

# コンテナ型システム 導入概説書

V 1.0

## 改訂履歴

Version	日付 (改訂者)	改訂内容	改訂理由	確認者 日付	備考
1.0	2021/10/25 (須田博英)				初版

---

---

目次

1.	コンテナ型システムの概要	1
1.1	機能と特徴	1
1.2	コンテナ型システムの起動・終了方法	2
1.3	コンテナ型システムの操作画面	3
2.	アプリケーション開発環境のコンテナ型システム導入例	4
2.1	バージョン管理システム Git	4
2.2	プロジェクト管理システム Redmine	5
2.3	リモートデスクトップ版 Ubuntu	6
2.4	Visual Studio 互換開発 MonoDevelop	7
2.5	ゲームエンジン Unity	8
3.	教育・研修アプリケーションのコンテナ型システム導入例	9
3.1	学習管理システム Moodle	9
3.2	電子ポートフォリオ管理システム Mahara	10
3.3	全文検索システム Fess	11
4.	機械学習環境のコンテナ型システム導入例	12
4.1	データサイエンス学習の動作環境 Jupyter Notebook (R、Python カーネル)	12
4.2	機械学習の動作環境 Jupyter Notebook (C#カーネル)	13
5.	参考資料	15
5.1	Web サイト	15

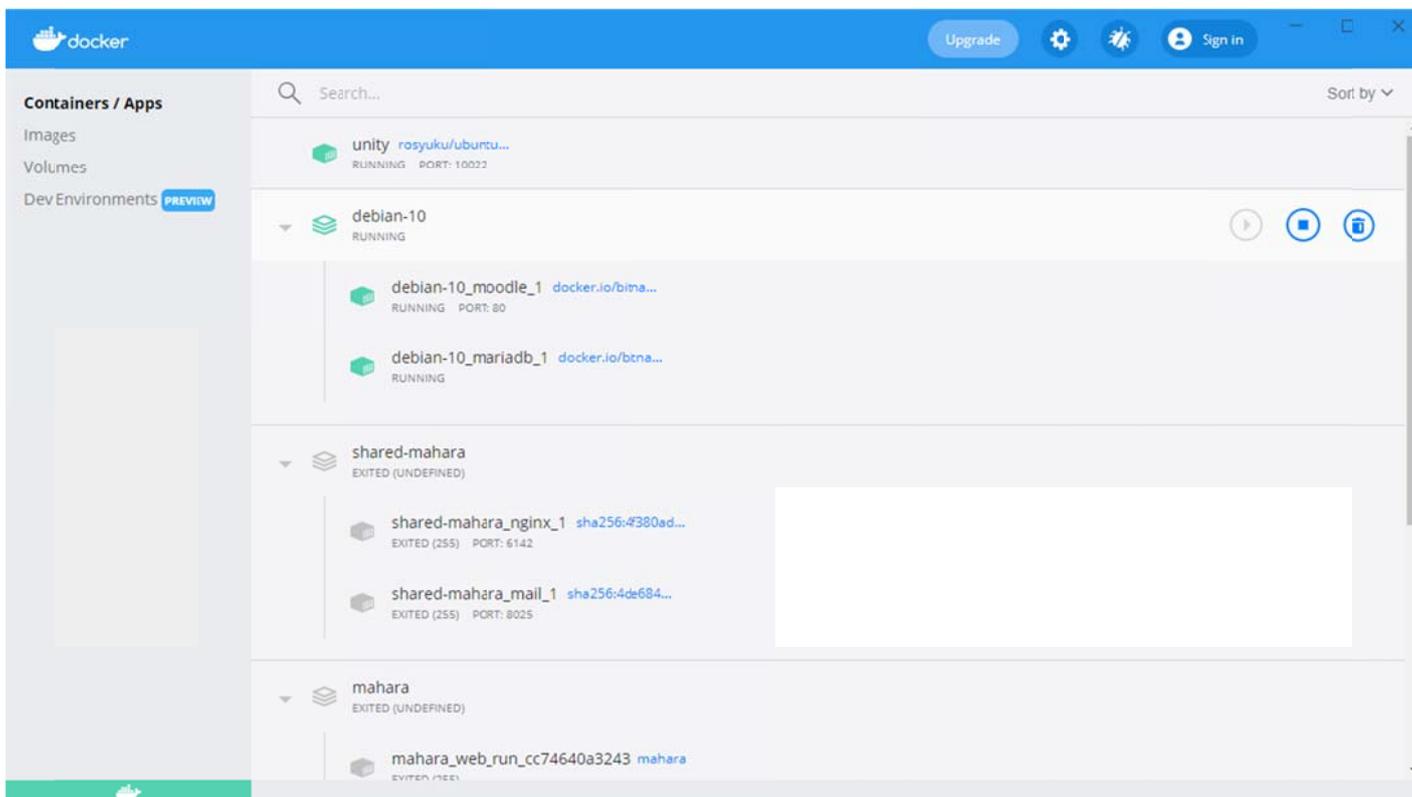
## 1. コンテナ型システムの概要

### 1.1 機能と特徴

コンテナ型システムとは、コンテナ型仮想マシン環境 Docker で動作するコンテナ構成のシステムとします。コンテナ型システムは、起動が高速で、コンピュータ資源の消費量が少なく、動作も軽快です。また、開発から運用まで幅広い領域で、顕著な作業の効率化を行うことができます。そして、次のような機能と特徴があります。

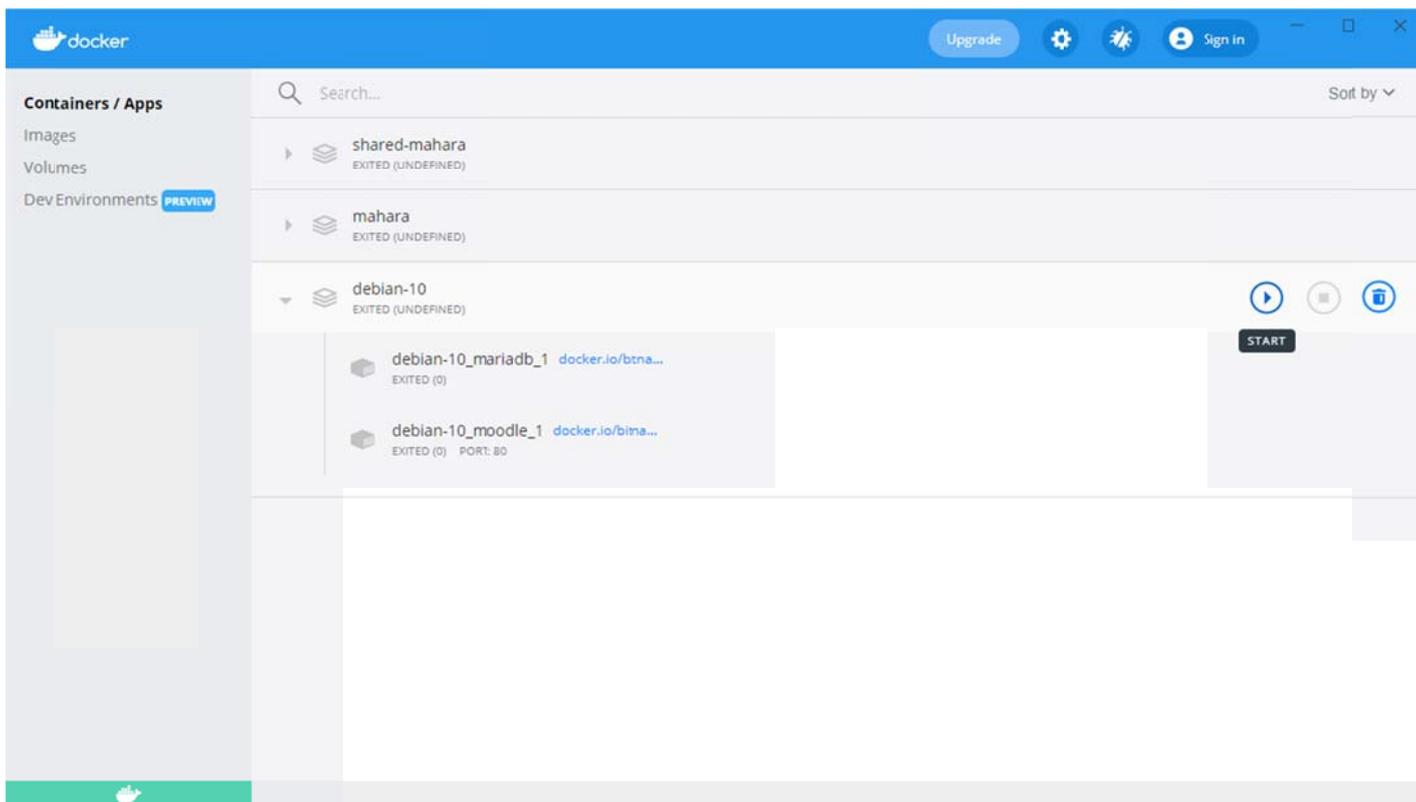
- ・コンテナ型システムは、アプリケーションとその実行環境が一体となっており、開発環境、テスト環境、本番環境などの環境に依存しない頒布が行え、その導入も非常に容易です。
- ・開発したアプリケーションの Docker イメージを用いると、開発時の環境とは別の環境で、設定やデータなどと同じ状態で動作をさせることができます。
- ・コンテナ型の仮想マシンは実マシンと隔離されているため、導入したコンテナが実マシン内を汚すこともなく、完全な削除も容易です。したがって、使い捨て的な利用も可能です。
- ・Docker Hub などの公開レジストリに、基本 OS (Ubuntu、CentOS など)、サーバアプリ (Git、Redmine、Fess、Moodle、Mahara など)、開発環境 (R、Python、C#、Unity など)、機械学習環境 (Jupyter Notebook 用言語カーネルなど)、等多種多様なコンテナが豊富に登録されており、無料で利用できるものも多くあります。
- ・複雑なインストール作業を必要とするサーバ群を導入する場合でも、公開レジストリに Docker コンテナがあれば、短時間で導入し稼働させることができます。
- ・Docker は、Windows、Mac、Linux などで作動します。また、Linux ベースと Windows ベースのコンテナがありますが、現時点では Linux ベースのコンテナが主力です。
- ・Docker は、Apache License 2.0 で提供され、無償で利用することができます。

