

ArcGIS API for JavaScript を使用して Web アプリを開発してみよう！

本日のタイムスケジュール ~2日目~

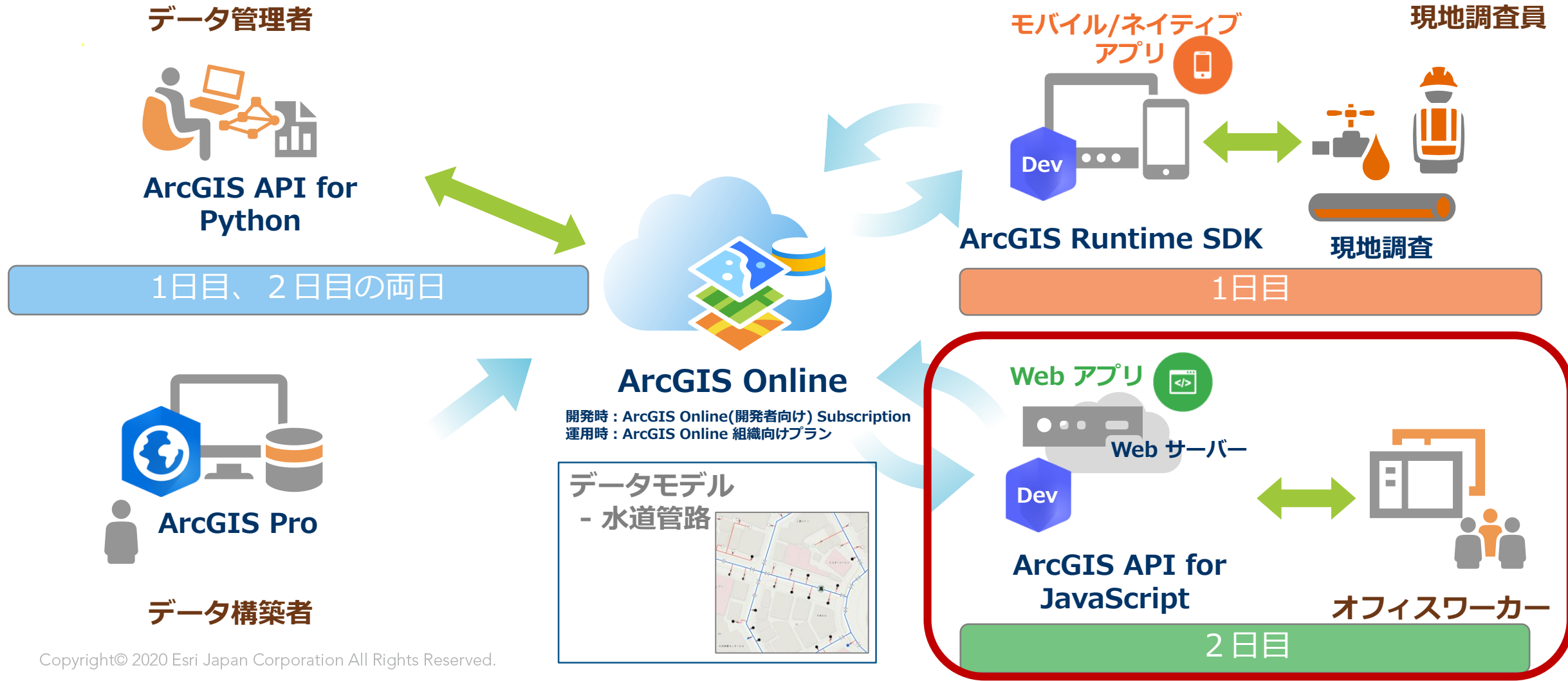
時間	概要	詳細
13:00~13:05	シナリオ説明	本セッション
13:05~13:25	ArcGIS プラットフォームを活用したデータ構築	Web GIS 基礎 ~ArcGIS Online を使ってみよう！~
13:25~13:45		ArcGIS Pro を使用した Web マップの作成
13:45~13:50		休憩
13:50~14:10		ArcGIS API for Python を使用した Web マップ更新
14:10~14:15		休憩
14:15~15:55	ArcGIS プラットフォームを活用した Web アプリ開発ハンズオン	ArcGIS API for JavaScript を使用して Web アプリを開発してみよう！
15:55~16:00		まとめ

