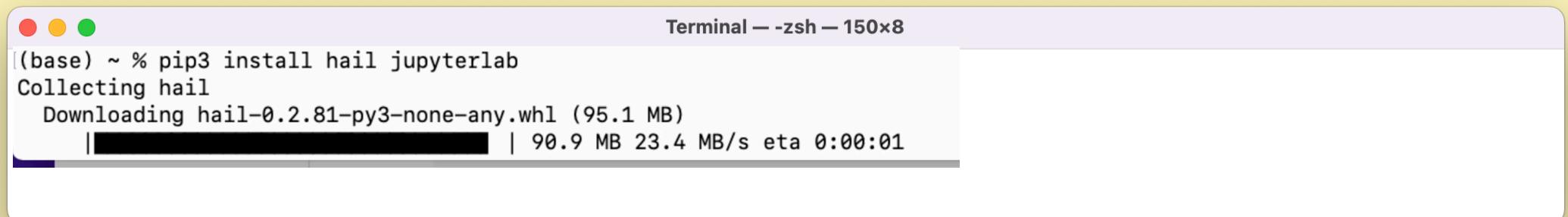


- miniforgeのインストール (約1分)



```
curl https://github.com/conda-forge/miniforge/releases/latest/download/Miniforge3-MacOSX-x86_64.sh -OL  
bash Miniforge3-MacOSX-x86_64.sh
```

- hailとjupyterlabのインストール (約4分)



```
(base) ~ % pip3 install hail jupyterlab  
Collecting hail  
  Downloading hail-0.2.81-py3-none-any.whl (95.1 MB)  
 |██████████| 90.9 MB 23.4 MB/s eta 0:00:01
```

この間に、javaやminiforge、使うバージョンなどについて解説します



1. DEMO 実機にインストール

Javaのバージョンを整理

Java 8 とは？？

Stable LTSとあるのが安定版

Javaの最新バージョンは18

Javaの最新の安定版は17

今回使うのは8

明示的に8を指定してインストールしないと、17が入ってしまい
Hailが動かない場合があるので注意！



| | Supported | Ready for Production | Role | Comment |
|--------|-----------|----------------------|--------------------------------------|---|
| JDK 8 | + | ✓ | Stable LTS | Available as non-mainline 8u backport. Check with your vendor for availability. See known vendors list below. |
| JDK 9 | ! | ✗ | | Discontinued, migrate to 11/17 as soon as possible . |
| JDK 10 | ! | ✗ | | Discontinued, migrate to 11/17 as soon as possible . |
| JDK 11 | ✓ | ✓ | Stable LTS | In mainline OpenJDK 11u since 11.0.9 . Requires opt-in during build time, check with your vendor for availability. See known vendors list below. |
| JDK 12 | ! | ✗ | | Discontinued, migrate to 17 as soon as possible . |
| JDK 13 | ! | ! | | Discontinued, migrate to 17 |
| JDK 14 | ! | ! | OpenJDK(オープンソースのJava)のホームページより | |
| JDK 15 | ! | ! | Documentation, bug reports, etc. ... | |
| JDK 16 | ! | ! | | Discontinued, migrate to 17. |
| JDK 17 | ✓ | ✓ | Stable LTS | In mainline OpenJDK builds. |
| JDK 18 | 💡 | 💡 | Dev/Test | In mainline OpenJDK builds. |

<https://wiki.openjdk.java.net/display/shenandoah/Main>



Java 8のRelease

1. DEMO 実機にインストール

Java 8

セキュリティやbugfixのために
できるだけ最新を使いたい

本日時点で、
Java 8 の最新は 8u322

Java 8のrelease 8u322
を使う!



<https://wiki.openjdk.java.net/display/jdk8u/Main>

Multi-Language VM

OpenJDK Wiki

Dashboard > JDK 8u > Main

Main

Created by Iris Clark, last modified by Andrew Hughes on Dec 28, 2021

Welcome to OpenJDK 8 Updates!

OpenJDK 8 updates are a [separate project](#) of OpenJDK. [Andrew Haley](#) serves as the Project Lead. The list of Reviewers, Committee

Maintainers

- Andrew Haley
- Andrew Hughes
- Severin Gehwolf

Releases

Latest GA release: 8u312

Latest Generally Available (GA) binary releases of the OpenJDK jdk8u project are available at: <https://adoptopenjdk.net/upstream.html>

Latest Early Access (EA) binary releases of the OpenJDK jdk8u project are available at: <https://adoptopenjdk.net/upstream.html?var=ea>

Most recent and past release details:

- 8u312-b07 (GA), October 19th 2021 [Release] [Tag] [Binaries]
- 8u302-b08 (GA), July 20th 2021 [Release] [Tag] [Binaries]
- 8u292-b10 (GA), April 20th 2021 [Release] [Tag] [Binaries]
- 8u282-b08 (GA), January 19th 2021 [Release] [Tag] [Binaries]
- 8u275-b01 (GA), November 5th 2020 [Release] [Tag] [Binaries]
- 8u272-b10 (GA), October 20th 2020 [Release] [Tag] [Binaries]



利用可能なJavaのビルドの種類と選択

Java 8

Azulu ?

複数のベンダーが
OpenJDKをbuildして
パッケージを制作して提供。
私たちは、それから選択し、
インストールして使用しています。

選択のポイント!!