下図は、Docker for Windows のダッシュボード画面で、実行中のコンテナは緑色となっています。



## 1.2 コンテナ型システムの起動・終了方法

Docker for Windows の場合、起動はダッシュボード画面の [START] ボタンで行い、終了は [STOP] ボタンで行います。また、動作状況もビジュアルに確認することができます。



コマンドラインからの起動の場合、「docker start コンテナ名」で行い、終了は「docker stop コンテナ名」で行います。また、起動しているか終了しているかを「docker ps」で確認することができます。

```

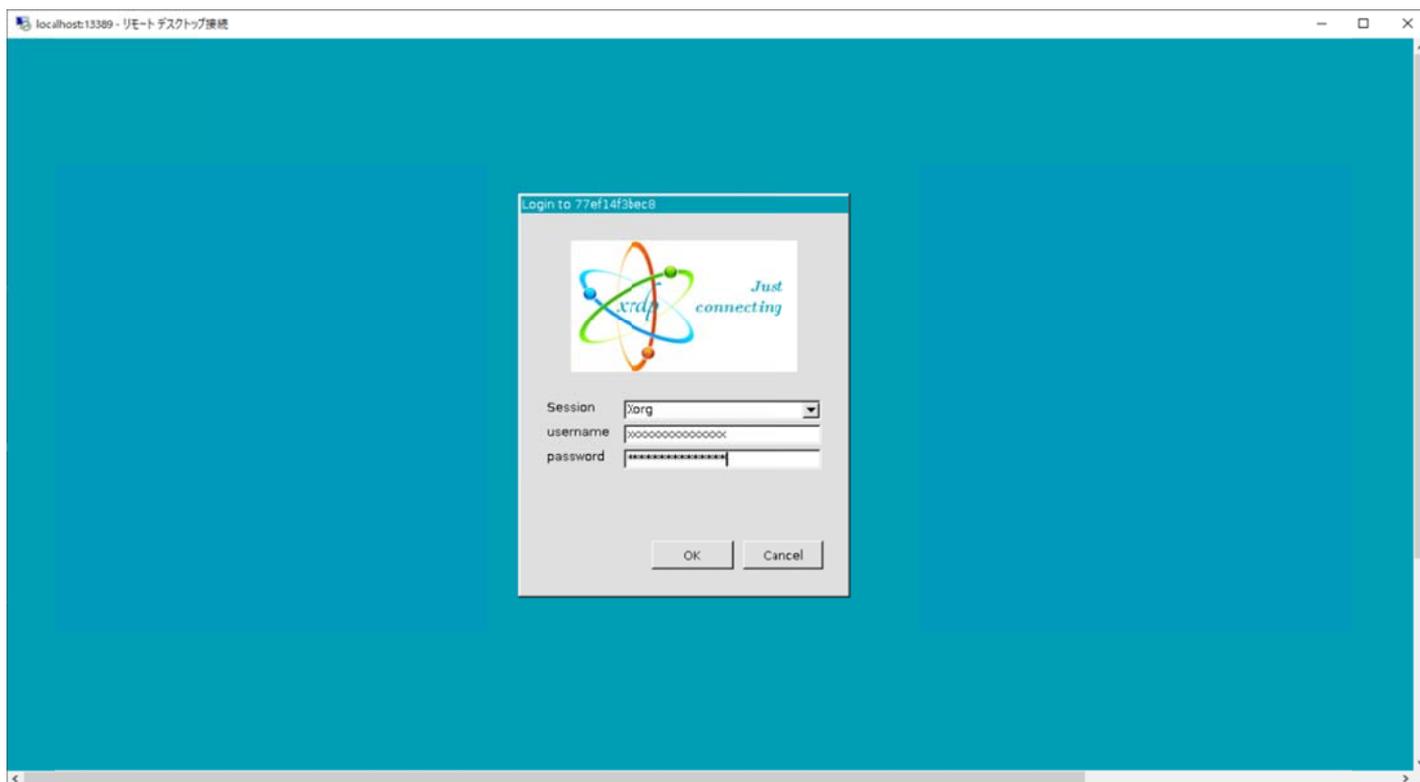
10.2200.1:22 - root@ibc-net8 - VI
ファイル名 編集 設定 コントロール ヘルプ
[root@ibc-net8 ~]#
[root@ibc-net8 ~]# docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
docker.io/jupyter/datascience-notebook  latest             9e84f3a58ed        22 months ago     4.91 GB
docker.io/codellibs/fess                   latest             7463f390faf5       22 months ago     1.77 GB
docker.io/sameersbn/mysal                  latest             c5c4d81209bf       2 years ago        268 MB
docker.io/skerv/gitbucket                  latest             8f619f1435a        5 years ago        520 MB
[root@ibc-net8 ~]#
[root@ibc-net8 ~]#
[root@ibc-net8 ~]#
[root@ibc-net8 ~]# docker ps -a
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS
ba8621bd0f19       7463f390faf5       "/bin/sh -c /usr/s..." 15 months ago      Exited (1) 4 months ago
e89cc5f0c046       skerv/gitbucket    "java -jar /opt/g..." 21 months ago      Exited (255) 17 months ago    0.0.0.0:20080->8080/tcp
2b2f851e1733       jupyter/datascience-notebook "jupyter --start..." 21 months ago      Exited (255) 17 months ago    0.0.0.0:8888->8888/tcp
cad3af637ab1       codellibs/fess:latest "/bin/sh -c /usr/s..." 21 months ago      Exited (255) 17 months ago    9200/tcp, 9300/tcp, 0.0.0.0:10084->8080/tcp
[root@ibc-net8 ~]#
[root@ibc-net8 ~]#
[root@ibc-net8 ~]#
[root@ibc-net8 ~]# docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS              NAMES
[root@ibc-net8 ~]#
[root@ibc-net8 ~]#
[root@ibc-net8 ~]# docker start fess
fess
[root@ibc-net8 ~]#
[root@ibc-net8 ~]#
[root@ibc-net8 ~]#
[root@ibc-net8 ~]# docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS              NAMES
cad3af637ab1       codellibs/fess:latest "/bin/sh -c /usr/s..." 21 months ago      Up 29 seconds      9200/tcp, 9300/tcp, 0.0.0.0:10084->8080/tcp    fess
[root@ibc-net8 ~]#
[root@ibc-net8 ~]#
[root@ibc-net8 ~]#
[root@ibc-net8 ~]# docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS              NAMES
cad3af637ab1       codellibs/fess:latest "/bin/sh -c /usr/s..." 21 months ago      Up 9 hours         9200/tcp, 9300/tcp, 0.0.0.0:10084->8080/tcp    fess
[root@ibc-net8 ~]#
[root@ibc-net8 ~]#
[root@ibc-net8 ~]# docker stop fess
fess
[root@ibc-net8 ~]#
[root@ibc-net8 ~]#
[root@ibc-net8 ~]#
[root@ibc-net8 ~]# docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS              NAMES
[root@ibc-net8 ~]#
[root@ibc-net8 ~]#