内容

1. 業務の定義とセクションの位置づけ
2. ArcGIS API for JavaScript の紹介
3. ハンズオンの概要
4. ハンズオン開始
5. 最新情報

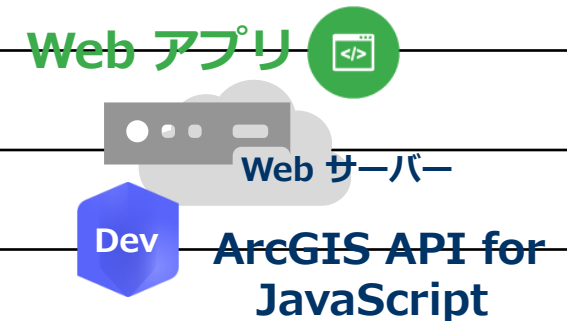
業務の定義とセクションの位置づけ

ArcGIS プラットフォームを活用した 水道業務システムのアプリケーション構成



水道業務システムの機能要件

No	ユーザー種別	機能要件
1	データ構築者	水道データの作成・編集・更新ができること
2	データ管理者	水道データを定期更新で反映できること
		調査対象エリアを作成できること（オフラインでも使用可）
3	現地調査員	モバイル端末で漏水調査ができること
		現地調査した写真が添付できること
		ネットワークが繋がらない環境で調査ができること
4	オフィスワーカー	Web ブラウザから参照できること
		印刷ができること
		水道データの検索ができること
		基本の地図機能と概観図が使用できること



ArcGIS API for JavaScript

Web アプリ

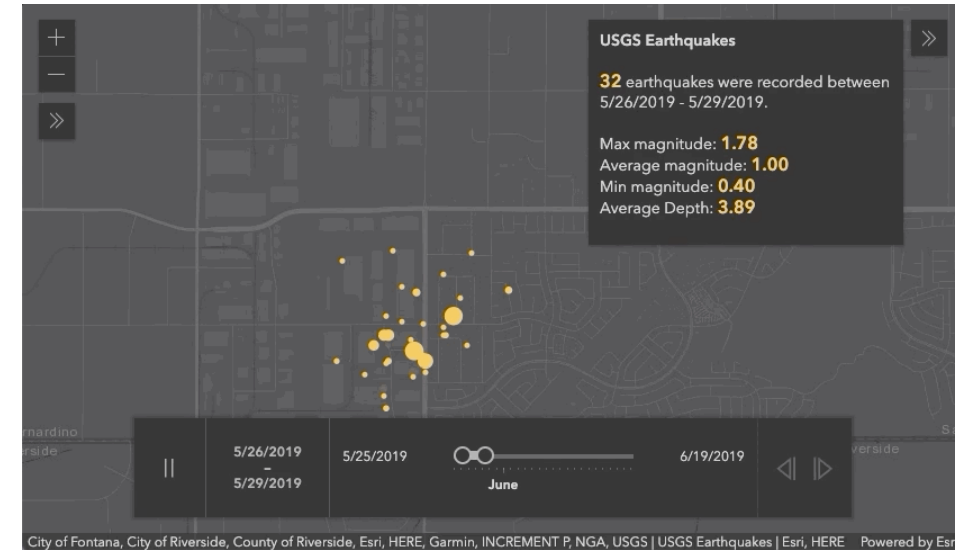
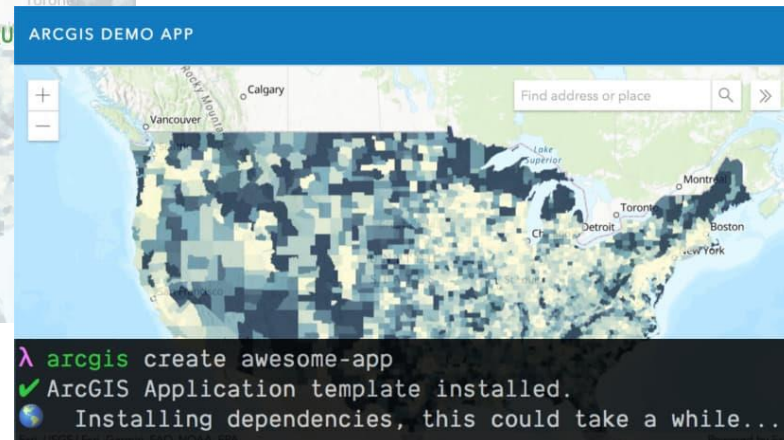
ArcGIS API for JavaScript

