



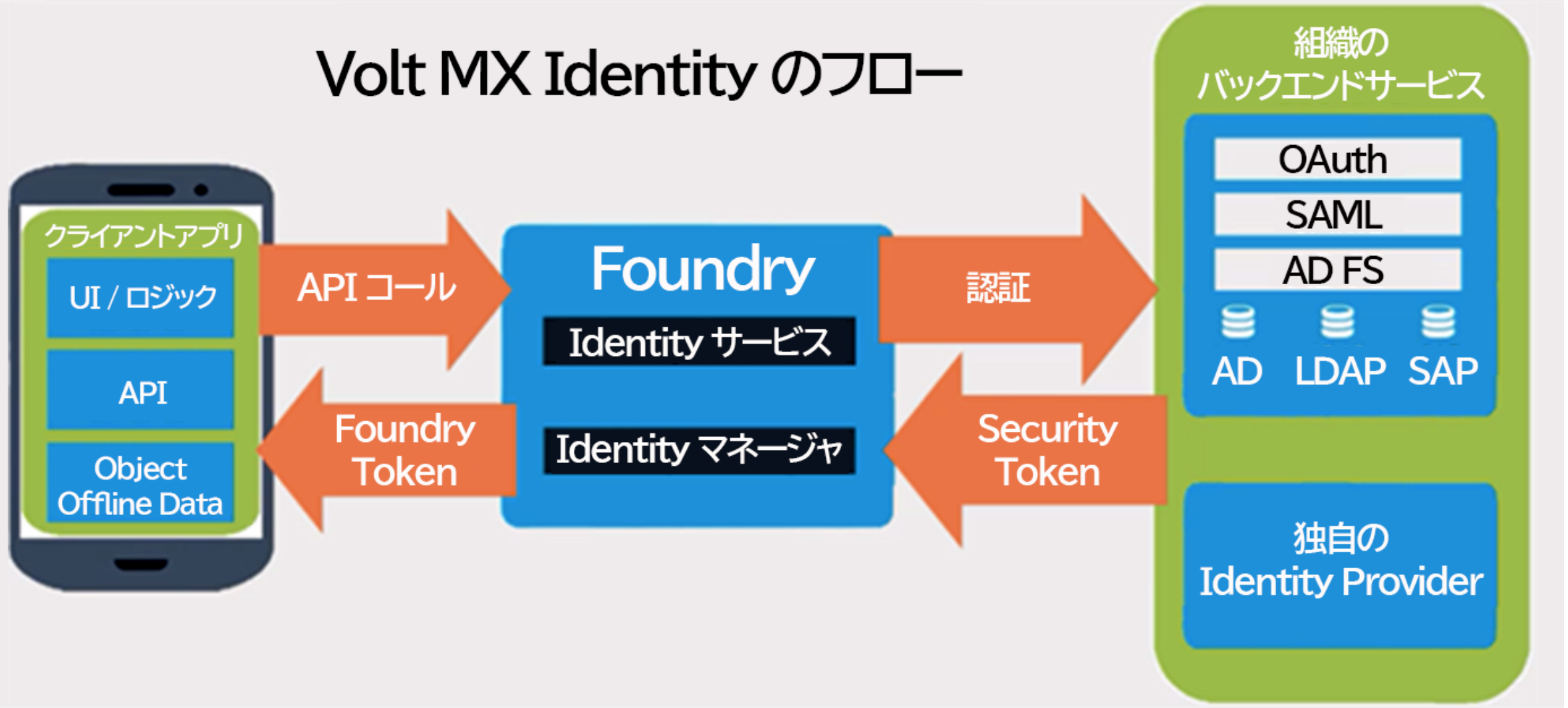
HCL Volt MX

Identity サービスの概要


I

Identity 概要

Volt MX Identity のフロー



Volt MX Foundryへのアクセス



HCL Volt MX


Sign in to your account

Email

Password

User Store

SIGN IN



HCL Volt MX

Sign in to your account

Email

Password

[Forgot Password?](#) **SIGN IN**

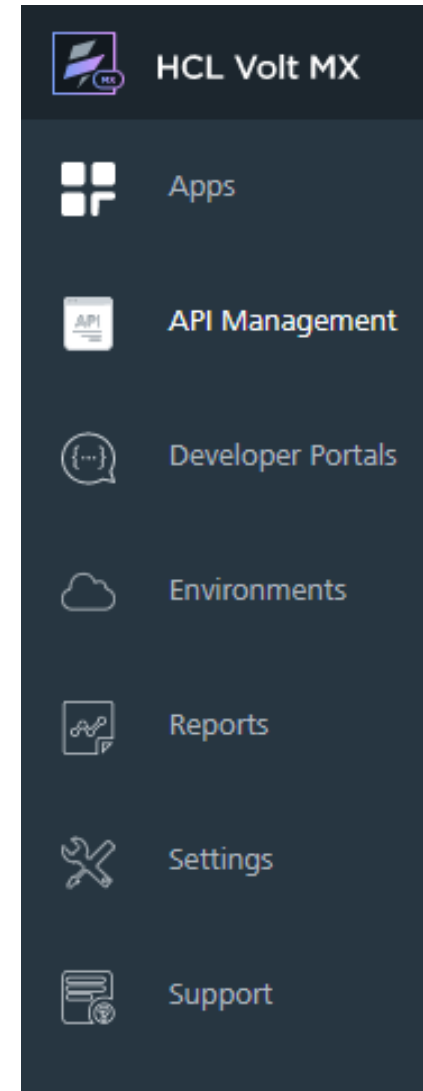
[Create Account](#)

オンプレ: <http://hclvoltmx:8080/mfconsole>

クラウド: <https://manage.demo-hclvoltmx.com/>

Foundry Console へのアクセス

- 左側のペインにマウスオーバーすると、Foundryのさまざまな機能が表示されます。
- **Apps**
 - 新しいアプリを作成したり、既存のアプリをインポートしたり、既存のアプリのリストを表示できます。
- **API Management**
 - アプリサービス(ID、統合、オーケストレーション)をアプリ内でリンクまたは構成せずに構成および管理(作成、編集、削除)することができます。
- **Developer Portals**
 - Volt MX Foundryを使用して作成されたAPIを公開するためのポータルを作成できます。社内外のパートナーチームの開発者は、作成されたポータルにアクセスしてAPIを探索し、テストできます。