```

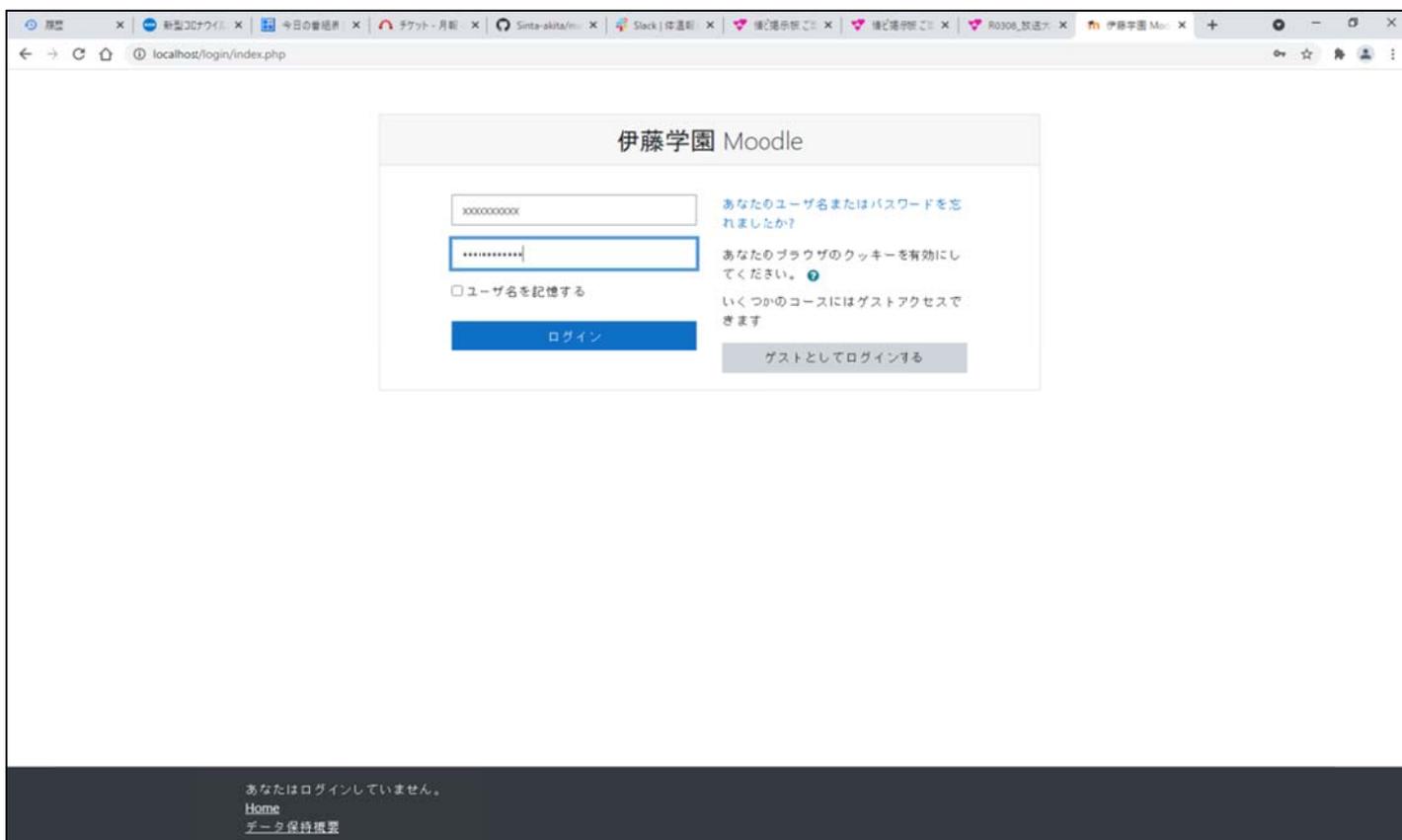
### 1.3 コンテナ型システムの操作画面

操作画面は、利用する Docker コンテナのアプリケーションによって、コマンドライン、リモートデスクトップ、Web ブラウザなど異なります。

下図では、Docker for Windows 上のリモートデスクトップ版 Ubuntu へリモートデスクトップ接続しています。



下図では、Docker for Windows 上の学習管理システム Moodle へ Web ブラウザからログイン接続しています。



## 2. アプリケーション開発環境のコンテナ型システム導入例

プライベートな Git サーバ、Redmine サーバ、Ubuntu のリモートデスクトップ環境、Unity 開発環境の導入例を説明します。

### 2.1 バージョン管理システム Git

Git(ギット)は、ソースコードの管理を行う分散型バージョン管理システムで、変更履歴を記録・追跡することができます。導入は公開レジストリから行えます。

The image shows two screenshots related to Git. The top screenshot is a browser view of the GitBucket 3.9 interface for a repository named 'root / sudaxp'. It displays 4 commits, 1 branch (master), and 0 releases. A table lists recent commits:

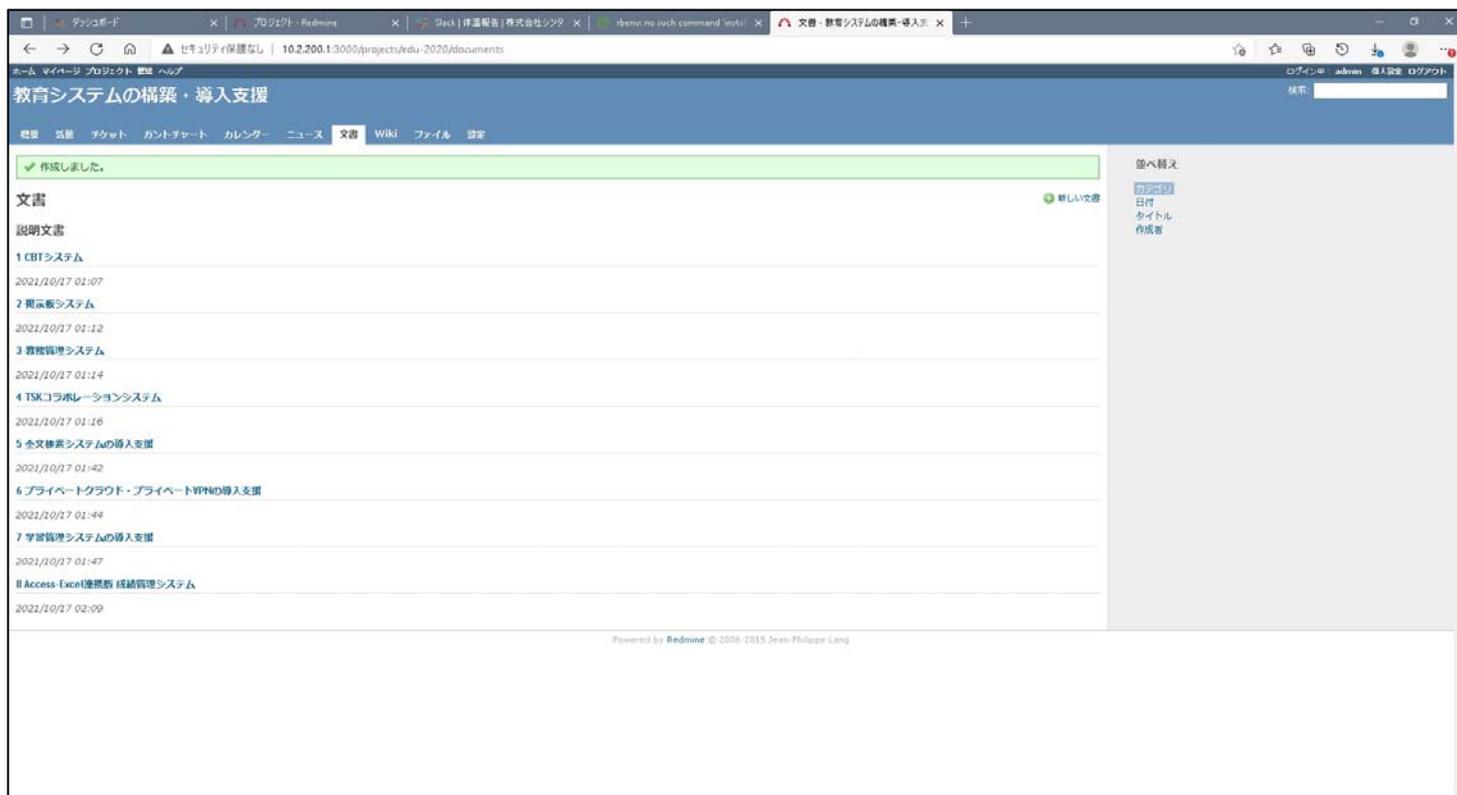
Commit Hash	Author	Message	Time
13ccd13a8a	須田 博英	2019年度対応	2 days ago
7d61af1	須田 博英	Windows Server 2012 R2 対応	2 days ago
4105c7f	須田 博英	20190817までの参考サイト追加	5 days ago
c5631ed	須田 博英	Initial Commit	5 days ago

The bottom screenshot shows the Git GUI application. The commit history is displayed in a table:

Commit Hash	Message	Time	Author	Commit
13ccd13	2019年度対応	2019/8/18 12:44	須田 博英 <suda@>	13ccd13
7d61af1	Windows Server 2012 R2 対応	2019/8/18 12:40	須田 博英 <suda@>	7d61af1
4105c7f	20190817までの参考サイト追加	2019/8/18 11:36	須田 博英 <suda@>	4105c7f
c5631ed	Initial Commit	2019/8/15 19:33	須田 博英 <suda@>	c5631ed

## 2.2 プロジェクト管理システム Redmine

Redmine(レッドマイン)は、プロジェクトの運営を支援するためのさまざまな機能を備えている Web ベースのプロジェクト管理システムです。導入は公開レジストリから行えます。



