

IRISを手軽に使う3つの方法

VSCoDe／テンプレート／パッケージマネージャ

Speaker



飯島 美穂子

教育サービス担当

mihoko.iijima@intersystems.com

トレーニングを担当しています（オンラインもやっています）。

Agenda & Benefits



IRISを手軽に使える3つの方法をご紹介します。

- VSCode から IRIS へ接続する方法
- 開発者コミュニティが公開しているIRISの開発環境テンプレートの使い方
- ObjectScript パッケージマネージャの使い方



このセッションの主な目標

- ObjectScriptエクステンションの使用方法を確認できます。
- 開発環境テンプレートを自分好みに変更する方法を確認できます。
- コミュニティサポートのパッケージマネージャを利用するための手順を確認できます。

方法その1

VSCoodeからIRISへ接続する

以下の準備ができれば接続できます。

- VSCoode
- ObjectScriptエクステンションのインストール
- settings.json の設定（接続情報の設定）

VSCodeからIRISへ接続する

ObjectScriptのエクステンションをインストールする

The screenshot shows the VS Code Extensions view. The search bar contains 'intersystems'. The search results list several extensions, with 'InterSystems ObjectScript Extension Pack' selected. The details for this extension are shown, including its name, publisher (InterSystems Developer Community), version (1.0.1), and a note that it is enabled globally. Below the details, a section titled 'Extension Pack (3)' lists three sub-extensions: 'InterSystems ObjectScript', 'InterSystems Server Manager', and 'InterSystems Language Server'. A red box highlights the search bar and the search results list, with the text '2. Intersystemsで検索' and '1. クリック' respectively. A red box highlights the 'InterSystems ObjectScript Extension Pack' in the details section, with the text '3. InterSystems ObjectScript Extension Pack を選択'. A dashed green box highlights the 'Extension Pack (3)' section, with the text '3種のエクステンションがインストールされます。'.

EXTENSIONS: MA **2. Intersystemsで検索** ObjectScript Extension Pack ×

intersystems

InterSystems ObjectScript 1.0.7
InterSystems ObjectScript language support...
InterSystems Developer Community

InterSystems Language Server 1.0.6
A language server for InterSystems Object...
InterSystems Corporation

InterSystems Server Manager 1.0.2
Helper extension for defining connections ...
InterSystems Developer Community

InterSystems IRIS 0.1.0
e SQLTools extension supporti...
InterSystems Developer Community **Install**

InterSystems ObjectScript Extensi... 1.0.1
Extensions for ObjectScript development
InterSystems Developer Community **Install**

SQLTools 0.23.0
Database management done right. Con...
Matheus Teixeira **Inst**

Serenji 3.1.3
Editor and debugger for InterSy...
George James Software

Caché Relationship Creator 0.0...
Generate Relationship between...
SwifMobi **Install**

InterSystems ObjectScript Extension Pack intersystems-community.objectscript-pack
InterSystems Developer Community | 623 | ★★★★★ | Repository | License | v1.0.1
Extensions for ObjectScript development
Disable **Uninstall** *This extension is enabled globally.*

3種のエクステンションがインストールされます。

Extension Pack (3)

InterSystems ObjectScript
InterSystems ObjectScript langu...
InterSystems Developer Com...

InterSystems Server Manager
Helper extension for defining c...
InterSystems Developer Com...

InterSystems Language Server
A language server for InterSyste...
InterSystems Corporation

1. クリック

3. InterSystems ObjectScript Extension Pack を選択

InterSystems ObjectScript Extension Pack

quip Visual Studio Code to be a powerful editor and debugger for [InterSystems](#) ObjectScript