Foundry Console へのアクセス（続き）

- **Environments**

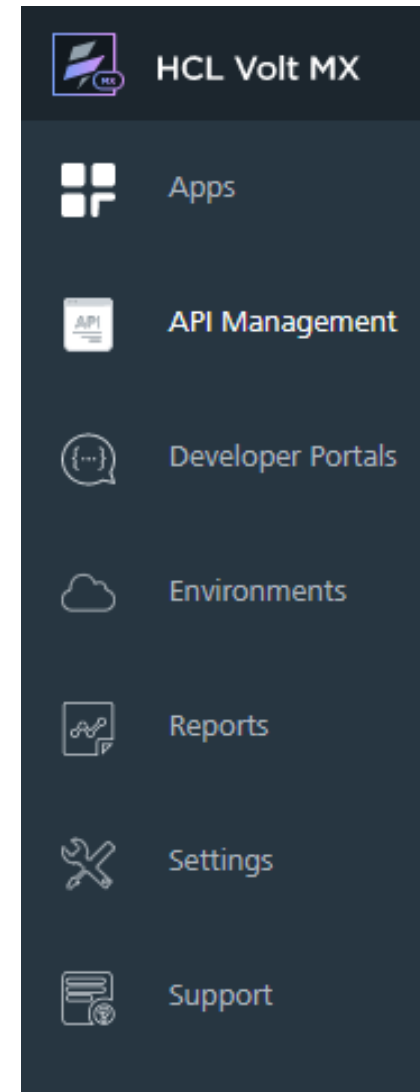
- アプリを公開するための環境を作成する必要があります。環境には、Volt MX Foundry Integration、Volt MX Foundry Engagement、Volt MX Foundry Sync、およびVolt MX Foundry Managementなど、少なくとも1つのサーバーまたはすべてのサーバーの組み合わせが含まれます。

- **Settings**

- 設定を使用して、スーパーユーザーは、IDプロバイダの構成、ユーザーの追加、新しいユーザーとグループのインポート、ユーザーへのロールの割り当て、ユーザーとグループの削除、プロキシサーバーの構成、およびレポートサーバーの構成によってタスクを管理できます。

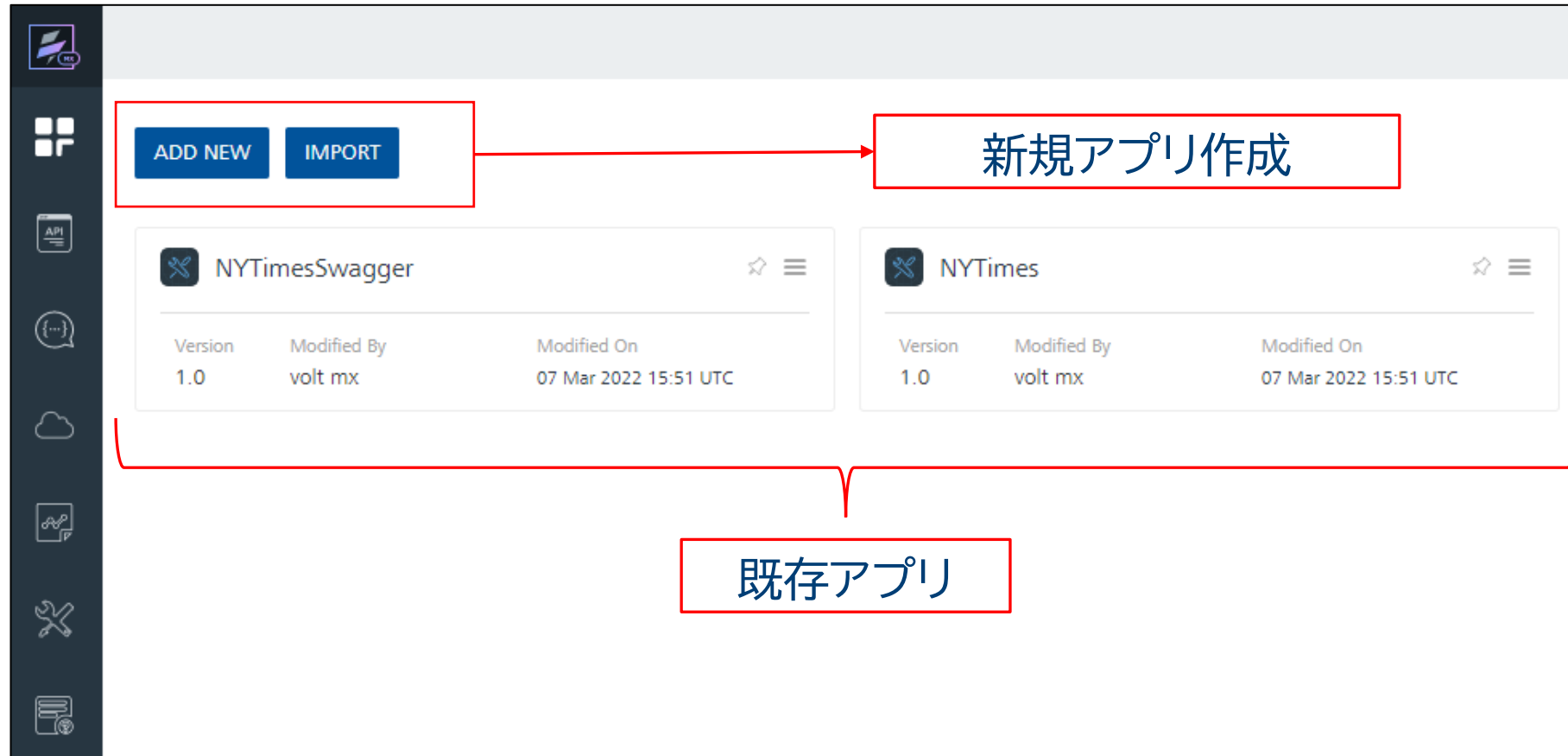
- **Support**

- サポートタブには、さまざまなグループがリストアップされているbasecampのページが表示されます。これらのグループを使用すると、開発者は、問題の任意のキッドの間にサポートを受けられます。