下図の Redmine のシステムは、Redmine 本体と Postgres データベースのコンテナで構成されています。

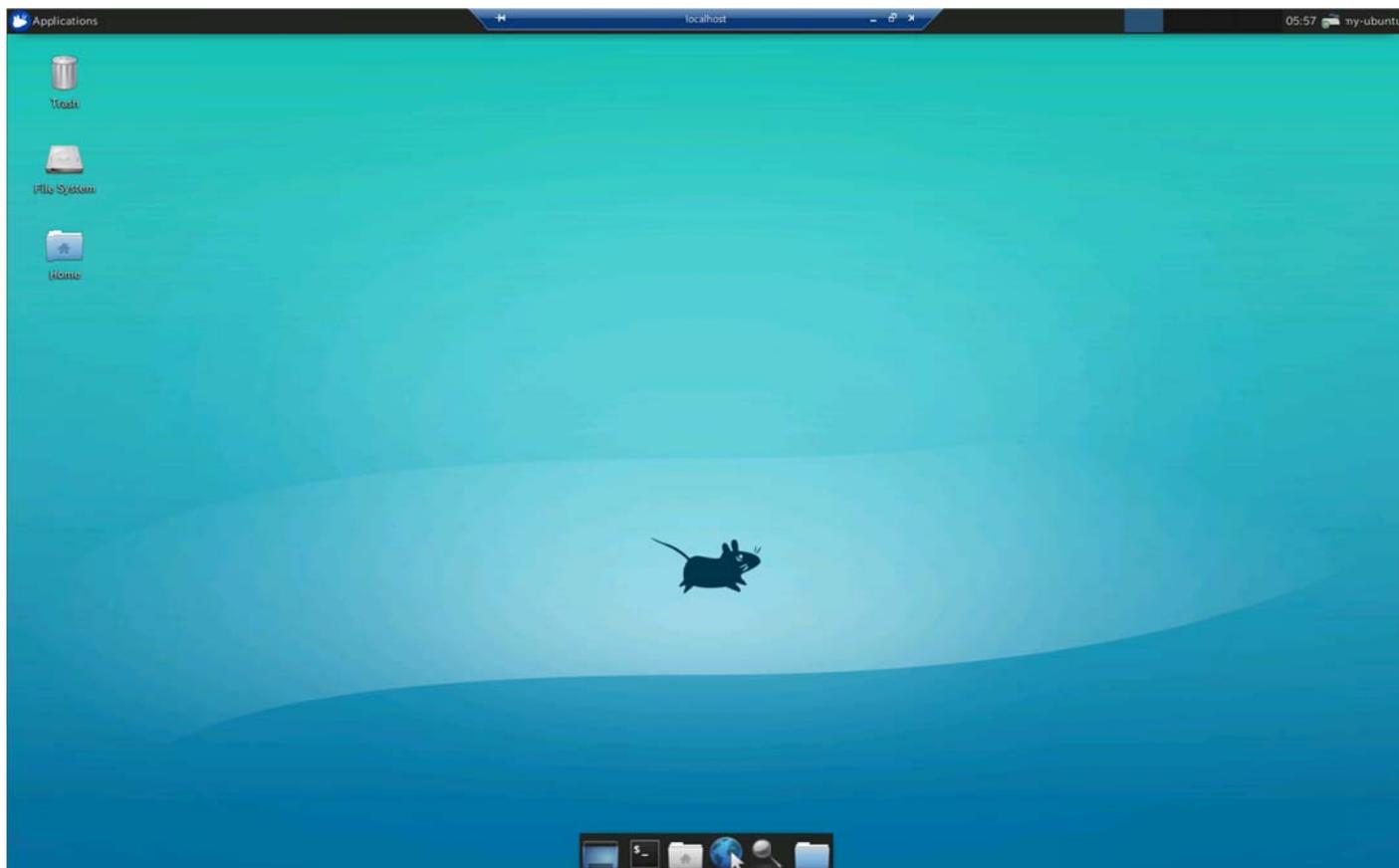
```

102.200.1.22 - root@aic-net8 ~ VT
ファイル 編集 設定 実行 コマンド 実行ログ ヘルプ
a49b446cc15b: Pull complete
8d27130dcbaf: Pull complete
902182b2d33e: Pull complete
80082c2acd82: Pull complete
fc954d89ac0e: Pull complete
0117929a0153: Pull complete
cae65198443f: Pull complete
faba7721aa88: Pull complete
c54110c85e5e: Pull complete
77af9ec5ac8a: Pull complete
59429277fad3: Pull complete
ee2e46b2b88a: Pull complete
e982c7424a1: Pull complete
Digest: sha256:0d53dd852689a00b3d4a08afb1e8336a478275dcbecf151f11f13d54b88ae
Status: Downloaded newer image for docker.io/redmine:3.0.7
3565802f3ea6dd7976f1dc8150fcdc432b56eb6e6786acd04192faa98517374d
[root@aic-net8 ~]#
[root@aic-net8 ~]# docker ps -a
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS              NAMES
3565802f3ea6      redmine:3.0.7      /docker-entrypoin... 53 seconds ago     Up 26 seconds      0.0.0.0:3000->3000/tcp, 5432/tcp              redmine
ce11e484b4a4      postgres:9.5.0    /docker-entrypoin... 4 minutes ago     Up 4 minutes       5432/tcp              postgres
ba86218bd0f9      7463f390faf5      /bin/sh -c /usr/s... 15 months ago     Exited (1) 5 months ago
e89c2a0cd048      sken/siftbucket   java -jar /opt/si... 21 months ago     Exited (255) 18 months ago    0.0.0.0:20080->8080/tcp    naughty_goldwasser
2b2f65f1733      jupyter/datascience-notebook /bin/sh -c start... 21 months ago     Exited (255) 18 months ago    0.0.0.0:8888->8888/tcp    ds20190708
cad3af637ab1      code4libs/fess:latest /bin/sh -c /usr/s... 21 months ago     Up 2 weeks          9200/tcp, 9300/tcp, 0.0.0.0:10084->8080/tcp    fess
[root@aic-net8 ~]#
[root@aic-net8 ~]#
[root@aic-net8 ~]#
[root@aic-net8 ~]# docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
docker.io/jupyter/datascience-notebook    latest             9e64f3a158ed       22 months ago     4.31 GB
docker.io/code4libs/fess                    latest             f40838eb07ab       23 months ago     389 MB
docker.io/redmine                            3.0.7             7463f390faf5       5 years ago       264 MB
docker.io/postgres                          9.5.0             b2b0df88a221       5 years ago       520 MB
docker.io/sken/siftbucket                   latest            8f6b19fb435a       5 years ago
You have new mail in /var/spool/mail/root
[root@aic-net8 ~]#
[root@aic-net8 ~]#
[root@aic-net8 ~]# docker ps -a
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS              NAMES
3565802f3ea6      redmine:3.0.7      /docker-entrypoin... 13 hours ago     Up 13 hours       0.0.0.0:3000->3000/tcp, 5432/tcp              redmine
ce11e484b4a4      postgres:9.5.0    /docker-entrypoin... 13 hours ago     Up 13 hours       5432/tcp              postgres
ba86218bd0f9      7463f390faf5      /bin/sh -c /usr/s... 15 months ago     Exited (1) 5 months ago
e89c2a0cd048      sken/siftbucket   java -jar /opt/si... 21 months ago     Exited (255) 18 months ago    0.0.0.0:20080->8080/tcp    naughty_goldwasser
2b2f65f1733      jupyter/datascience-notebook /bin/sh -c start... 21 months ago     Exited (255) 18 months ago    0.0.0.0:8888->8888/tcp    ds20190708
cad3af637ab1      code4libs/fess:latest /bin/sh -c /usr/s... 21 months ago     Up 2 weeks          9200/tcp, 9300/tcp, 0.0.0.0:10084->8080/tcp    fess
[root@aic-net8 ~]#
[root@aic-net8 ~]#
[root@aic-net8 ~]#

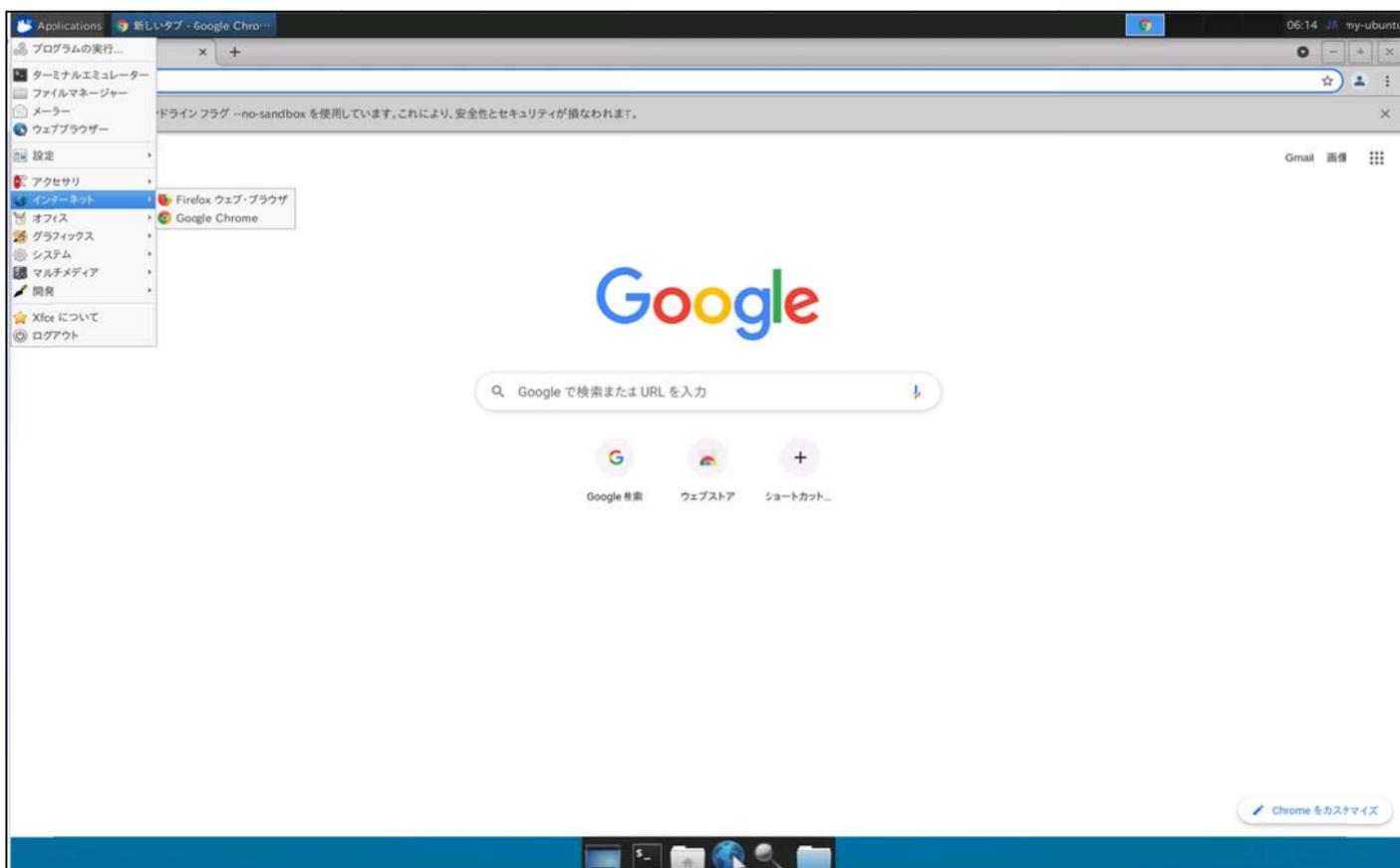
```

## 2.3 リモートデスクトップ版 Ubuntu

下図では、リモートデスクトップ版 Ubuntu へリモートデスクトップ接続しています。デスクトップ環境は、シンプルな見た目、簡単な使用感、軽量・高速な動作などの特長をもつ Xfce となっています。導入は公開レジストリから行えます。