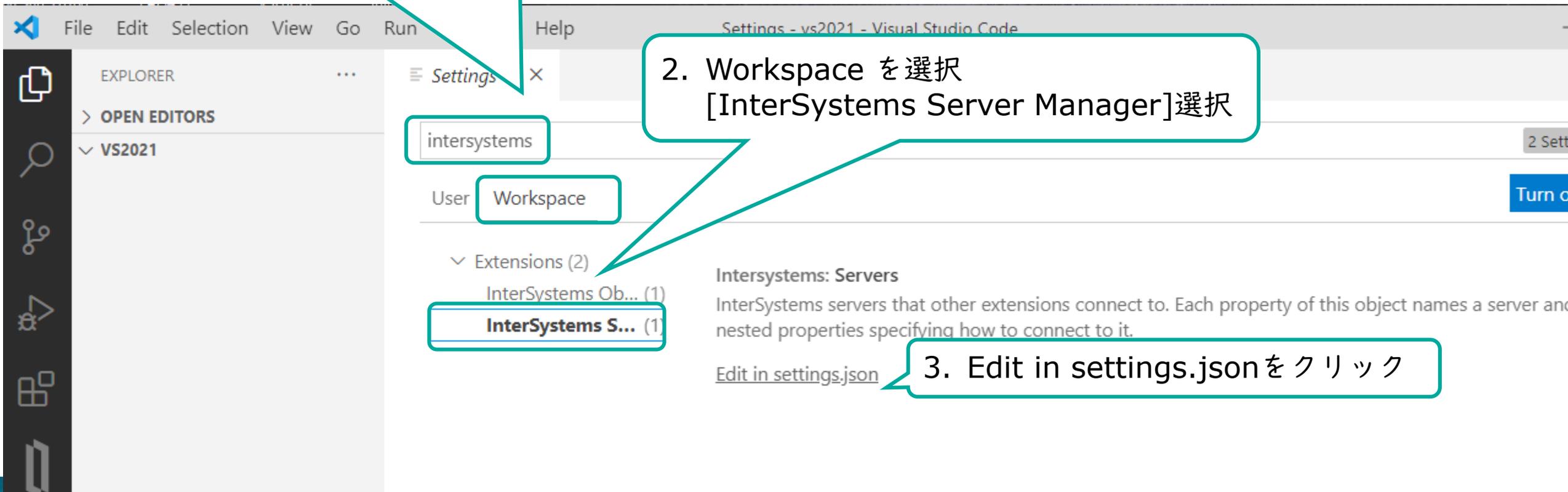
VSCodeからIRISへ接続する

接続情報を設定する (settings.jsonの作成)

1. ワークスペースのフォルダを選択した後
File > Preferences > settings >
intersystems で検索 >

2. Workspace を選択
[InterSystems Server Manager]選択

3. Edit in settings.jsonをクリック



VSCodeからIRISへ接続する

接続文字列を設定する (settings.jsonの修正)

```
Settings settings.json ×  
.vscode > settings.json > intersystems.servers > tryiris > webServer > # por
```

```
1 {  
2   "objectscript.conn": {  
3     "active": true,  
4     "ns": "USER",  
5     "server": "tryiris"  
6   },  
7   "intersystems.servers": {  
8     "tryiris": {  
9       "webServer": {  
10        "scheme": "http",  
11        "host": "127.0.0.1",  
12        "port": 52773  
13      },  
14      "username": "_system"  
15    }  
16  }
```

“intersystems.servers”の接続情報は複数作成できます。アクセスするときに、“objectscript.conn”の“server”プロパティを変更するだけで接続先が簡単に変更できます。

“objectscript.conn”プロパティでは、接続するIRISの名称 (“intersystems.servers”で作成したサーバ名) と、ネームスペース名を指定します。

“intersystems.servers”プロパティに接続したいIRISの名称を設定します (例ではtryiris)。“webServer”プロパティに接続先IRISのホスト名 (host) とWebサーバポート (port) を指定し、“username”プロパティに接続時に使用するユーザー名 (_system) を指定します。

補足：ネームスペース＝仮想の作業環境

作業環境で何を作る？

- スキーマ（クラス／テーブル）
- ロジック（メソッド／ストアド）
- データ（永続オブジェクト／レコード）

作成した内容は永続的に利用したい・・・

⇒データベースを利用したい

補足：データベースの中身

スキーマ（クラス／テーブル）

データ（永続オブジェクト／レコード／グローバル）

ロジック（メソッド／ストアドプロシージャ／ルーチン）

上記全ての内容がデータベースに格納されます。

補足：ネームスペースとデータベースの関係

ネームスペースは、仮想の作業環境で使いたいデータベースを指定する論理定義

データベースは、データやロジック、クラスやテーブルのスキーマを格納する場所

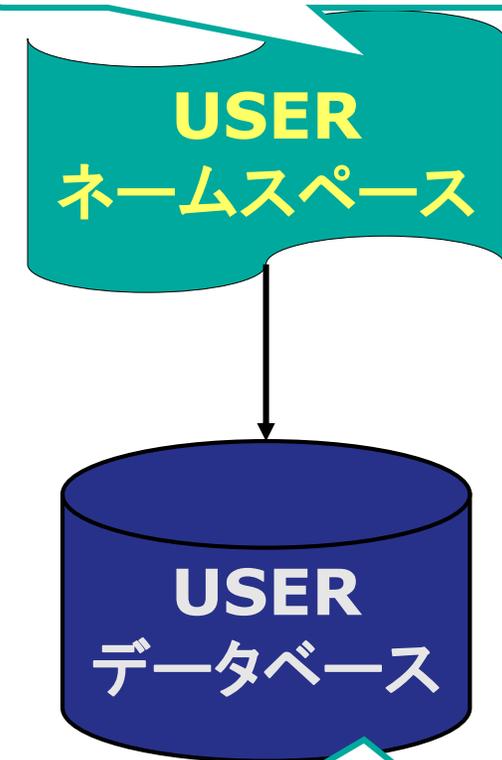
ネームスペースを特定する



データベースが特定できる

使いたい情報をすべて利用できる

ネームスペース/データベース名には英字から始まる2文字以上の英数字を組み合わせた文字列を指定します。



ネームスペースとデータベースの名称を同一名に設定するルールはありませんが、慣習としてよく利用される設定方法です。

実演1

VSCodeからIRISへ接続してみます

- WindowsにインストールしているIRISに接続します。
- ObjectScriptの記述例（クラス／ルーチン）は開発者コミュニティの記事もご参照ください。
jp.community.intersystems.com/node/482976