OSによって現実的な選択が異なります

Oracle Javaは利用条件要確認

macで無償利用なら、
AdoptOpenJDKや
Azul Zuluが利用可能。

今回はAzul ZuluのJavaを使います

| ビルド | LTS | パーミッシブ | TCK | 商用サポート |
|---|-----|-----------|-----|------------------|
| AdoptOpenJDK | Yes | Yes | No | オプション(IBM) |
| Amazon Corretto | Yes | Yes | Yes | オプション(AWS上) |
| Azul Zulu Builds of OpenJDK | Yes | Yes | Yes | オプション |
| BellSoft Liberica JDK | Yes | Yes | Yes | オプション |
| Eclipse Temurin | Yes | Yes | Yes | オプション(Azul, IBM) |
| Microsoft Build of OpenJDK | Yes | Yes | Yes | オプション(Azure上) |
| Oracle Java SE | Yes | (バージョン次第) | Yes | Yes |
| Oracle OpenJDK | No | Yes | Yes | No |
| Red Hat OpenJDK | Yes | Yes | Yes | Yes |
| SapMachine | Yes | Yes | Yes | オプション(SAP製品) |

<https://wiki.openjdk.java.net/display/shenandoah/Main>



Miniconda ??

1. DEMO 実機にインストール

Install Hail on Mac OS X

- Install Java 8 or Java 11. We recommend using a [packaged installation from Azul](#) (make sure the OS version and architecture match your system) or using [Homebrew](#):

```
brew cask install adoptopenjdk8  
brew install --cask adoptopenjdk8
```

- Install Python 3.7 or later. We recommend [Miniconda](#).
- Open Terminal.app and execute [`pip install hail`](#).
- [Run your first Hail query!](#)

Miniconda

ソフトウェアを簡単にインストールできる、
パッケージ管理ツールです。
特に、Python関連の管理が得意です。

Minicondaというと、右のMinicondaを指しますが、
今回は「miniforge」という別のものを代わりに
使います。



The screenshot shows the official Miniconda documentation page on [docs.conda.io](https://docs.conda.io/en/latest/miniconda.html). The page has a green header bar with the Conda logo. The main content area is titled "Miniconda" and contains information about it being a free minimal installer for conda, including Python and other packages. It lists "System requirements" and provides "Latest Miniconda Installer Links" for Windows, macOS, and Linux. A sidebar on the left includes links for Conda, Conda-build, and Miniconda, along with "Search docs".

| Platform | Name | SHA256 hash |
|----------|--------------------------------|--|
| Windows | Miniconda3 Windows 64-bit | b33797064593ab2229a8135dc69001bea85cb5 |
| | Miniconda3 Windows 32-bit | 24f438e57ff2efefce1e93858d4e90d13f585895 |
| MacOSX | Miniconda3 Mac OSX 64-bit bash | 7864e9721f43e2c7d288314cc635f5fe48234 |
| | Miniconda3 Mac OSX 64-bit pkg | 8fa371ae97218c3c805cd5f8401f40156d1586 |

<https://docs.conda.io/en/latest/miniconda.html>



Miniforgeである理由

1. DEMO 実機にインストール

標準的にはanaconda/minicondaが使われていましたが、2020年4月30日の発表で、利用が有償になるケースがでてきました。商用利用や200人以上の企業は注意が必要です。

miniforgeは無償で利用できる代替え手段のひとつです。

Miniforge

 Build miniforge passing

This repository holds a minimal installer for [Conda](#) specific to [conda-forge](#). Miniforge allows you to install the conda package manager with the following features pre-configured:

- [conda-forge set as the default \(and only\) channel.](#)
 - Packages in the base environment are obtained from the [conda-forge](#) channel.
- Optional support for PyPy in place of standard Python interpreter (aka "CPython").
- Optional support for [Mamba](#) in place of Conda.
- An emphasis on supporting various CPU architectures (x86_64, ppc64le, and aarch64 including Apple M1).

It can be compared to the [Miniconda](#) installer.

<https://github.com/conda-forge/miniforge>

Miniforgeは使用するリポジトリが無償で利用できる「conda-forge」にセットされています。



hail ✈ jp

Python

Hailの要求は3.7以上 (最近変わりました)

Intel macなら、3.7, 3.8, 3.9でもOK

M1 macだと・・3.9でないと簡単にインストールできない

miniforgeをインストールすると自動的にpython-3.9.7がインストールされるので、そのままこれを使います。

Pyrhonは3.9.7を選択



HailとJupyterlabのバージョン

1. DEMO 実機にインストール

Hail

Jupyterlab

ここまで選択を間違えなければ、これらは最新(=バージョン指定無し)でよい

Hailは最新を選択

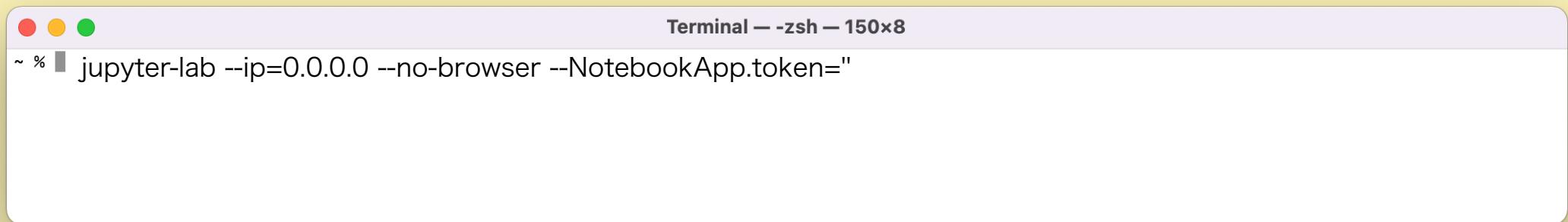
Jupyterlabも最新を選択



1. DEMO 実機にインストール

動作確認をしてみましょう

jupyterlabを起動して



```
Terminal — zsh — 150x8
~ % jupyter-lab --ip=0.0.0.0 --no-browser --NotebookApp.token="
```