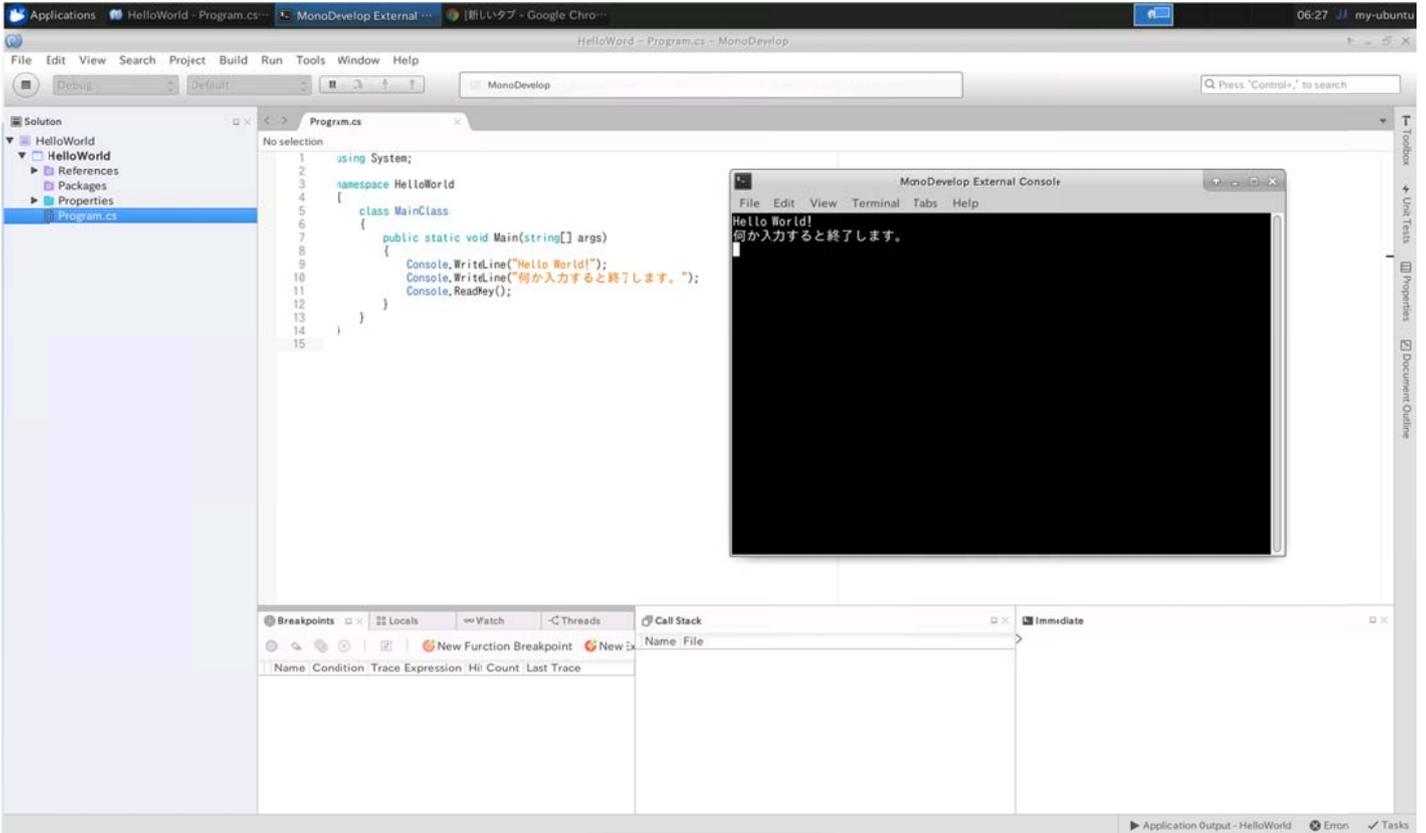
下図では、Web ブラウザ Chrome を利用しています。Web ブラウザは Firefox と Chrome が利用できます。



## 2.4 Visual Studio 互換開発 MonoDevelop

下図は、Visual Studio 互換開発環境 MonoDevelop で C#を使った「Hello, World!」のテスト画面です。

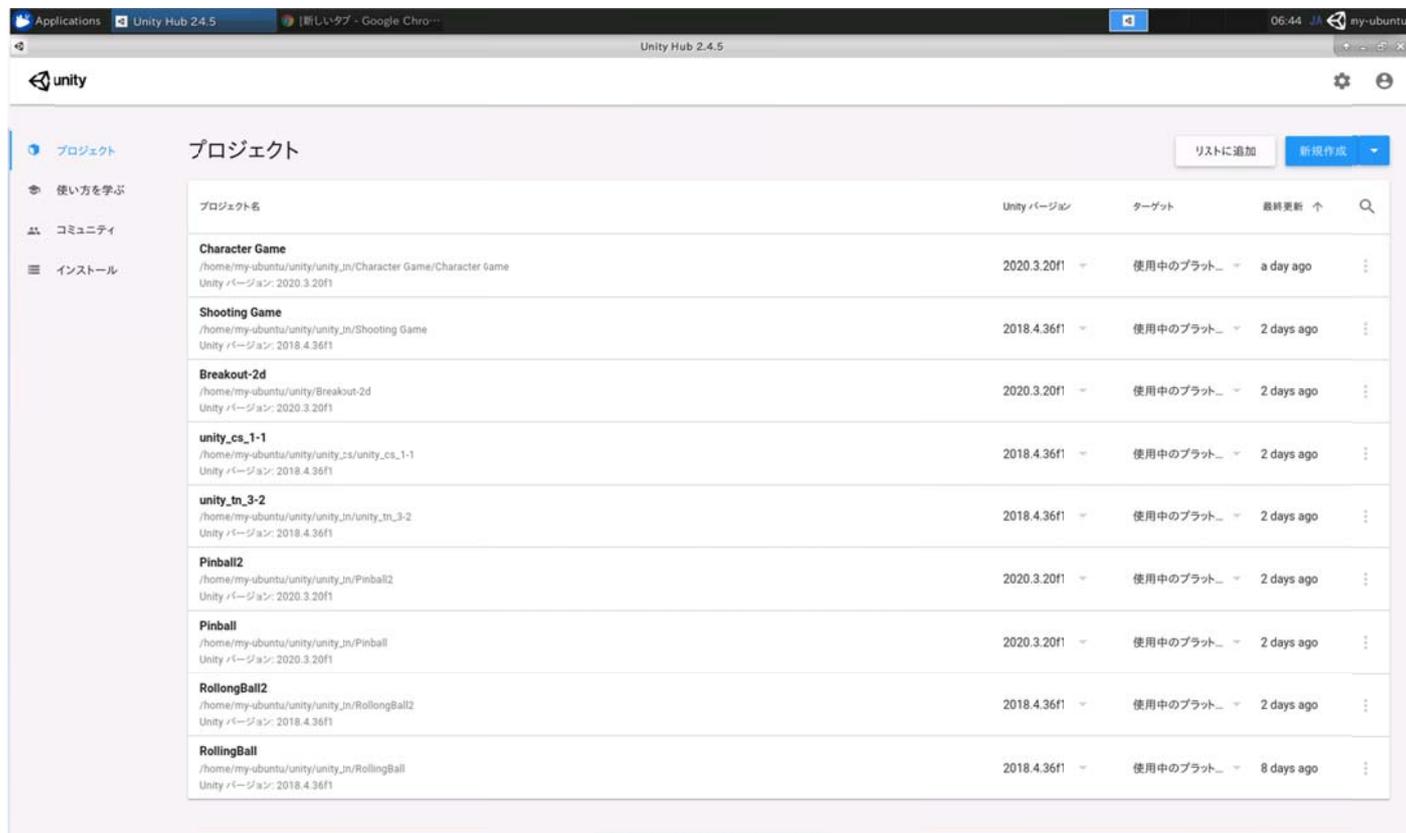
また、Unity 開発環境でのプログラミング言語は C#のため、Visual Studio Tools for Unity と同様に Unity と連携した開発にも利用されます。導入は公開レジストリから行えます。