方法その2

開発環境テンプレートを使う

以下の準備が整ったら利用できます。

- Git/VSCoDe/Dockerのインストール
- テンプレートのダウンロード (git clone)

開発環境テンプレートとは

<https://jp.community.intersystems.com/node/487581>

コンテナビルド時にIRISの開発環境を自動で設定する仕組みを含めたもの

昨年、開発者コミュニティでは、IRISと異なる技術とを組み合わせでプログラミングの腕を競い合う「**IRISプログラミングコンテスト**」を開催しました。

<https://jp.community.intersystems.com/tags/contest>

- 4月 REST API
- 5月 Native API アプリケーション
(Native APIとは、Python/Node.js/Java/.NETからキーバリュータイプで操作できるアクセス方法)
- 6~7月 AI/MLソリューション
(IRISのAutoMLを利用する方法/IRISからPythonを実行できるPythonGatewayを利用する方法)
- 8月 FHIRアプリケーション
- 9月 フルスタックアプリケーション
(IRISをデータサーバとして任意のフレームワークを利用してWeb/デスクトップ/モバイルUIを作成)
- 10~11月 Interoperability (相互運用性) を使用したソリューション、またはメンテナンスを支援するソリューション
- 12月 IRISの分析機能 (IRIS BI、IRIS NLP、IntegratedML、InterSystems Reports) を使用した分析ソリューション
- 1月 IRISのマルチモデルアクセスの中から2つのモデルでアクセスするアプリケーション

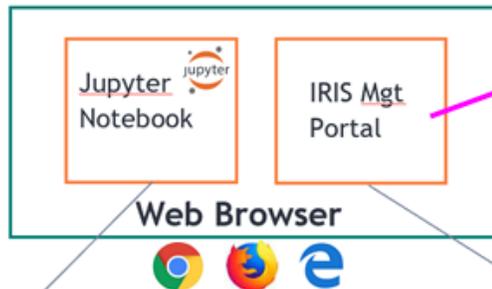
コンテストは終了しましたが、テンプレートは自由にご利用いただけます。

コンテストのテンプレート 一部ご紹介

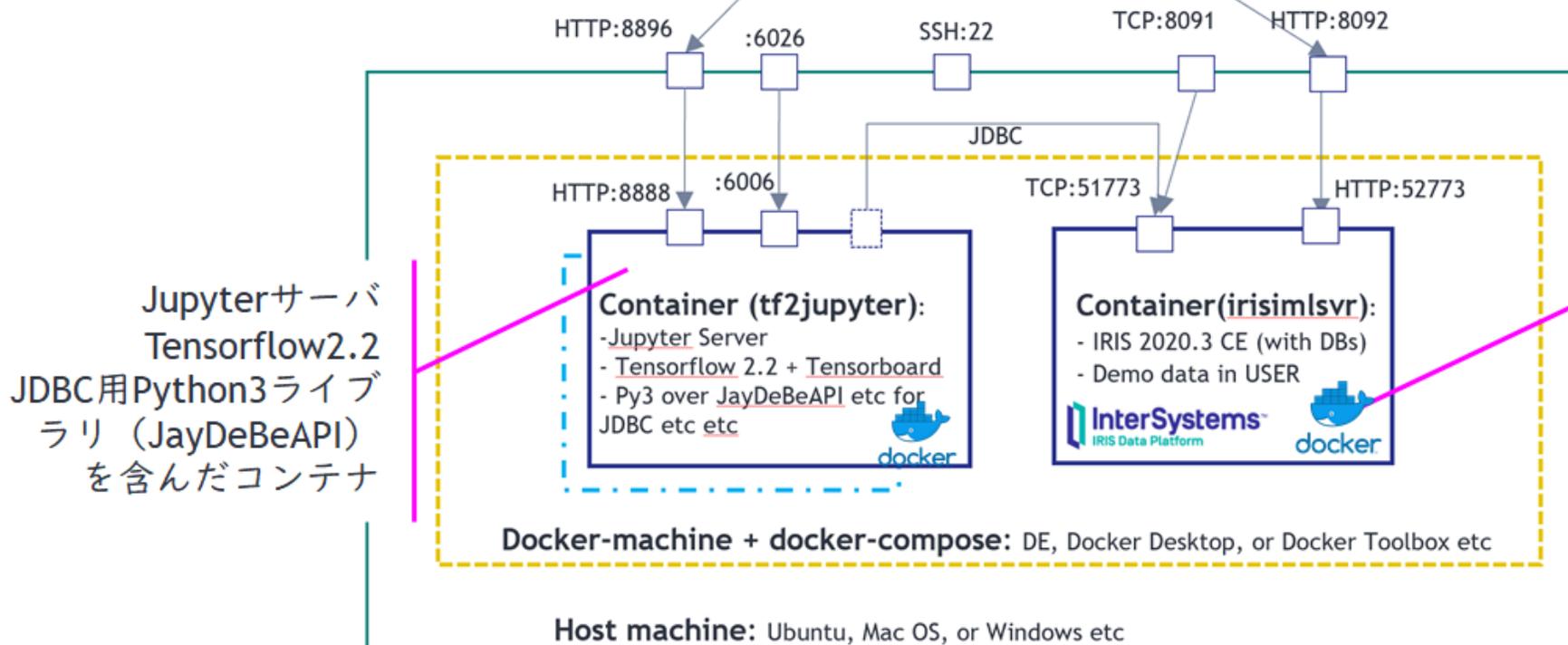
開催月のテーマに沿った技術とIRISがすぐに利用できる環境をテンプレートとして用意していました。