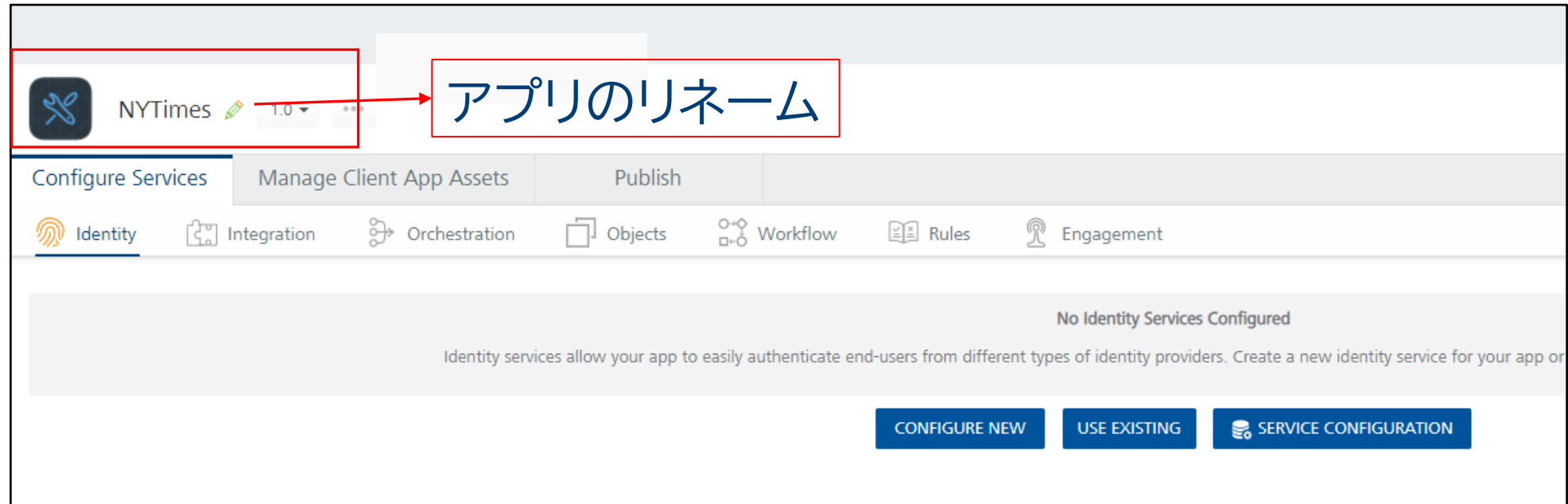
新規アプリの作成

- コンソールにログイン後、ADD NEW または IMPORT ボタンをクリックして、新しいアプリを作成します。



アプリのリネーム

- アプリはデフォルト名で作成されます。
- スクリーンショットに示すように、この名前を変更できます。





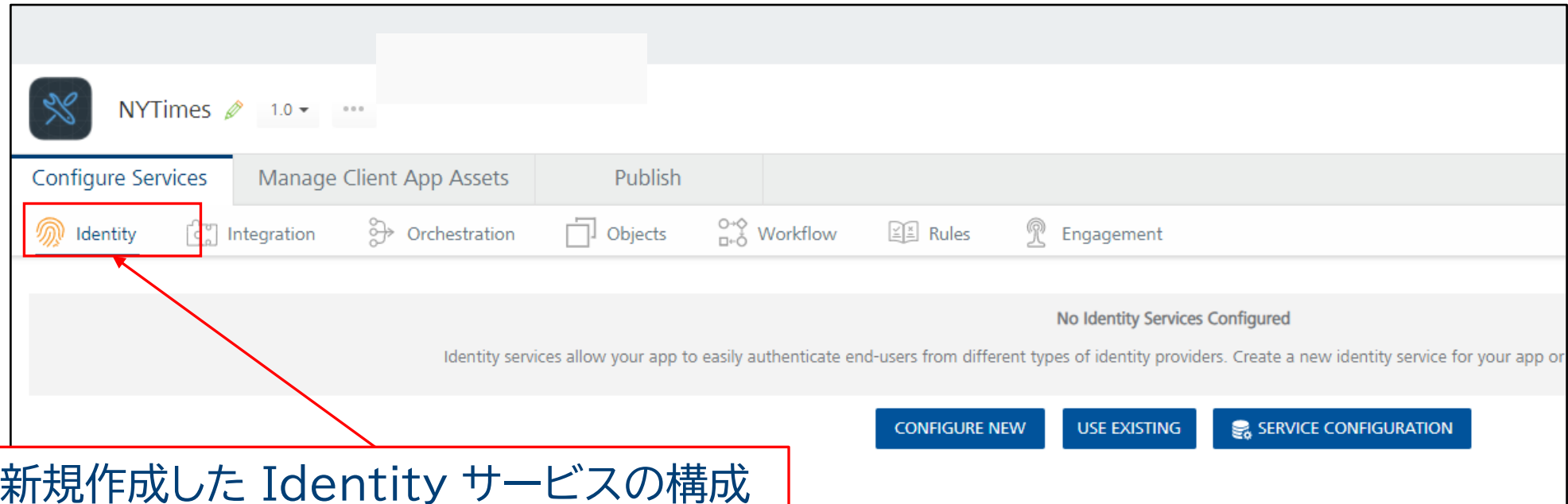
HCL Volt MX

Identity サービス Foundry Apps の構成

I

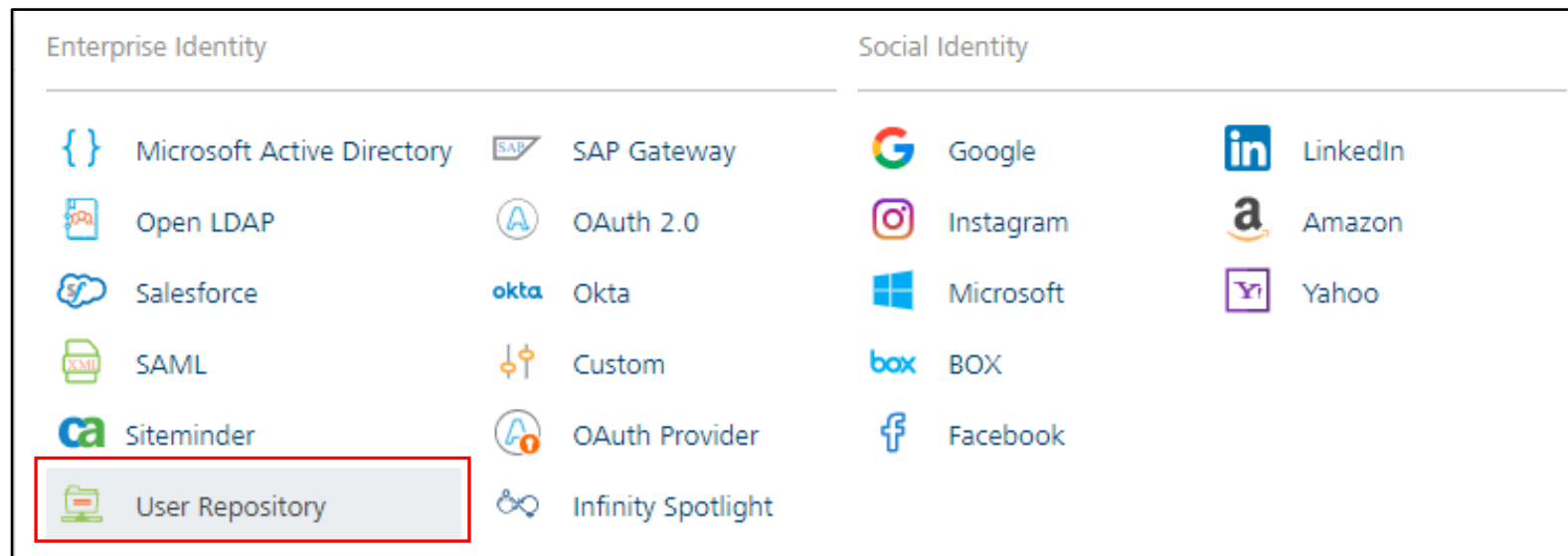
Identity サービスの構成

- Identity サービスを作成するには、Configure Services → Identity にある CONFIGURE NEW ボタンをクリックします。