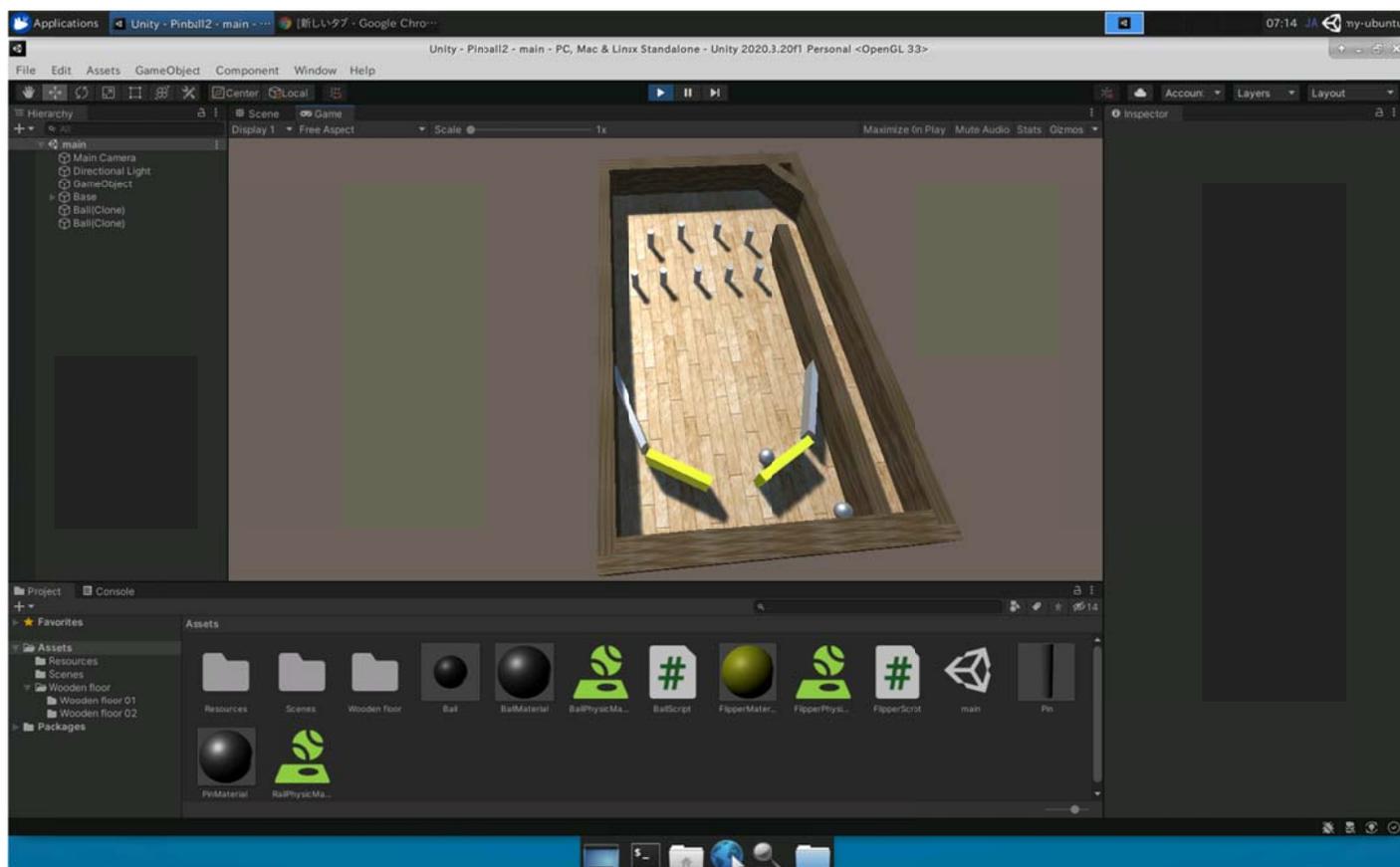
## 2.5 ゲームエンジン Unity

Unity は、ゲーム以外にもバーチャル技術に応用され、医療、セキュリティ、建築設計など様々な分野のインタラクティブ体験に利用されています。

下図は、Unity を利用するための推奨環境である Unity Hub の画面です。Unity Hub は、プロジェクトと Unity バージョンの設定、ダウンロード、管理などを行うアプリケーションです。



下図は、Unity エディタで開発したゲームのテスト画面です。Unity は、WEB ブラウザ、Windows、Mac、Linux、iOS、Android、PlayStation、Xbox、Wii などに対応した開発ができます。



### 3. 教育・研修アプリケーションのコンテナ型システム導入例

#### 3.1 学習管理システム Moodle

Moodle(ムードル、Modular Object-Oriented Dynamic Learning Environment)は、学習管理システム(LMS、Learning Management System)と呼ばれるシステムで、資料の掲示、課題の提示と回収、問題の提示と評価、アンケート、掲示板、共同作業、連絡など多くの機能があります。導入は公開レジストリから行えます。

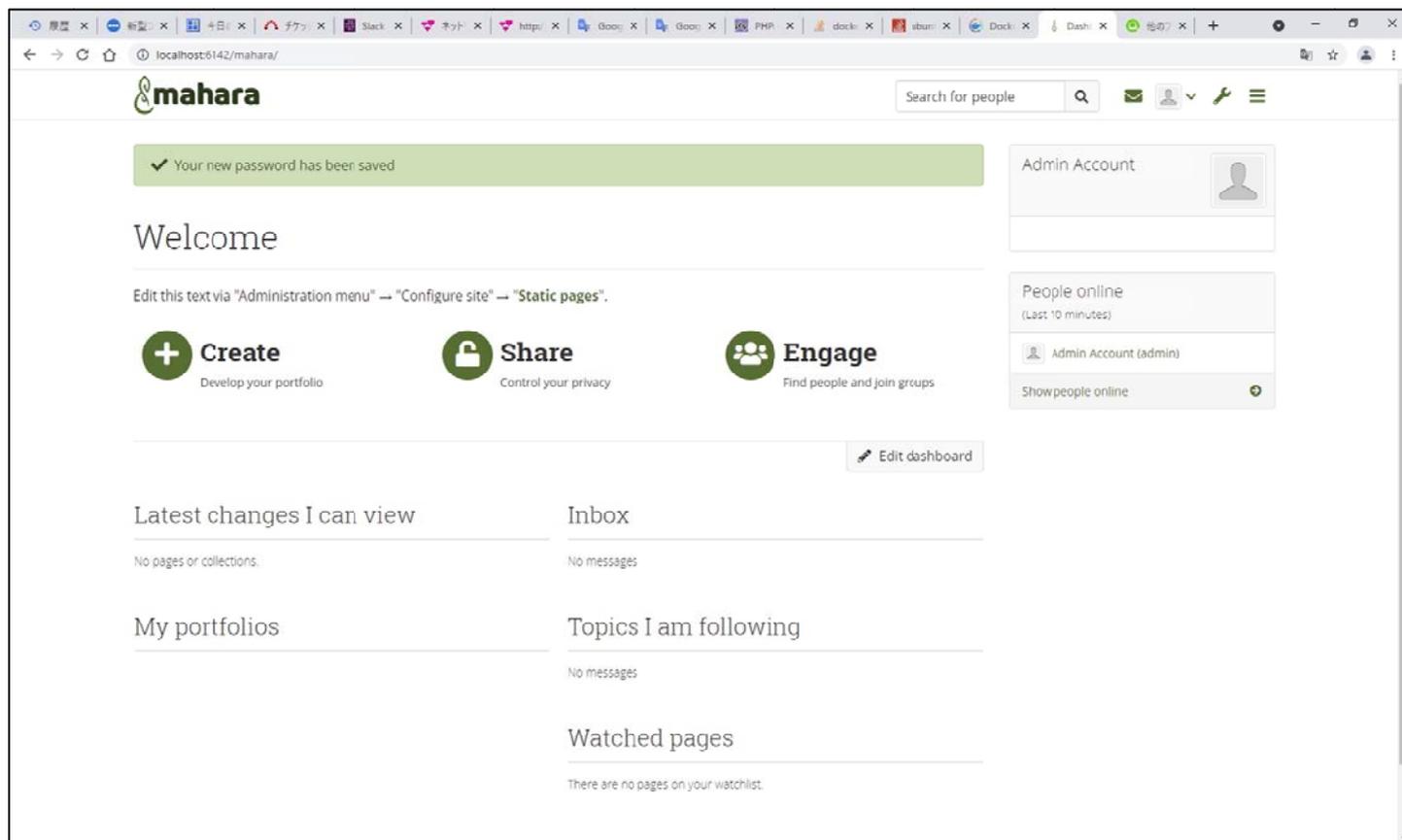
The screenshot shows a Moodle course dashboard for 'XXXXXX専門学校'. The main area displays a course overview with progress indicators for several subjects: '(21-1-SE)ネットワーク基礎' (0%), '(21-1-SE)セキュリティ基礎' (0%), '(21-1-GP)LHR I' (0%), and '(21-2-SE)JavaScript' (30%). The JavaScript section includes a brief description: 'Webサイトのホームページは、HTMLと呼ばれる言語によってその内容は記述されています。Webブラウザでホームページのソースコードを閲覧すると、HTMLの他に見栄えを整えるcss、動作を制御するJavaScript、数式を記述するMathML、画像を記述するSVGなど。' The right sidebar contains sections for 'プライベートファイル', 'オンラインユーザ' (1 user online), '最新バッジ', 'カレンダー' (October 2021), and '直近イベント'.