6~7月開催コンテストテンプレート 図解

テンプレートの提供内容は開発環境の**土台のみ**
(=データベースとその他必要な言語の実行環境/モジュールの準備)



参加者はブラウザからIRISの管理ツールやJupyter Notebookにアクセスできます。具体的なコード記述はテンプレート内のJupyter Notebookを利用したり、お好みの利用方法を選択したり**自由**です。



Jupyterサーバ
Tensorflow2.2
JDBC用Python3ライブラリ (JayDeBeAPI) を含んだコンテナ

データベース (IRIS) 用コンテナ Docker hub から [InterSystems IRIS Community Edition](#) のイメージを pull

シンプルなテンプレートもあります

openexchange.intersystems.com/package/objectscript-docker-template

IRISの開始とサンプルコードのインポートだけを行うテンプレートもあります。

objectscript-docker-template

[DETAILS](#) [RELEASES](#) [ISSUES](#)

⚠ This application is not supported by InterSystems Corporation. Use it at your own risk.

What's new in this version

iris.script path fixed, docker volume and build paths changed

intersystems-objectscript-template

This is a template for InterSystems ObjectScript Github repository. The template goes also with a few files which let you immedietly compile your ObjectScript files in InterSystems IRIS Community Edition in a docker container

docker-compose.yml

Dockerfile使用
ポート番号の指定
ボリュームのマウント指定

ビルド時使用

Dockerfile

ビルド時にiris.scriptに記載されたIRISのコマンドを実行します。

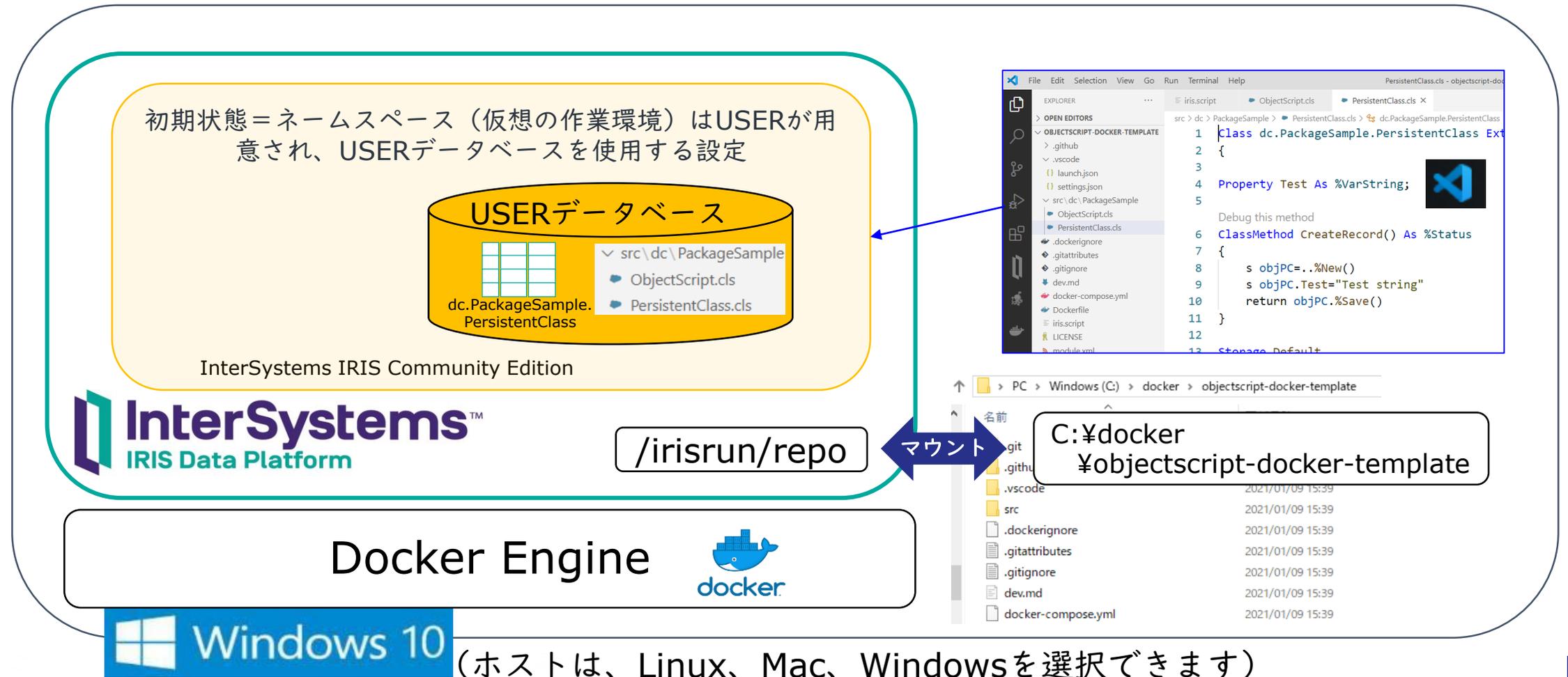
IRISのスクリプト
(iris.script)