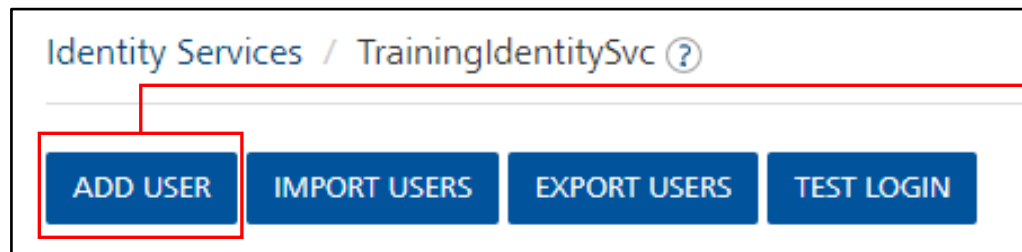
Identity サービスの構成

- Foundry は、以下のスクリーンショットに示すように、一連のエンタープライズおよびソーシャルの Identity サービスをサポートしています。
- このトレーニングでは「User Repository」ID タイプを使用しますが、すべてのアダプタは同様の方法で動作します。
- User Repository Identity プロバイダを使用すると、外部の Identity Service Provider に依存することなく、Foundry を使用してユーザーとグループを管理するための認証装置として機能させることができます。



Identity サービスの構成（続き）

- IDサービスを作成後、ADD USER ボタンでユーザーを追加します。以下のように、必要な情報を入力し、「ADD USER」ボタンをクリックします。



Add User

Email/Username*

Please enter a valid email

First Name*

Last Name*

Phone

Password * ?

Re-Enter Password *

Add / Select Group (s)

Members

CANCEL

ADD USER

Identity サービスの構成（続き）

- TEST LOGIN ボタンをクリックして、ユーザー名とパスワードを入力し、SIGN IN ボタンをクリックします。

The image shows a screenshot of the Identity Services interface. On the left, a panel titled "Identity Services / TrainingIdentitySvc" contains four buttons: "ADD USER", "IMPORT USERS", "EXPORT USERS", and "TEST LOGIN". The "TEST LOGIN" button is highlighted with a red rectangle. A red arrow points from this button to a "Test Login" dialog box on the right. The dialog box has a title bar with "Test Login" and a close button. It contains two input fields: "User ID *" with the value "foundry@hcl.com" and "Password *" with masked characters ".....". At the bottom of the dialog are "CANCEL" and "SIGN IN" buttons.

Identity サービスの構成（続き）

- これで、サーバーからの応答が表示されます。
 - Profile
 - profile_attributes
 - claims_token
 - provider_token
 - refresh_token
- 次のステップでは、このアプリを公開し、クライアントアプリでこのアプリを利用できるようにします。

```
Identity Response

1  {
2    "profile": {
3      "userid": "foundry@hcl.com",
4      "email": "foundry@hcl.com",
5      "firstname": "Foundry",
6      "lastname": "Student",
7      "user_attributes": {
8        "groups": [
9          "Members"
10       ],
11       "user_id": "foundry@hcl.com",
12       "phone": ""
13     },
14     "profile_attributes": {
15       "firstname": "Foundry",
16       "userid": "foundry@hcl.com",
17       "email": "foundry@hcl.com",
18       "lastname": "Student"
19     }
20   },
21   "profiles": {
22     "TrainingIdentitySvc": {
23       "userid": "foundry@hcl.com",
```



HCL Volt MX

Identity サービス Foundry Apps のパブリッシュ

I

アプリの公開

- アプリを公開するには、Publish タブで環境を選択し、PUBLISH ボタンをクリックします。

The screenshot displays the HCL Software Academy interface for managing and publishing applications. The top navigation bar includes 'Configure Services', 'Manage Client App Assets', and 'Publish' (highlighted with a red box). Below this, the 'Service & Web Client' and 'Native Client' tabs are visible. The main content area shows the application 'hclvoltmx' with a 'DEFAULT VERSION' of '1.0'. The 'CONSOLES' section contains several icons. The 'SERVICE CONFIG' is set to 'Custom'. The 'APP STATUS' is 'Published' with a green checkmark and the timestamp '07 Mar 2022 21:58 UTC'. The 'ENVIRONMENT STATUS' is 'Available'. The 'HISTORY & ROLLBACK' section shows 'Snapshots'. At the bottom right, there are three buttons: 'UNPUBLISH', 'CONFIGURE & PUBLISH', and 'PUBLISH' (highlighted with a red box).

アプリの公開（続き）

- Publish ボタンをクリックすると、アプリのライフサイクルの公開が開始されます。アプリの公開ライフサイクルには、いくつかのアプリ公開ステージがあります。
- アプリが正常に公開されると、Foundry は以下のようなアプリ情報を表示します。

The screenshot shows the Foundry Publish interface. The top navigation bar includes 'Configure Services', 'Manage Client App Assets', and 'Publish'. The 'Publish' tab is active. Below the navigation bar, there are tabs for 'Service & Web Client' and 'Native Client'. The main content area shows a console for 'hclvoltxmx' with a 'CONSOLES' section. A red box highlights a message box with the following content:

✓ Published **NYTimes (v1.0)** to **hclvoltxmx**

✓ Skipping publish of Engagement Services as they are not configured

✓ Updated publish resources successfully


App Information

Primary App Key	71cf4efb186f471d55e08f16ee8b99a3
Primary App Secret	4c2f1840e8e4ecc23ec6bbf0d4e23c7a
Secondary App Key	8c814bc9daae342a4093770dcce25573
Secondary App Secret	c05a457977475dcf3f4b200c77ba8843
Service URL	http://hclvoltxmx:8080/authService/100000002/appconfig

アプリの公開（続き）

- App key, App secret の詳細が以下のように表示されます。

App Key

 SERVICE URL
<http://hclvoltmx:8080/>

App Key & Secret ?