下図は、物理学の問題の出題例です。入力した数式の解答は自動採点されます。

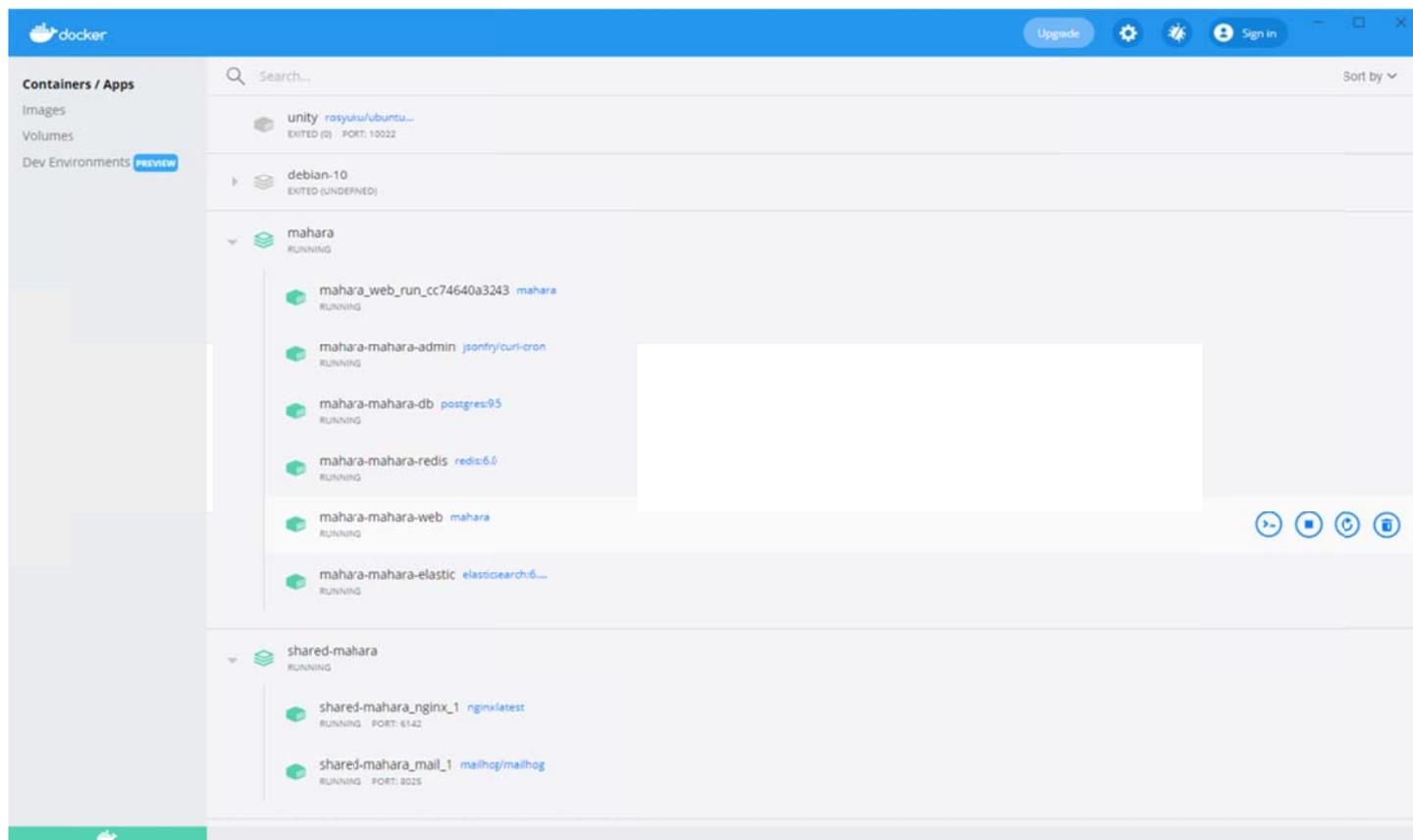
The screenshot shows a Moodle quiz question page titled '2021 Stack サンプル'. The question is about a mass-spring system. The text reads: '図の状況で、右の物体の位置 $x_2$ の満たす運動方程式は、' followed by a text input field containing the equation  $m_2 \frac{d^2 x_2}{dt^2} = k_2^* (-x_2 - l_2 + L) - k_1^* (x_2 - x_1 - l_1)$ . Below the equation, it says: 'ただし、STACK/MAXIMAの認識する書式の数式で答えよう。位置は $x_1(t)$ 、 $x_2(t)$ でなく単に $x_1, x_2$ と書いてね。STACK問題での数式による答え方参照。' A diagram shows two masses  $m_1$  and  $m_2$  on a horizontal surface with springs  $k_1$  and  $k_2$  and lengths  $l_1$  and  $l_2$ . The positions are labeled  $x_1(t)$  and  $x_2(t)$ . The input field shows the user's answer:  $k_2 \cdot (-x_2 - l_2 + L) - k_1 \cdot (x_2 - x_1 - l_1)$ . The system also displays the variables used in the answer:  $[L, k_1, k_2, l_1, l_2, x_1, x_2]$ . The page includes navigation buttons for '前のページ', '次のページ', and 'ジャンプ'.

### 3.2 電子ポートフォリオ管理システム Mahara

Mahara は、学習成果、キャリアなどさまざまな記録を蓄積・一元管理し、グループウェアや SNS の機能により、蓄積した記録の活用を支援する電子ポートフォリオシステムです。導入は公開レジストリから行えます。



下図では、コマンドラインから Mahara のシステムを起動後に、緑色 8 個のコンテナが実行中となっています。



### 3.3 全文検索システム Fess

FESS (Fess Site Search) は、複数サーバに分散した文字ファイル、オフィスファイル、PDF などの索引作成、Web ブラウザからの全文検索に利用されます。導入は公開レジストリから行えます。

The screenshot shows the FESS search interface with the search term '相転移'. The results list includes several PDF documents related to phase transitions in physics and materials science. On the right side, there are filter panels for '期間' (Period), 'サイズ' (Size), and 'ファイルの種類' (File Type).

期間	件数
1ヶ月以内	6
1年以内	9

サイズ	件数
- 10KB	6
10KB - 100KB	405
100KB - 500KB	30
500KB - 1MB	9
1MB -	19

ファイルの種類	件数
HTML	420
MS PowerPoint	2
PDF	36
その他	11

下図では、全格納ドキュメント数 4,234,369/41.44GB、検索対象ドキュメント数 860,151/39.75GB です。

The screenshot shows the FESS administration dashboard. At the top, it displays system statistics: 1 nodes, 41 indices, 88 shards, 4,284,369 docs, and 41.44GB of storage. Below this, there are several log files listed with their respective sizes and shard counts. The dashboard also includes a sidebar with navigation options like 'ダッシュボード', 'システム', 'クローラ', 'ユーザー', 'サジェスト', and 'システム情報'.

## 4. 機械学習環境のコンテナ型システム導入例

### 4.1 データサイエンス学習の動作環境 Jupyter Notebook (R、Python カーネル)

下図は、R 言語で Caret パッケージを使った機械学習のテスト画面です。計算量が膨大になるため、計算を CPU の 16 コアに並列化して計算速度を上げています。なお、Jupyter Notebook は対話形式の開発環境です。

テスト環境の Jupyter Notebook 用 R、Python カーネルの導入は公開レジストリから行えます。

```

In [9]: # -----
# caretパッケージによる予測
# -----

# 必要であれば、caretパッケージをインストール
install.packages("caret")
library(caret)

# 並列化演算を行う
install.packages("doParallel")
library(doParallel)

# コア数を検出
detectCores()

Updating HTML index of packages in '.Library'
Making 'packages.html' ... done
Loading required package: lattice
Loading required package: gsploit2
Updating HTML index of packages in '.Library'
Making 'packages.html' ... done
Loading required package: foreach
Loading required package: iterators
Loading required package: parallel

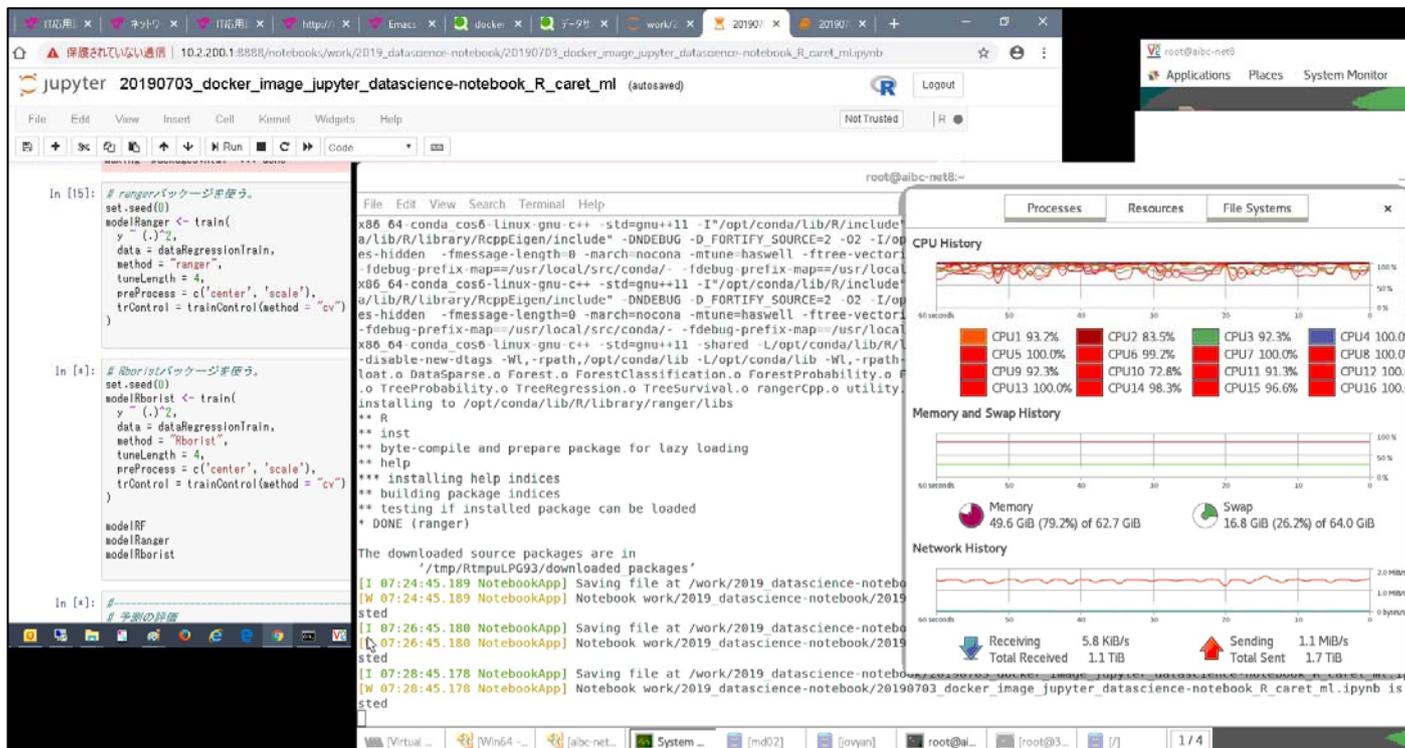
16

In [10]: cl <- makePSOCKcluster(16)
registerDoParallel(cl)

In [11]: # ニューラルネットワークによる予測
set.seed(0)
modelNnet <- train(
  y ~ (.)^2,
  data = dataRegressionTrain,
  method = "nnet",
  preProcess = c('center', 'scale'),
  trControl = trainControl(method = "cv"),
  tuneGrid = expand.grid(size=c(1:10), decay=seq(0.1, 1, 0.1)),
  linout = TRUE
)

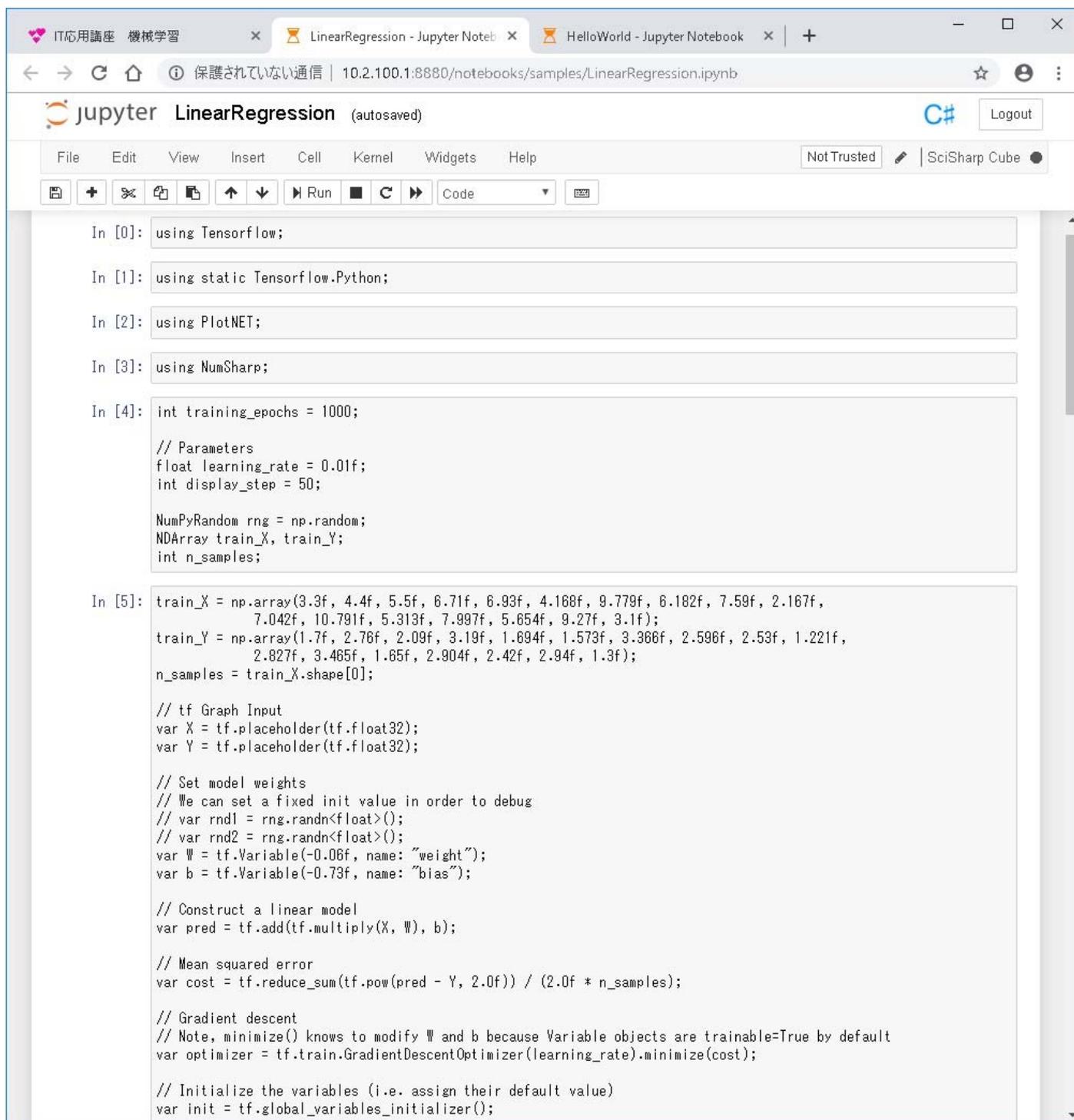
modelNnet
  