ブラウザに, `http://localhost:8888` を貼り付けます

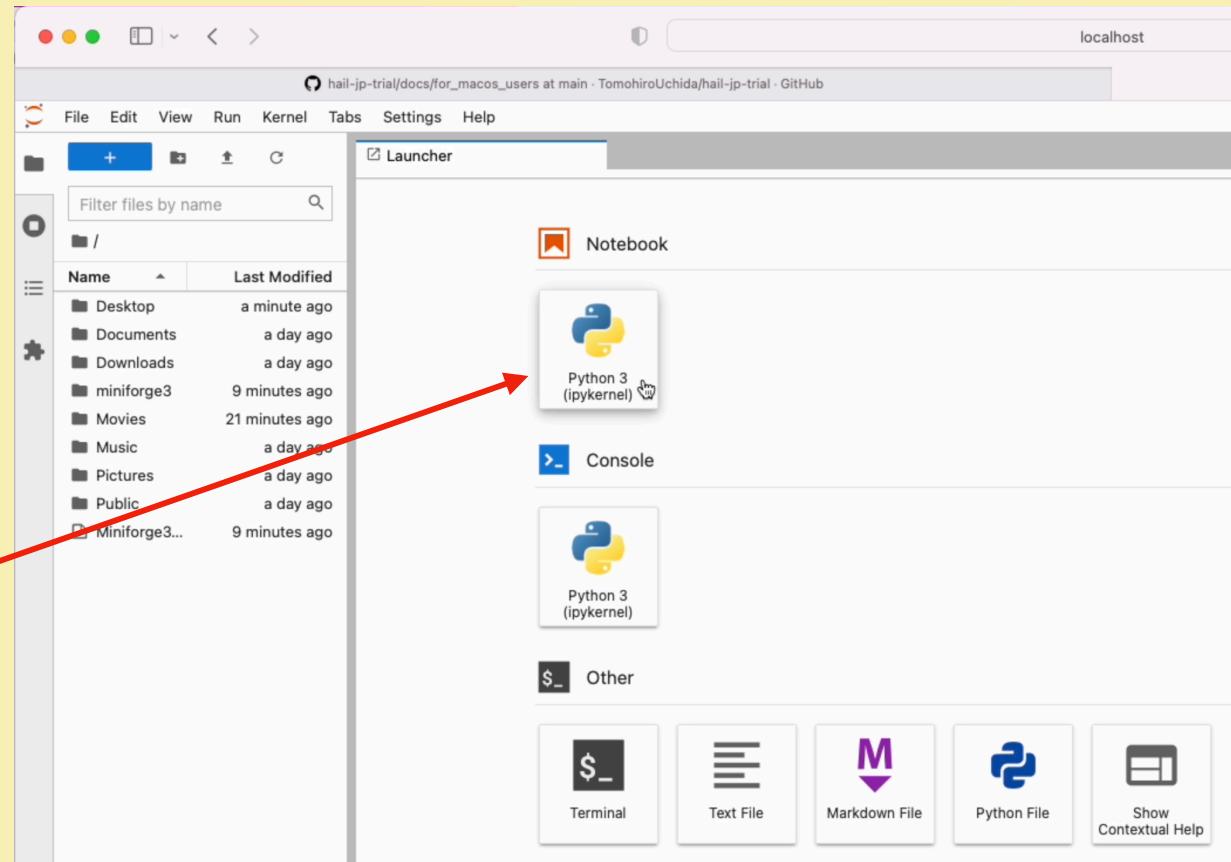


1. DEMO 実機にインストール

動作確認をしてみましょう

このような画面が出てきたらOK

Python3をクリックすると
「Python kernel」が起動します



hail.jp

1. DEMO 実機にインストール

動作確認をしてみましょう

Python kernelが起動したら、

import hail as hl
を入力して「Shift + Enter」

次に、
hl.init()
を入力して「Shift + Enter」
を押します。

このように、「Welcome to」や
Hailのステキなアスキーアートが出力されたらOKです！

The screenshot shows a Jupyter Notebook cell titled "Untitled.ipynb". The cell contains two lines of Python code: [1]: import hail as hl and [2]: hl.init(). The output of the cell is a series of log messages and ASCII art. The log messages include: 2022-01-13 11:18:15 WARN Utils:69 - Your hostname, utnoMac-mini.local resolves to 127.0.0.1; did you intend to bind to a network interface?; 2022-01-13 11:18:15 WARN Utils:69 - Set SPARK_LOCAL_IP if you need to bind to a specific interface.; 2022-01-13 11:18:16 WARN NativeCodeLoader:60 - Unable to load native-hadoop library. Using Java API.; Setting default log level to "WARN". To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(); Running on Apache Spark version 3.1.2; SparkUI available at <http://10.18.4.50:4040>; Welcome to; followed by a block of ASCII art resembling a stylized tree or mountain, and the text: version 0.2.81-edeb70bc789c; LOGGING: writing to /Users/ut/hail-20220113-1118-0.2.81-edeb70bc789c.log.