		Modified On	Modified By
Primary			
Key	71cf4...	03 Mar 2022 17:15:40 UTC	volt mx
Secret	4c2f1...		
Deprecate			
Secondary			
Key	8c814...	03 Mar 2022 17:15:40 UTC	volt mx
Secret	c05a4...		
Deprecate			

CLOSE



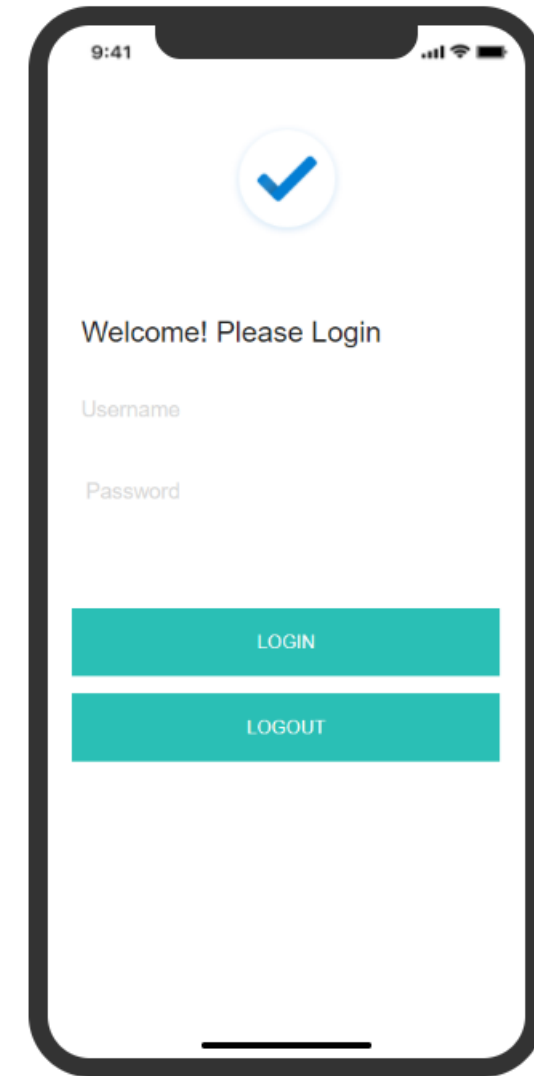
HCL Volt MX

Iris を使用した Identity サービス

I

Iris を使用した Identity Service のコール

- Foundry で ID サービスを作成したところで、Iris を使用して ID サービスを利用する 方法を説明します。
- すでに作成したログインフォームを使用するか、サンプルテンプレートを使用するか、ユーザ名とパスワードフィールドを持つフォームを作成して、ログインサービスに送信します。
- 右画面は一例です。

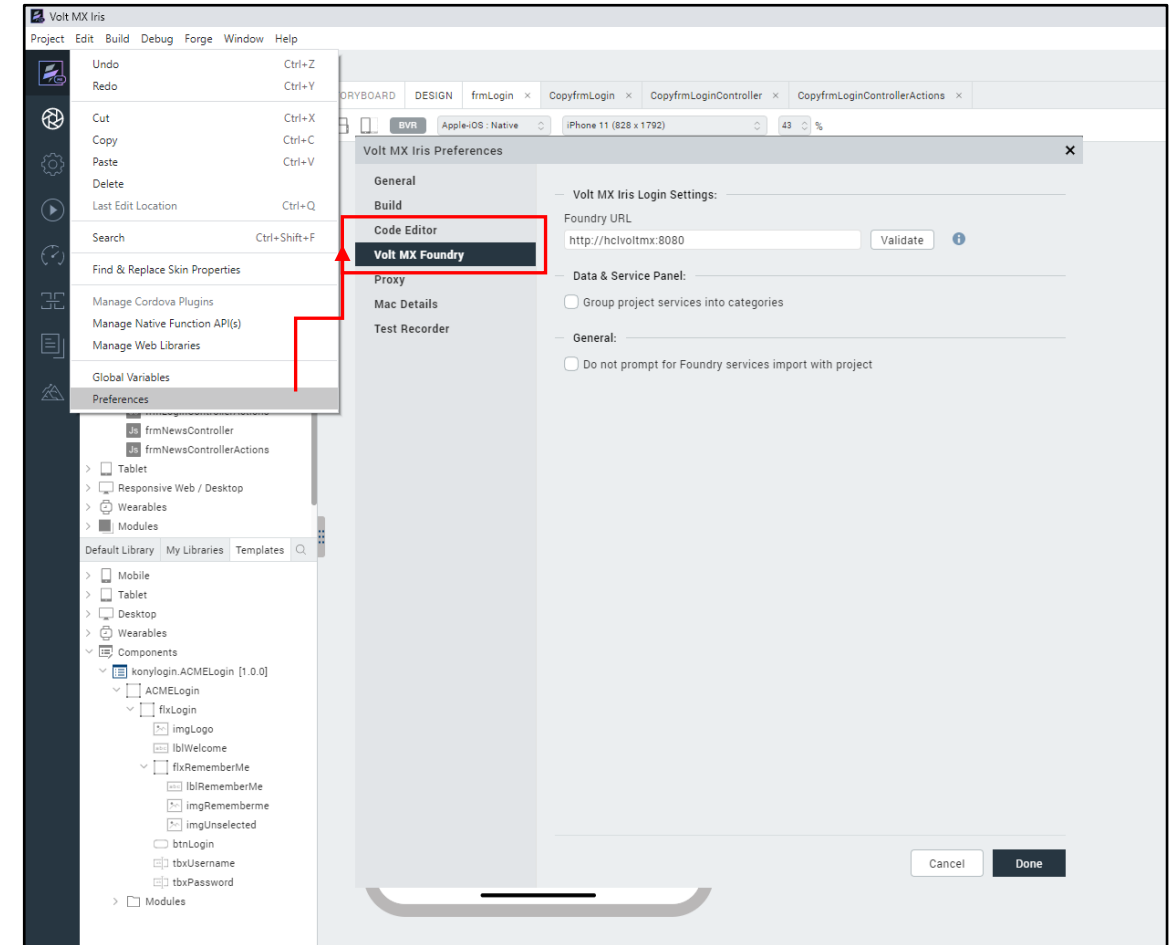


Iris を使用した Identity Service のコール（続き）

- Identity Service を呼び出すには、次の 3 つのステップを実行する必要があります。
 - Foundryの初期化
 - authClient”オブジェクトの取得
 - ログインAPIを呼び出す
- Foundryの初期化
 - 初期化メソッドはFoundryの構成を取得し、キャッシュに保存し、アプリケーションはキャッシュされた構成を使用します。これは同期呼び出しです。
 - そのためには、以下の手順でFoundryアプリケーションをIrisとリンクさせる必要があります。
 - そのためには、次のスクリーンショットに示すように、Irisの環境設定でFoundryのURLを変更する必要があります。デフォルトでは、<https://manage.demo-hclvoltmx.com> に接続されています。
 - ローカルまたはクラウドの Foundry インスタンスで、サービスを構成した Foundry インスタンスを指定する必要があります。