```

下図は、上記の機械学習で、CentOS 上の CPU16 コアすべてが 100%に近い負荷で動作している状態です。



## 4.2 機械学習の動作環境 Jupyter Notebook (C#カーネル)

下図は、Jupyter Notebook の C#カーネルで、対話形式に線形回帰の設定を行っている画面です。テスト環境の C#カーネルの導入は公開レジストリから行えます。



```

In [0]: using TensorFlow;

In [1]: using static TensorFlow.Python;

In [2]: using PlotNET;

In [3]: using NumSharp;

In [4]: int training_epochs = 1000;

// Parameters
float learning_rate = 0.01f;
int display_step = 50;

NumPyRandom rng = np.random;
NArray train_X, train_Y;
int n_samples;

In [5]: train_X = np.array(3.3f, 4.4f, 5.5f, 6.71f, 6.93f, 4.188f, 9.779f, 6.182f, 7.59f, 2.167f,
7.042f, 10.791f, 5.313f, 7.997f, 5.654f, 9.27f, 3.1f);
train_Y = np.array(1.7f, 2.76f, 2.09f, 3.19f, 1.694f, 1.573f, 3.366f, 2.596f, 2.53f, 1.221f,
2.827f, 3.465f, 1.65f, 2.904f, 2.42f, 2.94f, 1.3f);
n_samples = train_X.shape[0];

// tf Graph Input
var X = tf.placeholder(tf.float32);
var Y = tf.placeholder(tf.float32);

// Set model weights
// We can set a fixed init value in order to debug
// var rnd1 = rng.randn<float>();
// var rnd2 = rng.randn<float>();
var W = tf.Variable(-0.06f, name: "weight");
var b = tf.Variable(-0.73f, name: "bias");

// Construct a linear model
var pred = tf.add(tf.multiply(X, W), b);

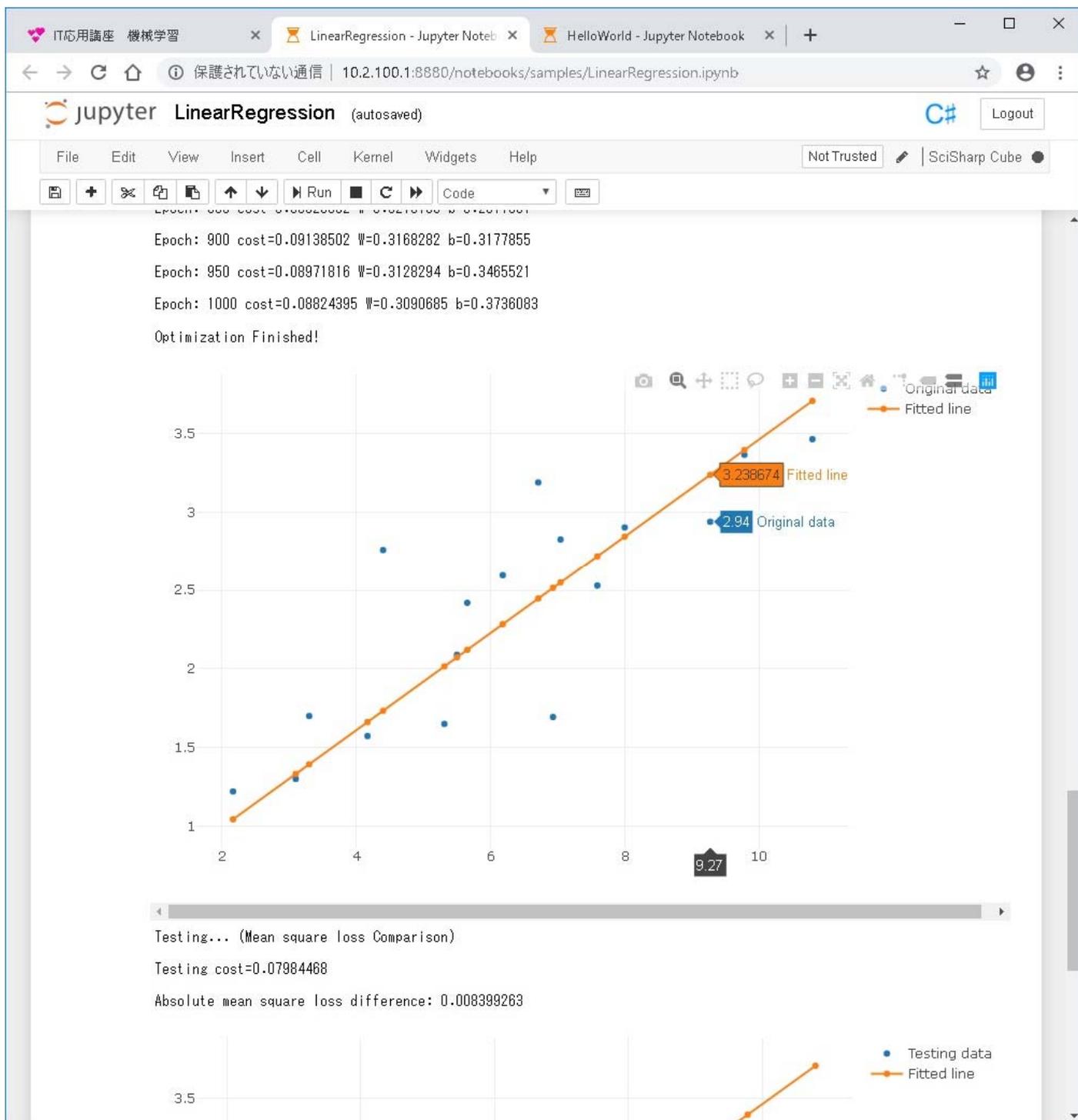
// Mean squared error
var cost = tf.reduce_sum(tf.pow(pred - Y, 2.0f)) / (2.0f * n_samples);

// Gradient descent
// Note, minimize() knows to modify W and b because Variable objects are trainable=True by default
var optimizer = tf.train.GradientDescentOptimizer(learning_rate).minimize(cost);

// Initialize the variables (i.e. assign their default value)
var init = tf.global_variables_initializer();
  
```

下図は、上記の線形回帰のテスト結果です。

Jupyter Notebook では、Web ブラウザから対話形式で各種のグラフや図形を美しく可視化し表示できます。



## 5. 参考資料

### 5.1 Web サイト

- ・ <https://www.docker.com/> …… Docker 公式サイト(ブラウザで日本語訳も可)
- ・ <https://docs.docker.jp/> …… Docker ドキュメント日本語化プロジェクト