2. Dockerを使ったHailを紹介



その前に、MacでDockerコンテナを利用するには？？

2. Dockerを使ったHailの紹介

標準的には、Docker Desktopが必要です

2021年8月31日にライセンス改訂があり、無償で使用できていたDocker Desktopは
「従業員数250人以上や年間収益1,000万ドル以上の企業」の場合は有料プランへの加入が必要になりました。

その場合はどのプランが必要？ -> 「Docker Pro \$5/month」以上のプランです

- Our [Docker Subscription Service Agreement](#) includes a change to the terms for Docker Desktop
- It **remains free** for small businesses (fewer than 250 employees AND less than \$10 million in annual revenue), personal use, education, and non-commercial open source projects.
 - It requires a paid subscription ([Pro, Team or Business](#)), for as little as \$5 per user per month, for professional use in larger businesses.
 - The effective date of these terms is August 31, 2021. **There is a grace period until January 31, 2022** for those that will require a paid subscription to use Docker Desktop.
 - The Docker Pro, Docker Team, and Docker Business subscriptions **now include** commercial use of Docker Desktop.
 - Check out our [FAQ](#) for more information. Or read our [latest blog](#).

small businessのほかは、
personal use, education,
non-commercial opensource project
がfreeのままであります。

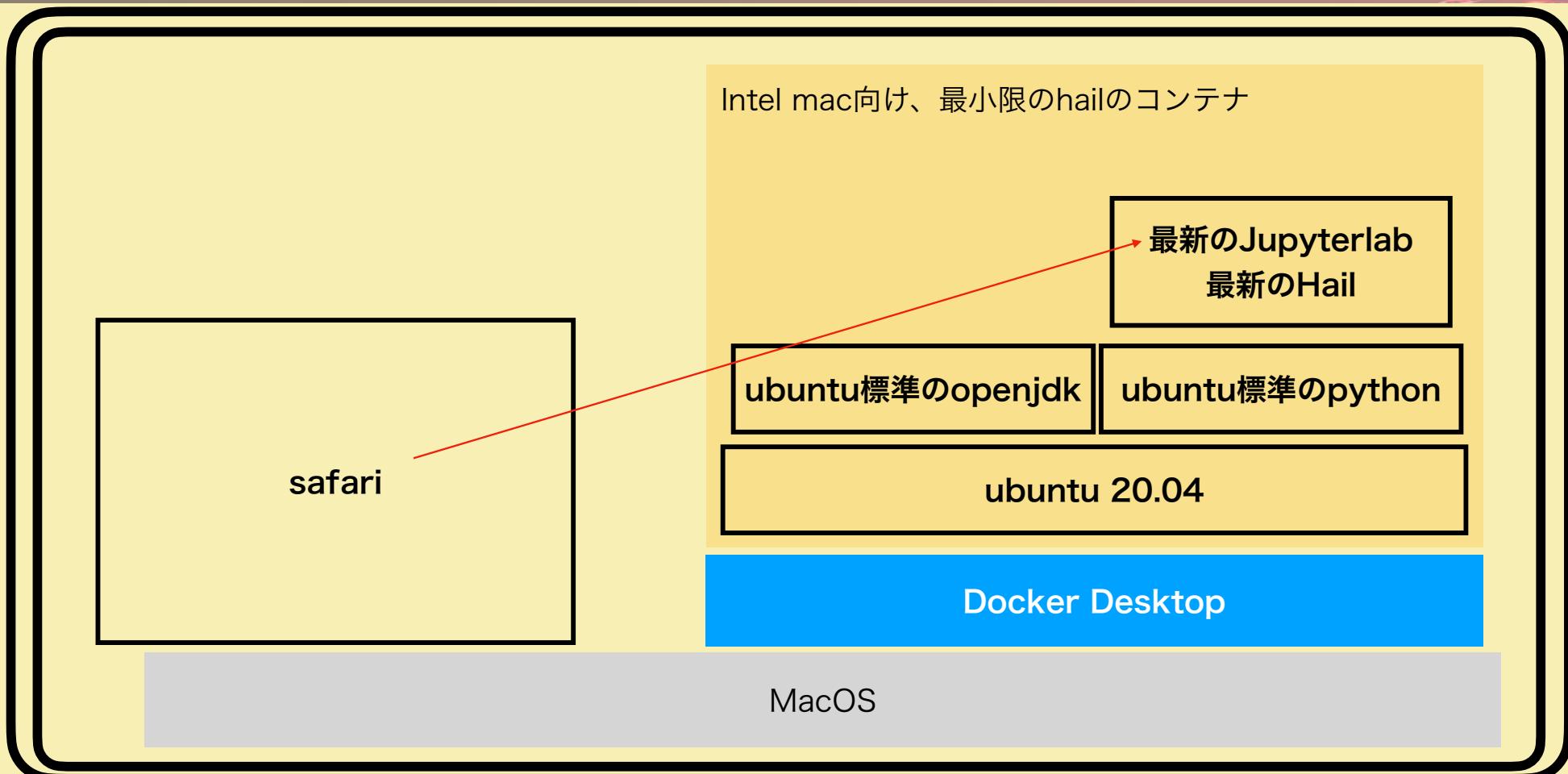
FAQにはより具体的なケースについて解説があります。

この条件は、2022年1月31日までの猶予期間があるので、今日(1/27)は無償で使えます。



Docker Desktopでコンテナを動かし、safariから操作

2. Dockerを使ったHailの紹介

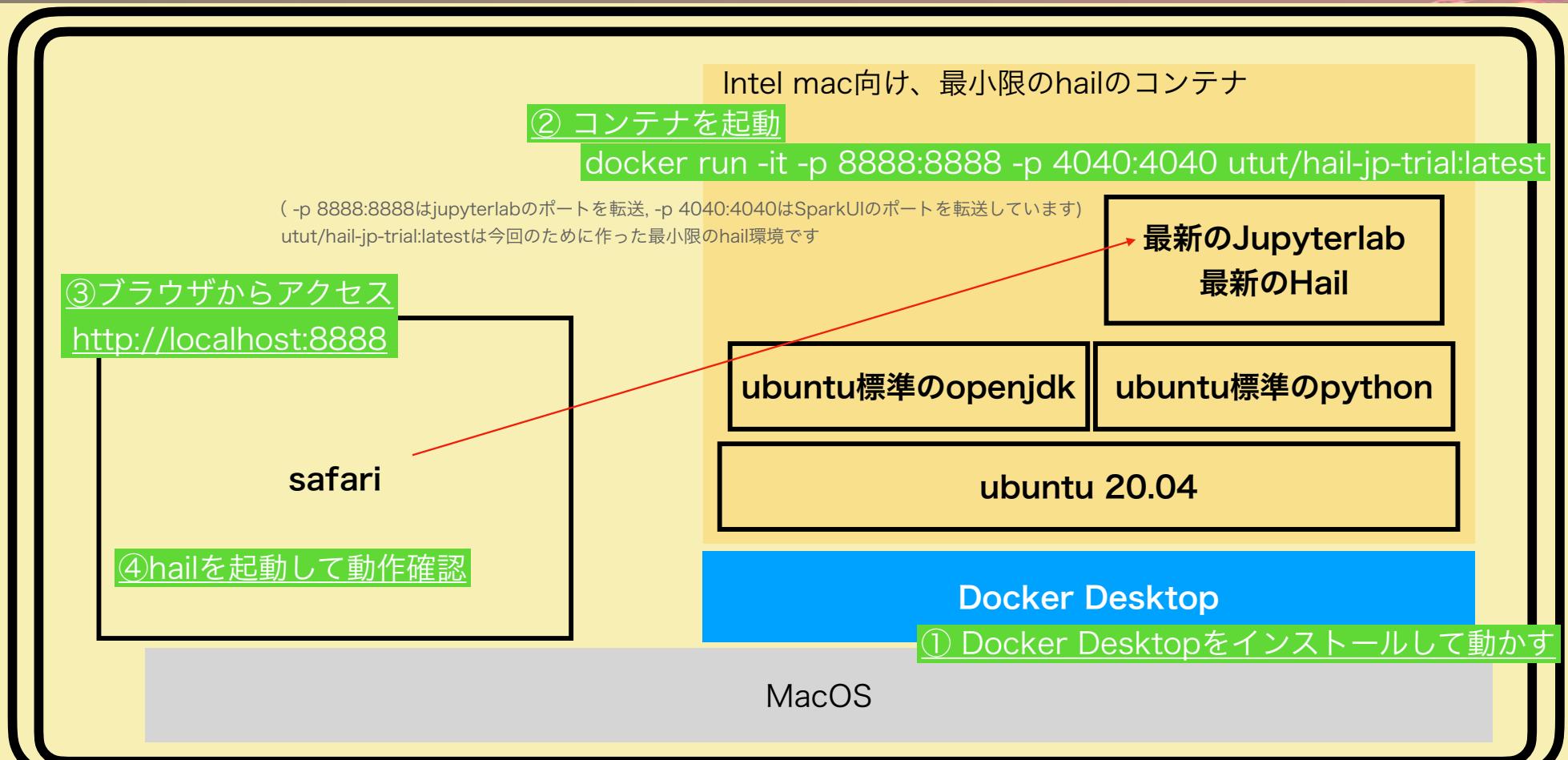


Apple Mac ハードウェア (Intel)