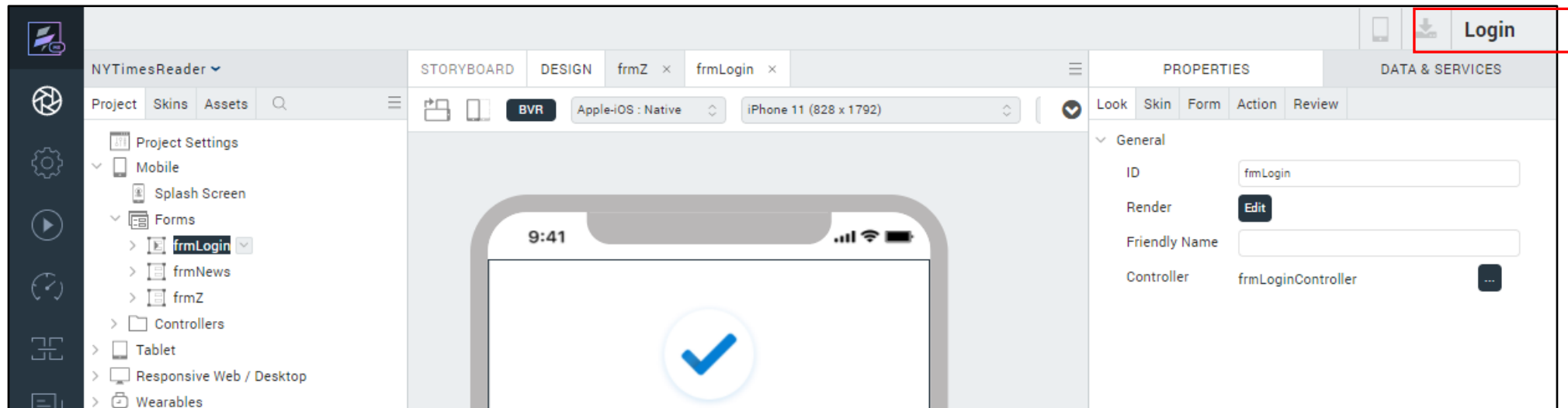
Iris を使用した Identity Service のコール (続き)

- Iris の環境設定にアクセスするには、“Iris ? Preferences (Mac) または Window → Preferences (Windows)”
- Foundry インスタンスの URL を入力します。
 - クラウド Foundry を使用している場合は、<https://manage.demo-hclvoltmx.com> を使用します。
 - ローカル Foundry を使用している場合は、<http://<<IPアドレス:ポート番号>>/> を使用します。
 - Validate をクリックし、バリデーションが成功したことを確認します。



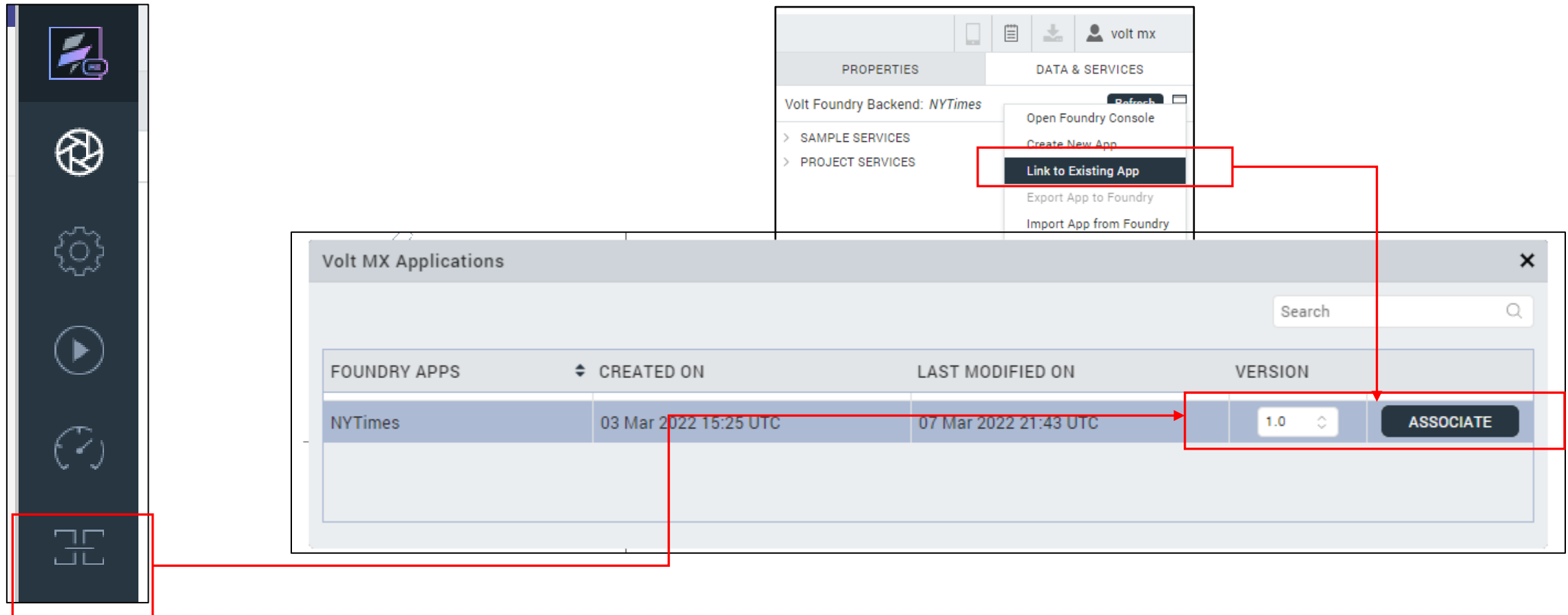
Iris を使用した Identity Service のコール (続き)

- validate successful というメッセージが表示されたら、Foundry 環境への接続は完了です。
- 以下の画面のように、Login ボタンをクリックすると、Foundry のログイン画面が表示されます。