- Dockerfileを利用して以下の作業を行っています。
 - 事前定義ユーザのパスワードを無期限に（コンテナ版IRIS初回アクセス時に初期パスワード変更画面が表示されますが、無期限に変更することで初期パスワード SYS のままアクセスできます）
 - サンプルのソースコードをインポート

シンプルなテンプレートもあります

openexchange.intersystems.com/package/objectscript-docker-template

イメージ図

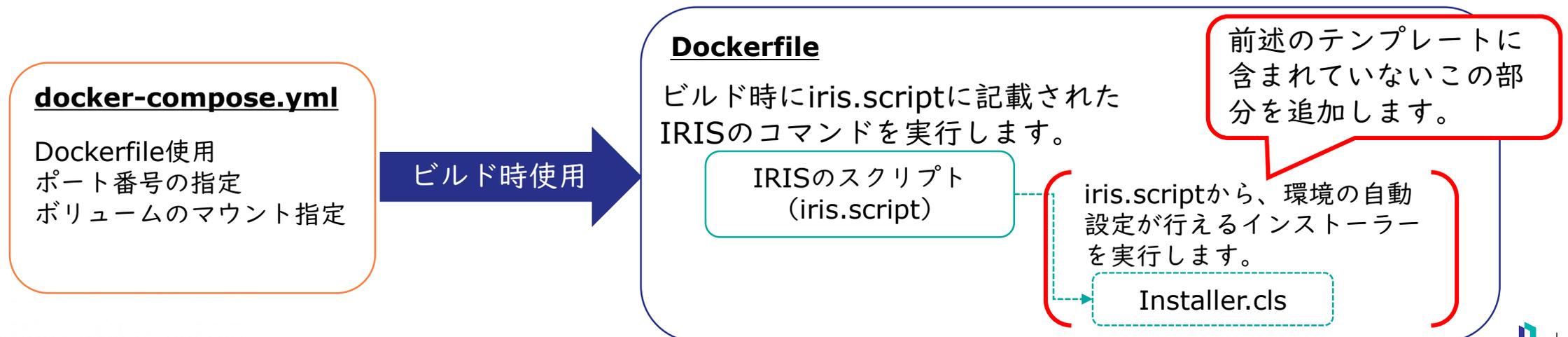


シンプルなテンプレートを真似して 好みの設定に変更してみる

任意名のネームスペースを追加し、オリジナルのソースコードをインポートします。

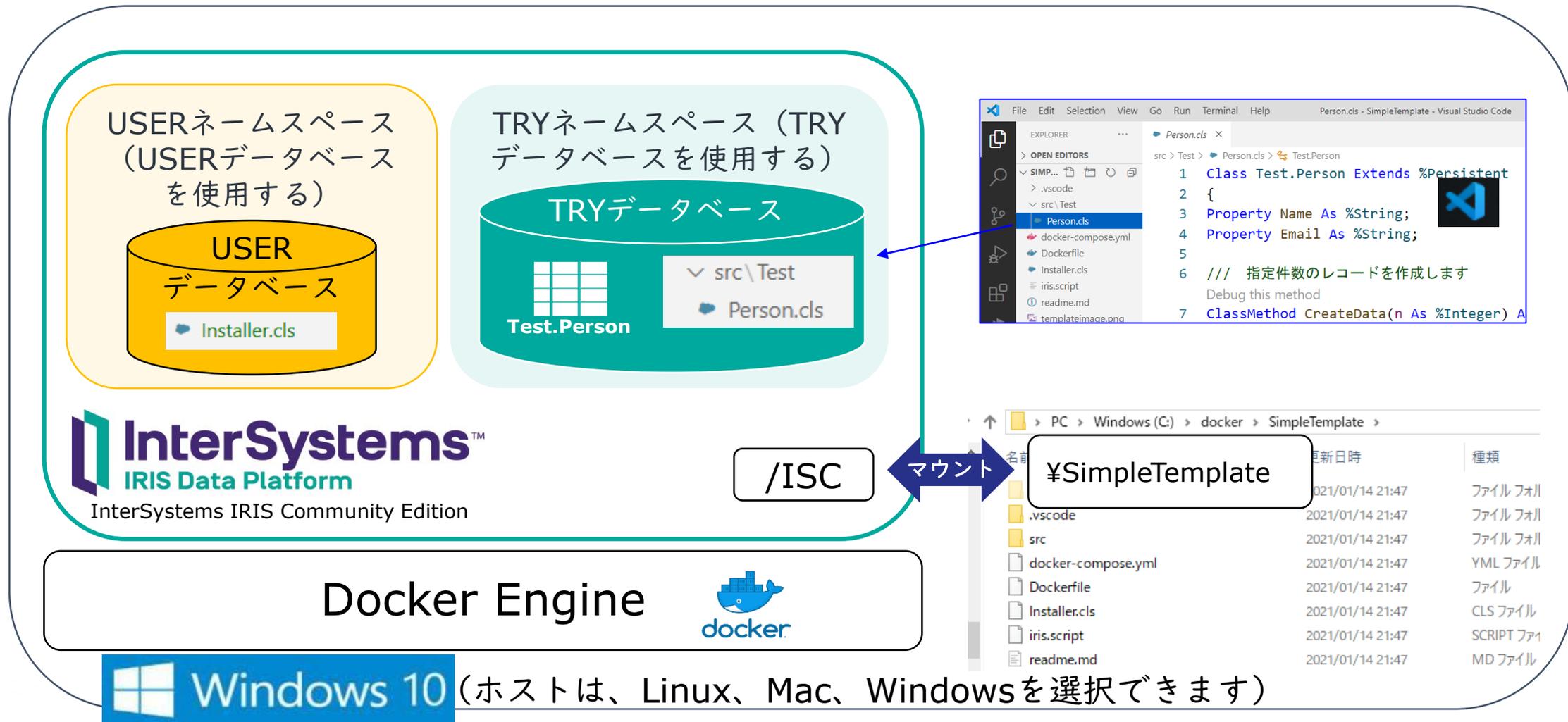
- Dockerfileに以下の変更を加えます。
 - IRISのインストーラークラス（構成設定を自動化できるクラス）のコードをコンテナにコピーし、インポートします。
 - インストーラーを実行しネームスペースを作成します。
- ご紹介内容の完成品は、開発者コミュニティ用Gitで公開しています。

<https://github.com/Intersystems-jp/SimpleTemplate>



シンプルなテンプレート 加工後

イメージ図



Dockerfileの中身

```
ARG IMAGE=intersystemsc/iris-community:2020.4.0.524.0-zpm
ARG IMAGE=store/intersystems/iris-ml-community:2020.3.0.304.0
FROM $IMAGE
```

```
USER root
```

```
# コンテナ内のワークディレクトリを /opt/try に設定 (後でここにデータベースを作成予定)
```

```
WORKDIR /opt/try
```

```
RUN chown ${ISC_PACKAGE_MGRUSER}:${ISC_PACKAGE_IRISGROUP} /opt/try
```

```
USER ${ISC_PACKAGE_MGRUSER}
```

```
# ファイルのコピー
```

```
COPY Installer.cls .
```

```
COPY src src
```

```