hail.jp

Docker Desktopでコンテナを動かし、safariから操作

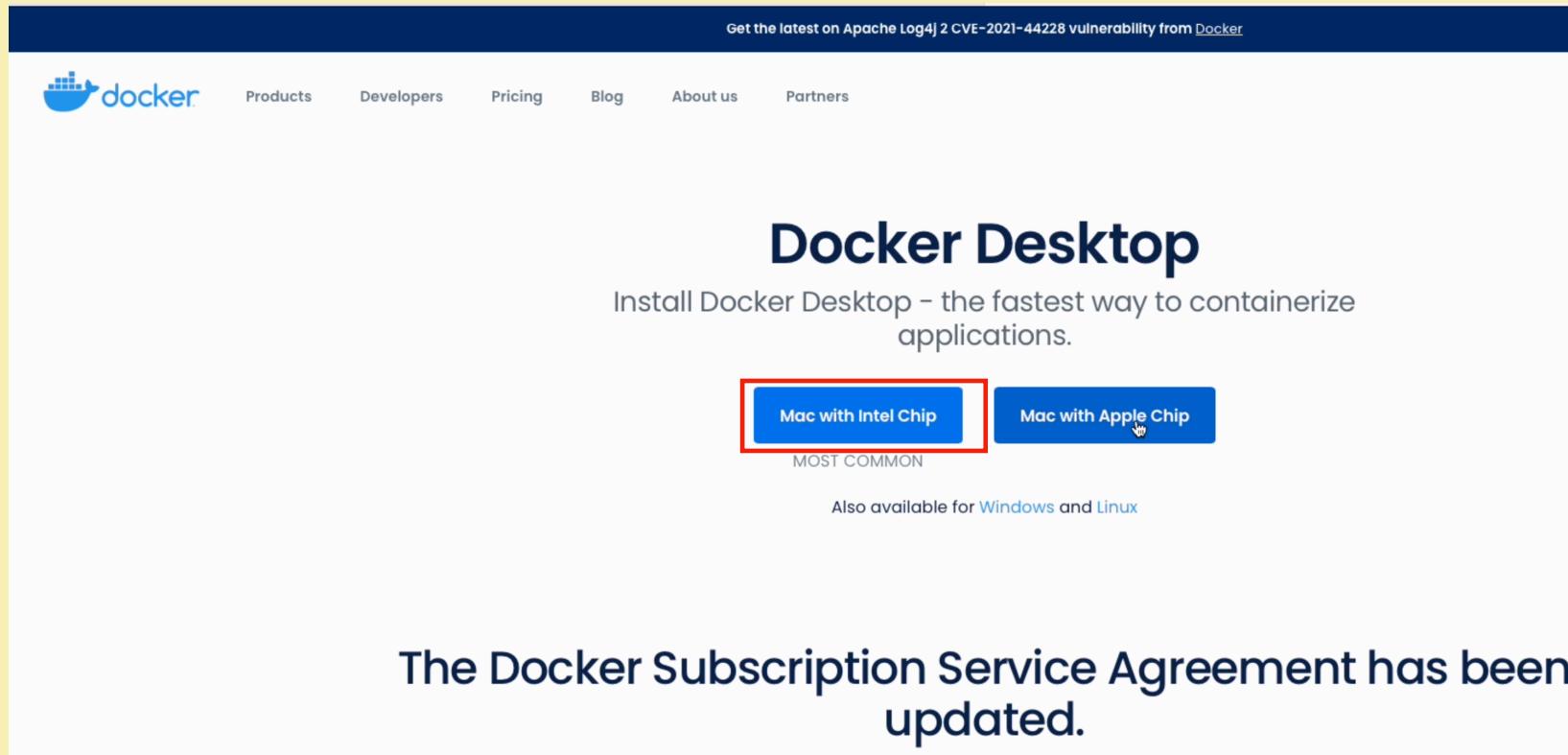
2. Dockerを使ったHailの紹介



Docker DesktopをIntel macへインストール

2. Dockerを使ったHailの紹介

- Docker Desktopのwebページを開き、インストーラーをダウンロード
Intelの場合は「Mac with Intel Chip」をダウンロードします



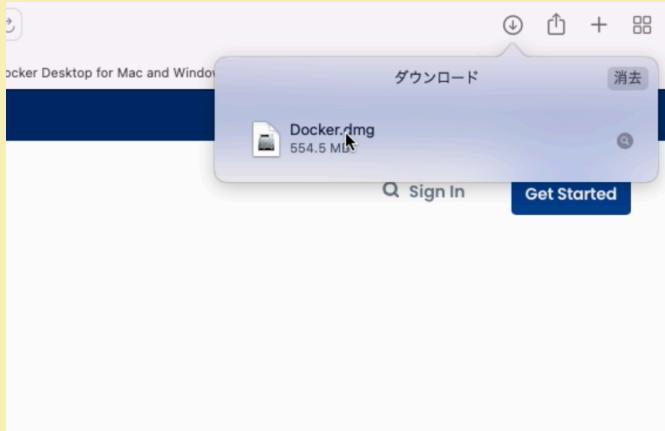
<https://www.docker.com/products/docker-desktop>



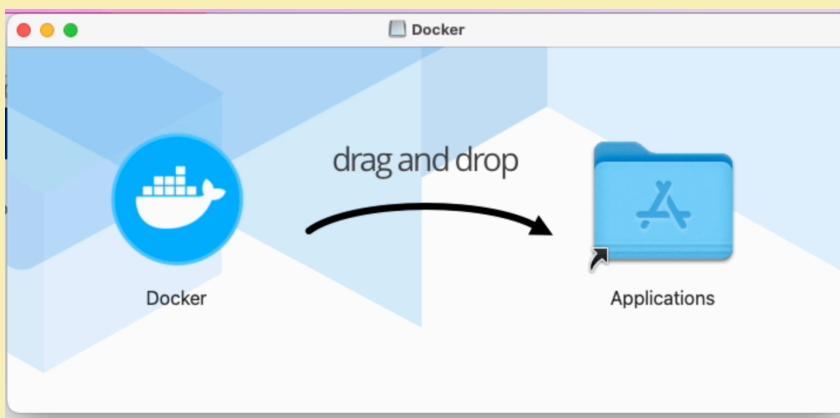
Docker DesktopをIntel macへインストール

2. Dockerを使ったHailの紹介

Downloadしたインストーラーを実行し、



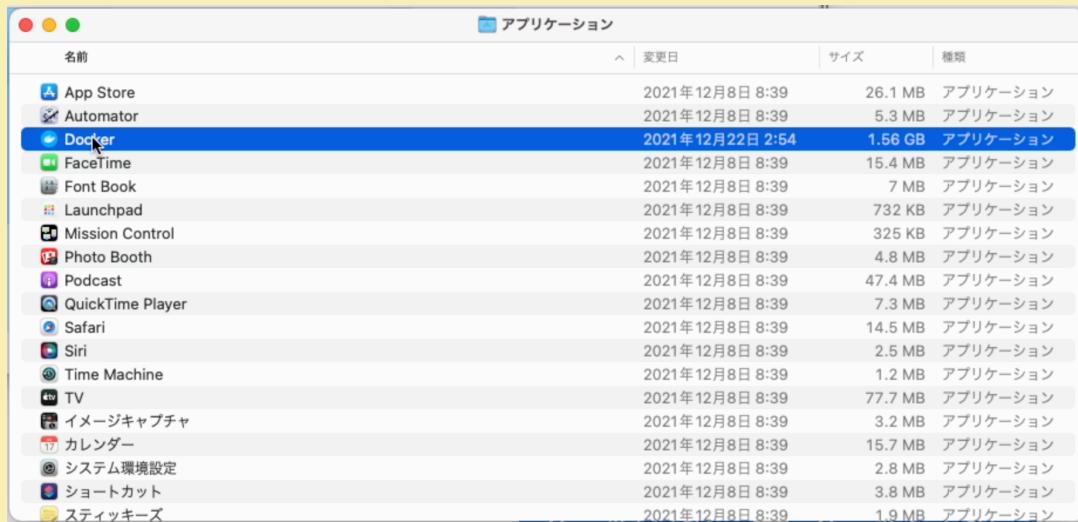
開かれたこの画面でDockerのアイコンをつかんで右のApplicationsに入れます