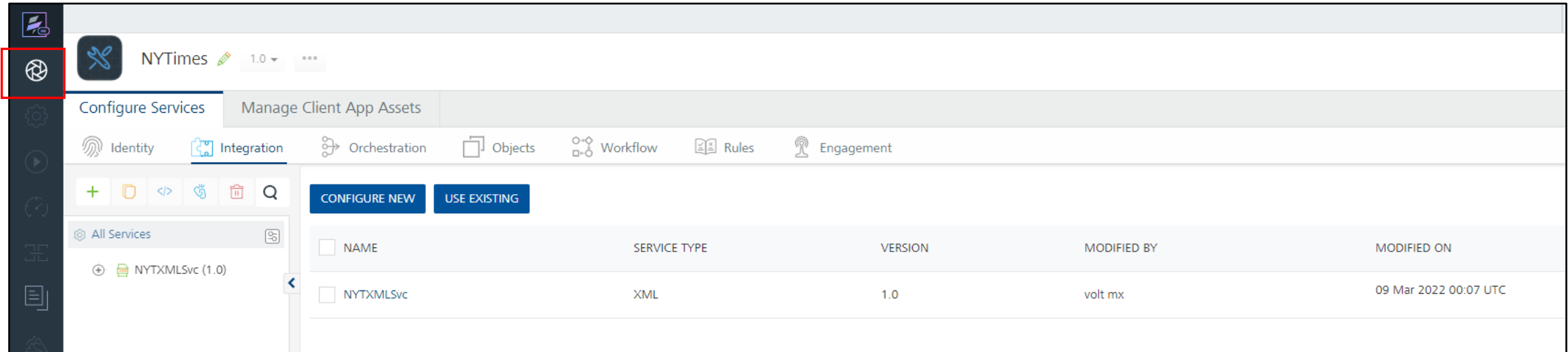
Iris を使用した Identity Service のコール (続き)

- ログインが成功したら、Foundry アプリと Iris アプリの関連付けを行えます。



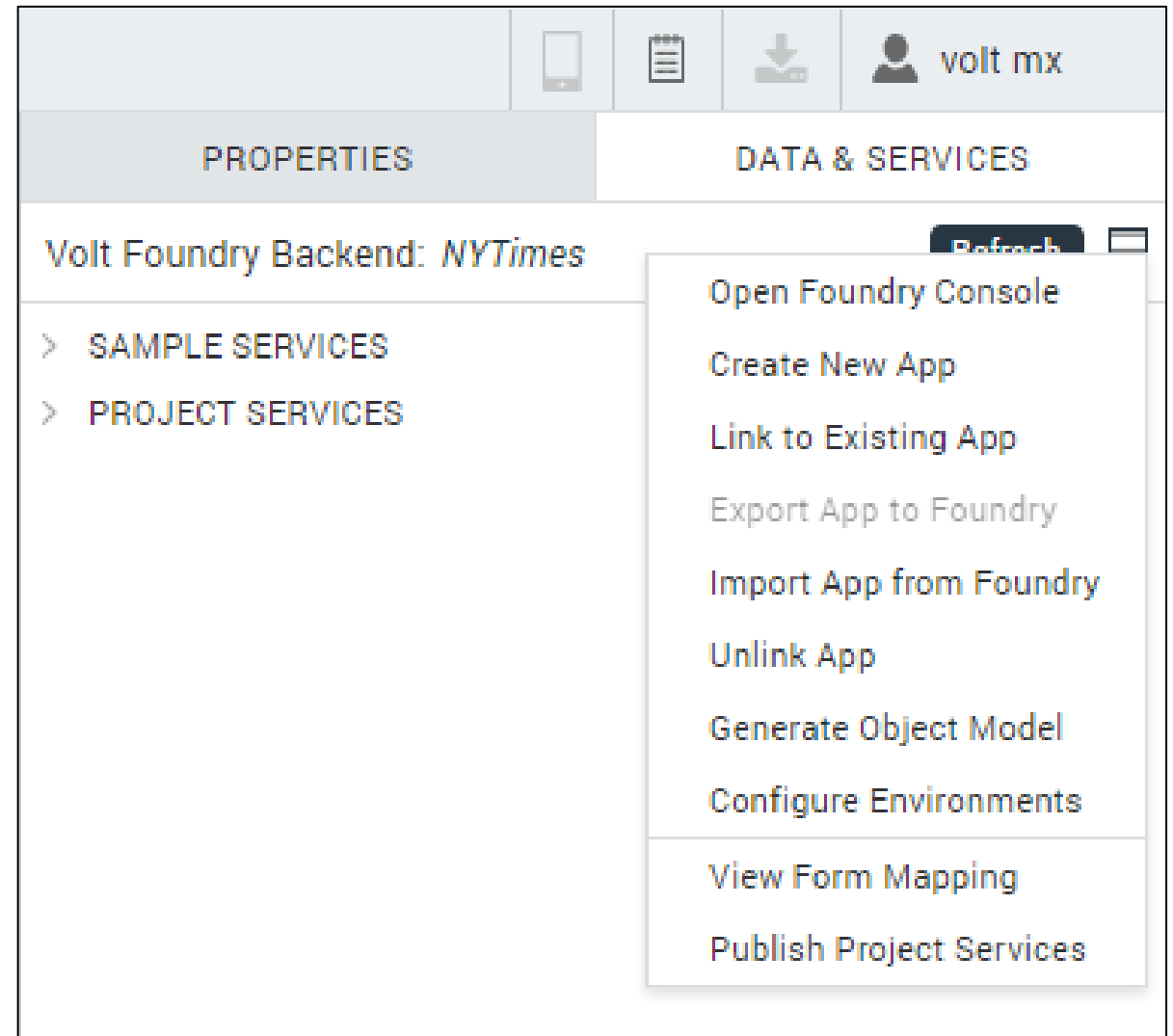
Iris を使用した Identity Service のコール（続き）

- 関連付けに成功すると、Visualized コンソール内に Foundry アプリが表示されます。Iris に戻りたい場合は、以下のように Iris アイコンをクリックします。



Iris を使用した Identity Service のコール (続き)

- DATA & SERVICES ペインで、関連するアプリ名やその他のオプションを確認できます。
- 更新ボタンの横にあるハンバーガーメニューを選択します。以下のことが可能です。
 - Foundry Consoleを開く
 - 新規アプリの作成
 - アプリのインポート
 - その他
- これで、FoundryアプリとIrisの関連付けができましたので、初期化処理を開始できます。



Iris を使用した Identity Service のコール (続き)

- Identity Service API を呼び出す前に、現在の SDK へのインスタンスが必要です。

```
var client = voltmx.sdk.getCurrentInstance();
```

- その後、Identity Service オブジェクトのインスタンスを取得します。

```
var serviceName = "TrainingIdentitySvc";
```

```
var identitySvc = client.getIdentityService(serviceName);
```

Identity Service のコール - ログイン

- ログインメソッド

```
identitySvc.login (options, success_callback, error_callback)
```