COPY iris.script iris.script
```

```
# iris.scriptに記載された内容を実行
```

```
RUN iris start IRIS ¥
```

```
&& iris session IRIS < iris.script ¥
```

```
&& iris stop IRIS quietly
```

iris.script の中身

(テンプレート内でIRISのコマンド実行に使用しているファイル)

```
// インストーラーの実行 (ネームスペースTRYの作成のみ)
// コンテナ内で実行するので、Installer.clsを配布したコンテナ内のディレクトリを指定します。
do $SYSTEM.OBJ.Load("/opt/try/Installer.cls", "ck")
// インストーラーの実行
set sc = ##class(App.Installer).setup()
// 作成したネームスペースに移動
set $namespace="TRY"
// アプリケーション用ソースなどあればここで一括インポートできます。
// src以下に配置したフォルダやファイルがコンテナ内では /opt/try/srcにコピーされています。
// 別環境でクラスやルーチン、グローバルをエクスポートしたXMLファイルもインポートできます。
do $System.OBJ.LoadDir("/opt/try/src", "ck", ,1)

// システムの設定を変更するため、%SYSネームスペースに移動します
set $namespace="%SYS"
// 事前定義ユーザのパスワードを無期限に設定する (デフォルトパスワードはSYS)
Do ##class(Security.Users).UnExpireUserPasswords("*")
// 日本語ロケールに変更 (コンテナがUbuntu英語版のためデフォルトは英語ロケール) を利用
Do ##class(Config.NLS.Locales).Install("jpuw")
halt
```

ソース用ディレクトリ
はコンテナビルド時
/opt/try/src

```
21 RUN iris start IRIS \ Dockerfileの一部抜粋
22   && iris session IRIS < iris.script \
23   && iris stop IRIS quietly
```