Docker DesktopをIntel macへインストール

2. Dockerを使ったHailの紹介

アプリケーションにDockerが入るので、ダブルクリックして起動します



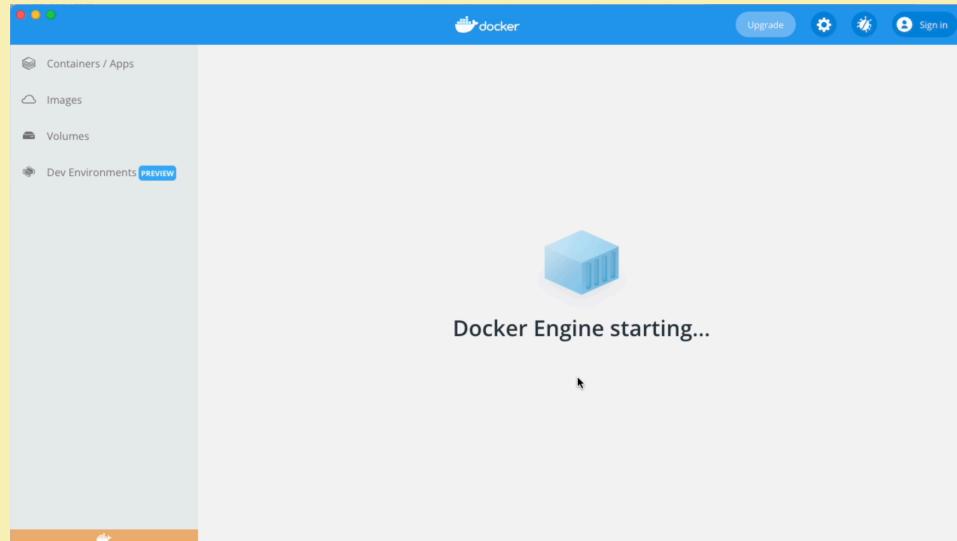
途中、管理者パスワードを入れるなどの作業が少しあります



Docker DesktopをIntel macへインストール

2. Dockerを使ったHailの紹介

このような画面が表示され、Dockerが起動されていきます



Docker Desktopが起動できると、
Terminalでdocker psといったdocker関連コマンドが利用できるようになります

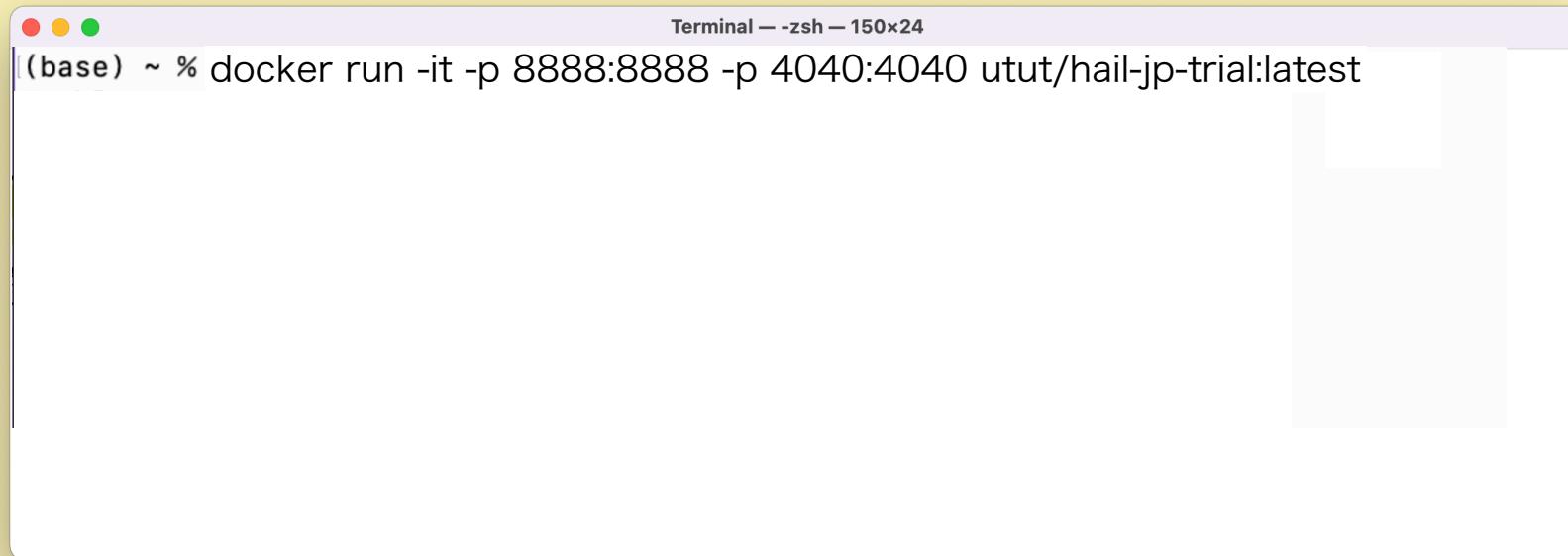
```
(base) ~ % docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
(base) ~ %
```



2. Dockerを使ったHailの紹介

動作確認をしてみましょう

jupyterlabを起動して



A screenshot of a macOS Terminal window titled "Terminal - zsh - 150x24". The window shows the command: "(base) ~ % docker run -it -p 8888:8888 -p 4040:4040 utut/hail-jp-trial:latest". The terminal is currently empty of output.