- 例

```
var options = {};  
options.userid = this.view.txtUserName.text;  
options.password = this.view.txtPassword.text;  
identitySvc.login(options,  
    function(response) {  
        alert("Login Success:" + JSON.stringify(response));  
    },  
    function(error) {  
        alert("Login Failure:" + JSON.stringify(error));  
    }  
)
```

Identity Service のコール - ログアウト

- ログアウトメソッド

```
identitySvc.logout (options, success_callback, error_callback)
```

- 例

```
var options = {};  
identitySvc.logout(options,  
    function(response) {  
        voltmx.print("#### Logout success:" + JSON.stringify(response));},  
    function(error) {  
        voltmx.print("#### Logout failure:" + JSON.stringify(error));},  
    options  
);
```

- 現在ログインしていない場合は、失敗を返します。

Identity Service のコール - GetProfile

- GetProfile メソッド

```
identitySvc.getProfile (forceFromServer, success_callback, error_callback)
```

- 例

```
var forceFromServer = false;
identitySvc.getProfile(forceFromServer,
  function(response) {
    voltmx.print("User profile is:" + JSON.stringify(response));},
  function(error) {
    voltmx.print("Failed to fetch profile:" + JSON.stringify(error));}
);
```

- 現在ログインしていない場合は、失敗を返します。

Iris を使用した Identity Service のコール

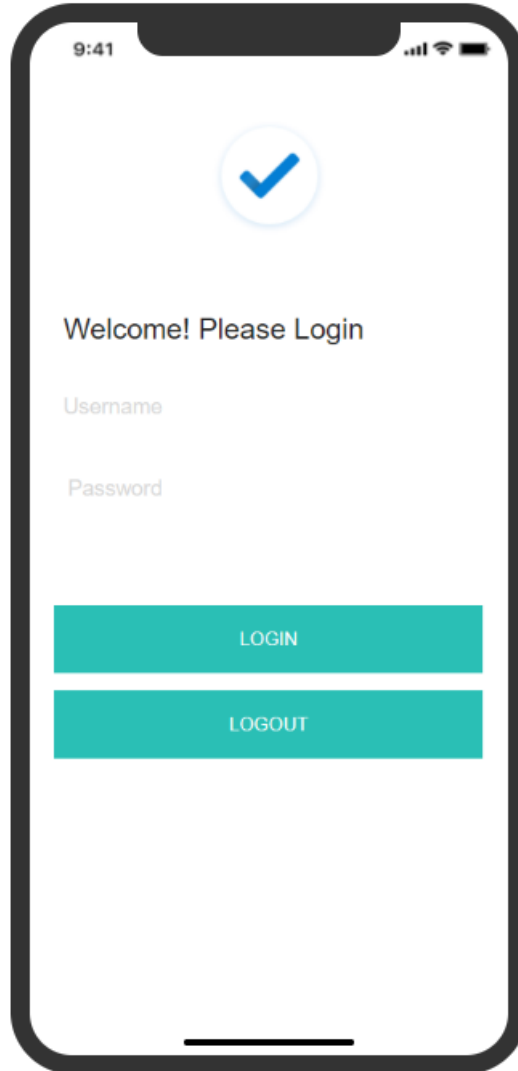
- ログインボタンの onClick イベントを使用して、入力されたユーザーとパスワードをマップし、**login** メソッドを呼び出します。
 - 成功または失敗に基づいて、出力メッセージを表示またはログに記録します。
 - 成功経路と失敗経路の両方を試す
- **logout** メソッド用のボタンを追加する
 - 成功または失敗に基づいて、出力メッセージを表示または記録します。
 - ログインしているときとログアウトしているときの両方で試してください。
- **getProfile** メソッド用のボタンを追加
 - 成功または失敗に基づき、出力メッセージを表示または記録します。
 - ログインしているときとログアウトしているときの両方で試してみてください。

Iris を使用した Identity Service のコール（続き）

- すべてのコードをコントローラに置くことができます
- スニペットで実行すれば、簡単にテストできます
- 完全な例が必要な場合は、サンプルをダウンロードしてください。
- 前の画面のサンプルコードを使用して、login メソッドを呼び出し、成功したら getProfile メソッドを呼び出すコードを作成します。

Login を使用した Identity サービスのコール

- これで、ビルド、実行、テストができるようになりました。



Login success: {}

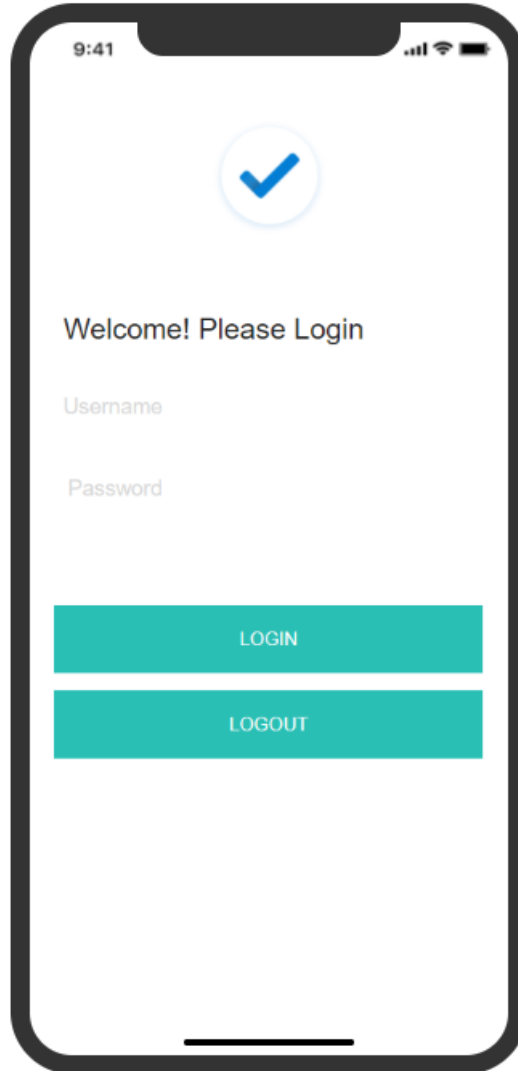
OK

Login failure: {"details":{"message":"Invalid User Credentials","errcode":-4,"errmsg":"Invalid User Credentials"},"mfcod":"Auth-4","opstatus":101,"message":"Invalid User Credentials"}

OK

Logout を使用した Identity サービスのコール

- ログアウトもテスト



Logout success: {}

OK

Logout failure: {"opstatus":107,"message":"Identity Provider's sessions is not active. Please login","details":{"mfcode":""}}

OK

HCL

www.hcltech.com

\$10 BILLION | 159,000+ IDEAPRENEURS | 50 COUNTRIES