インストーラーについて

インストーラー詳細は開発者コミュニティでご紹介しています：jp.community.intersystems.com/node/478966

構成設定を自動化できる機能で、設定したい内容をXMLで定義します。

```
<Manifest>
```

```
<!--インストーラーの中で使用する変数NamespaceにTRYを設定 -->
```

```
<Default Name="Namespace" Value="TRY"/>
```

```
<!--変数appにWebアプリケーションパス名を代入：例ではtry -->
```

```
<Default Name="app" Value="try" />
```

```
<!--ネームスペース作成用の定義 -->
```

```
<Namespace Name="${Namespace}" Code="${Namespace}" Data="${Namespace}" Create="yes" Ensemble="1">
```

```
<!-- データベース作成用定義-->
```

```
<Configuration>
```

```
<!--データベースディレクトリをDockerfile内で予め作成した /opt/try 以下に設定するように指定 -->
```

```
<Database Name="${Namespace}" Dir="/opt/${app}/data" Create="yes" Resource="%DB_${Namespace}"/>
```

```
</Configuration>
```

```
<!--ウェブアプリケーションパス /csp/try の作成 -->
```

```
<CSPApplication Url="/csp/${app}" Directory="${cspdir}${app}" ServeFiles="1" Recurse="1"
```

```
MatchRoles=":%DB_${Namespace}" AuthenticationMethods="32"/>
```

```
</Namespace>
```

```
</Manifest>
```

実演2

テンプレートを動かしてみます。

- 自分好みの設定を追加したテンプレートを使い、コンテナを開始します。

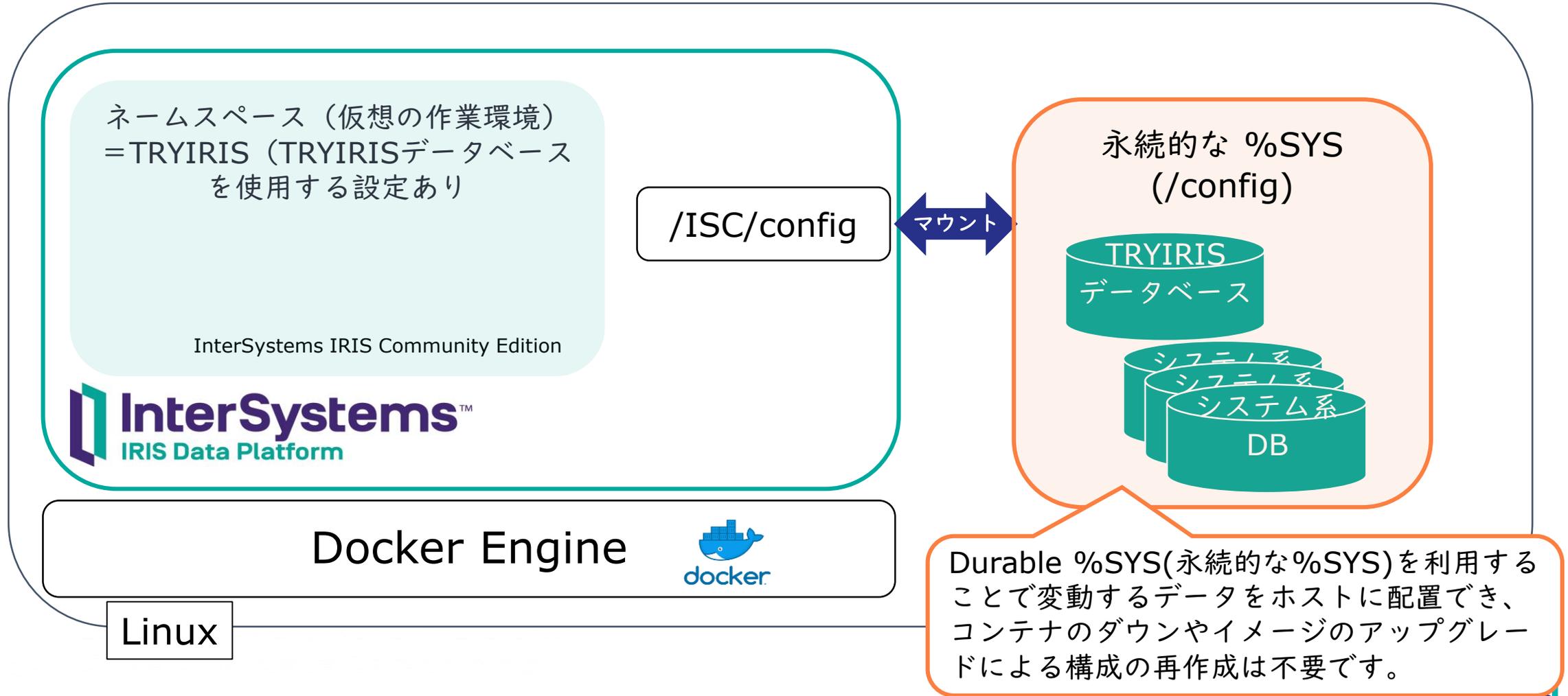
<https://github.com/Intersystems-jp/SimpleTemplate>