ブラウザに, `http://localhost:8888` を貼り付けます

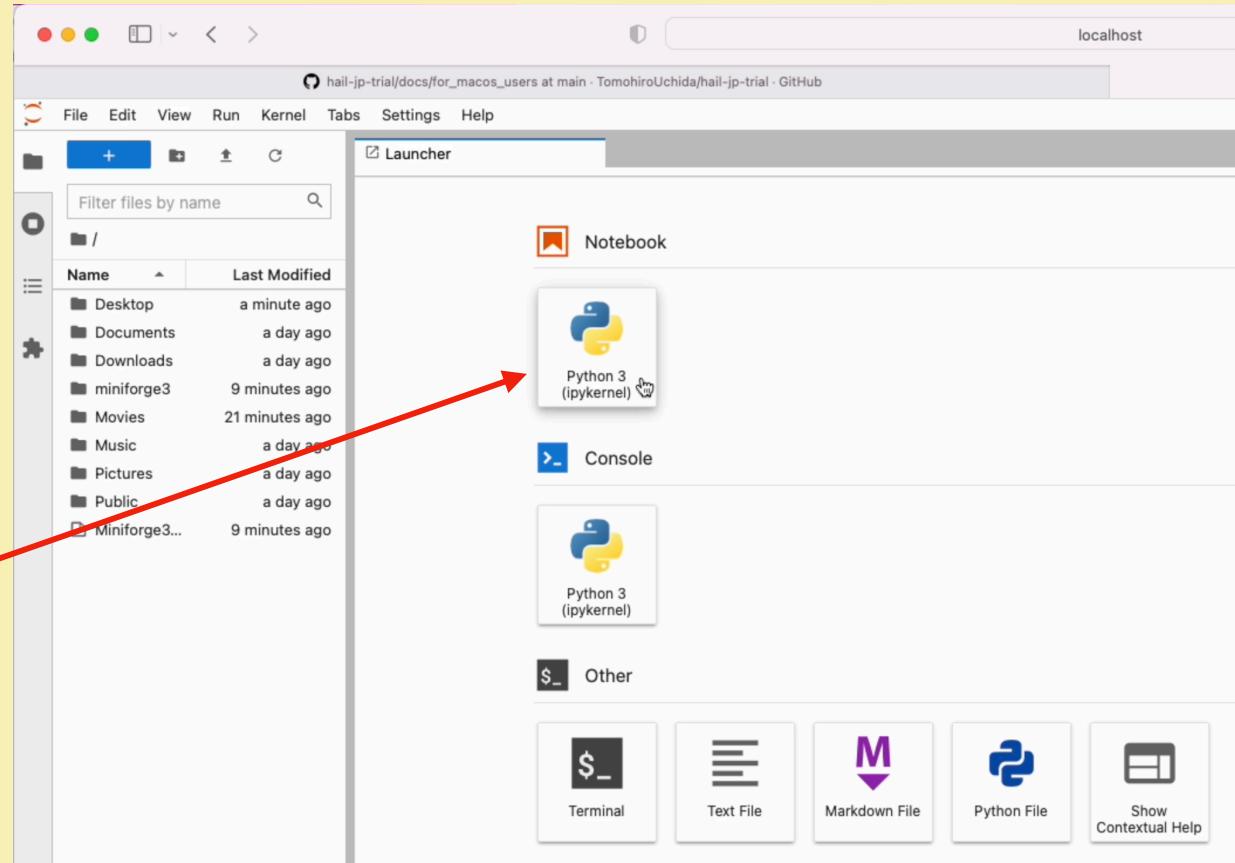


2. Dockerを使ったHailの紹介

動作確認をしてみましょう

このような画面が出てきたらOK

Python3をクリックすると
「Python kernel」が起動します



2. Dockerを使ったHailの紹介

動作確認をしてみましょう

Python kernelが起動したら、

```
import hail as hl
```

を入力して「Shift + Enter」

次に、

hl.init()

を入力して「Shift + Enter」

を押下します。

このように、「Welcome to」や

Hailのステキなアスキーアートが出力されたらOKです！

Untitled.ipynb

[1]: `import hail as hl`

[2]: `hl.init()`

```
2022-01-13 11:18:15 WARN  Utils:69 - Your hostname, utnoMac-mini.local resolves
2022-01-13 11:18:15 WARN  Utils:69 - Set SPARK_LOCAL_IP if you need to bind to .
2022-01-13 11:18:16 WARN  NativeCodeLoader:60 - Unable to load native-hadoop li
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel
Running on Apache Spark version 3.1.2
SparkUI available at http://10.18.4.50:4040
Welcome to
   __  
  / \_ / \_   _/ /  
 / _ \ / -`// /  
 / / / \_,/_//  version 0.2.81-edeb70bc789c
LOGGING: writing to /Users/ut/hail-20220113-1118-0.2.81-edeb70bc789c.log
```



3. 質疑応答

宜しければ、Hail-jp のslackでもご相談ください



3. 質疑応答

宜しければ、Hail-jp のslackでもご相談ください



次回案内

hail ~ jp