- インストーラーについて詳細は開発者コミュニティの記事やドキュメントをご参照ください。
 - jp.community.intersystems.com/node/478966
 - [ドキュメント](#)

コンテナ版 IRIS をさらに便利に使う

詳細は[TE7： Dockerコンテナ版InterSystems IRIS data platform の勧め]でご紹介しています。

ご参考：[永続的な %SYS の利用](#)



方法その3

jp.community.intersystems.com/tags/objectscript-package-manager-zpm

ObjectScriptパッケージマネージャ

- パッケージマネージャはコミュニティがサポートする機能です。
- 利用方法は以下の通りです。
 - クライアント機能のソースコードをIRISにインポート
 - zpmコマンドを利用してパッケージをインストール

ObjectScriptパッケージマネージャ

方法：Open Exchangeからクライアント機能をダウンロードしてIRISにインポートするだけ

DEVELOPER ENVIRONMENT ★13



ObjectScript Package Manager
The Client for InterSystems
ObjectScript Package Manager. ZPM

openexchange.intersystems.com



ObjectScript Package Manager Client - ZPM

chat on telegram

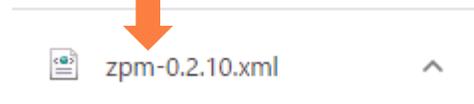
Helps to install ObjectScript, CSP and Frontend packages into InterSystems IRIS published on [ZPM Registry](#)

Installing ObjectScript Package Manager Client:

1. Download the **latest version** of zpm from the registry
2. Import the zpm.xml into IRIS and compile via any desired way (Management Portal, Studio or Terminal)

After that you can use PackageManager to install modules from [community repository](#) in any namespace.

3. Check if you call a zpm in command line and get the following: USER>zpm



メモ：パッケージマネージャを含んだコミュニティエディションのコンテナ版IRISも公開しています。

ObjectScriptパッケージマネージャ

サンプルアプリ (csvgen) のインストール

```
USER>zpm
zpm: USER>list

zpm: USER>search
registry https://pm.community.intersystems.com:
analyzethis          1.1.4
appmsw-telestat     1.0.2
                   省略

zpm: USER>install csvgen
```

```
[sslclient] Reload START
[sslclient] Reload SUCCESS
[sslclient] Module object refreshed.
                   省略

[csvgen] Configure START
[csvgen] Configure SUCCESS
[csvgen] Activate SUCCESS
zpm: USER>
```

```
test.csv
1 Name,Email
2 山田太郎,taro@mail.com
3 鈴木花子,hanako@mail.com
```

```
zpm: USER>quit

USER>set file="/ISC/test.csv"

USER>set st=##class(community.csvgen).Generate(file,"","Test2.Person")

Class name: Test.Person
Header: Name VARCHAR(250),Email VARCHAR(250)
Records imported: 2
USER>
```

実演3

パッケージマネージャを使ってみます

- クライアント機能のインストール
- サンプルアプリケーション (csvgen) のインストールとCSVのインポート

Key Takeaway

- VSCode に ObjectScript のエクステンションを追加し接続情報を設定するだけで、IRISへ接続できることが確認できました。
- 開発者コミュニティが提供する開発環境テンプレートを使えば、手軽に開発環境のベースを作成できることが確認できました。
 - コンテナの破棄、再作成などが簡単に行えるので、新機能を手軽にお試しいただけます。
- ObjectScript パッケージマネージャを利用することで、ちょっとしたサンプル／機能の追加を手軽に行えることが確認できました。

※ObjectScriptパッケージマネージャは開発者コミュニティがサポートする機能です。

Next Steps

TE7 : Dockerコンテナ版InterSystems IRIS data platform の勧め

TE11 : 開発者コミュニティの活用法

- 開発者コミュニティ jp.community.intersystems.com
 - これからIRISを体験する方向け (beginnerタグ)
jp.community.intersystems.com/tags/beginner
 - InterSystems 製品を初めて使用する方向け学習コンテンツなど
jp.community.intersystems.com/node/484466
 - ヒントとコツ (tips-tricksタグ)
jp.community.intersystems.com/tags/tips-tricks
 - 相互運用性 (interoperabilityタグ)
jp.community.intersystems.com/tags/interoperability
- TRY IRIS (www.intersystems.com/jp/TRY)
メールをご登録いただくだけで利用できるIRIS (プライベートクラウド上にご登録者別環境をご用意しています)

InterSystems 開発者コミュニティ

jp.community.intersystems.com

ぜひ、ご活用ください！

- **開発者同士の交流の場**として
技術的な質問&回答が行えます！
- **ヒントを探す場所**として
jp.community.intersystems.com/tags/tips-tricks
- **学びの場**として
セルフラーニングビデオ公開中！
jp.community.intersystems.com/tags/beginner