Web ブラウザで動作する GIS アプリケーション開発 API

- 2D/3D をサポートする強力な API
- WebGL を活用した高パフォーマンス描画
- 豊富なビジュアライゼーション
- 多様なデータソースをサポート
- 主要ブラウザをサポート

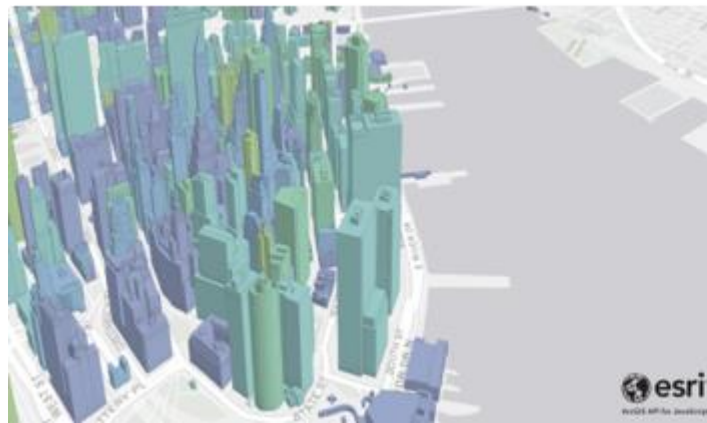
```
export const featureLayer = new FeatureLayer({
  portalItem: {
    id: 'b234a118ab6b4c91908a1cf677941702'
  },
  outFields: ['NAME', 'STATE_NAME', 'VACANT', 'HSE'],
  title: 'U.S. counties',
  opacity: 0.8
});

export const map = new ArcGISMap({
  basemap: {
    baseLayers: [
      new TileLayer({
```



マッピング

豊富なデータソースと多彩な地図表現を実現



データソース

- ArcGIS Online
- ArcGIS Enterprise
- 外部サービス
- ...

シンボル表現

- 個別値分類
- 数値分類
- ヒートマップ
- 時系列
- ...

地図操作

- ポップアップ
- ブックマーク
- 地図の回転
- ...

編集

どこでも GIS データを簡単に編集・共有



データ編集

- 図形編集
- 属性編集
 - 数値/テキスト/日時
 - リレーションシップ
 - ファイル添付
- 編集履歴の記録

※ バージョン 4.x での編集機能のサポートは限定的です。
将来のリリースでバージョン 3.x と同等の機能が実装される予定です。

検索・解析

Web サービスと連携し高度な GIS 処理を簡単に実行



検索

- ・ 属性検索
- ・ 空間検索
- ・ リレーションシップ
- ・ ...

解析・分析

- ・ ジオプロセッシング
- ・ ジオメトリ演算
- ・ ネットワーク
 - ・ ルート検索
 - ・ 到達圏検索
 - ・ 最寄施設検索
- ・ ジオコーディング
- ・ ...

ハンズオンの概要

ハンズオンの概要

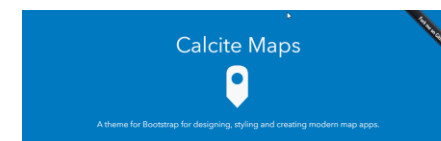
● 概要

- オフィスワーカーが利用するための要件を満たす、Web アプリケーションを開発
 - Esri が提供している Calcite Map を使用して、API が標準で用意しているウィジェットと独自のウィジェットを使用した開発を行う。

No	ユーザー種別	機能要件
4	オフィスワーカー	Web ブラウザから参照できること
		印刷ができること
		水道データの検索ができること
		基本の地図機能と概観図が使用できること

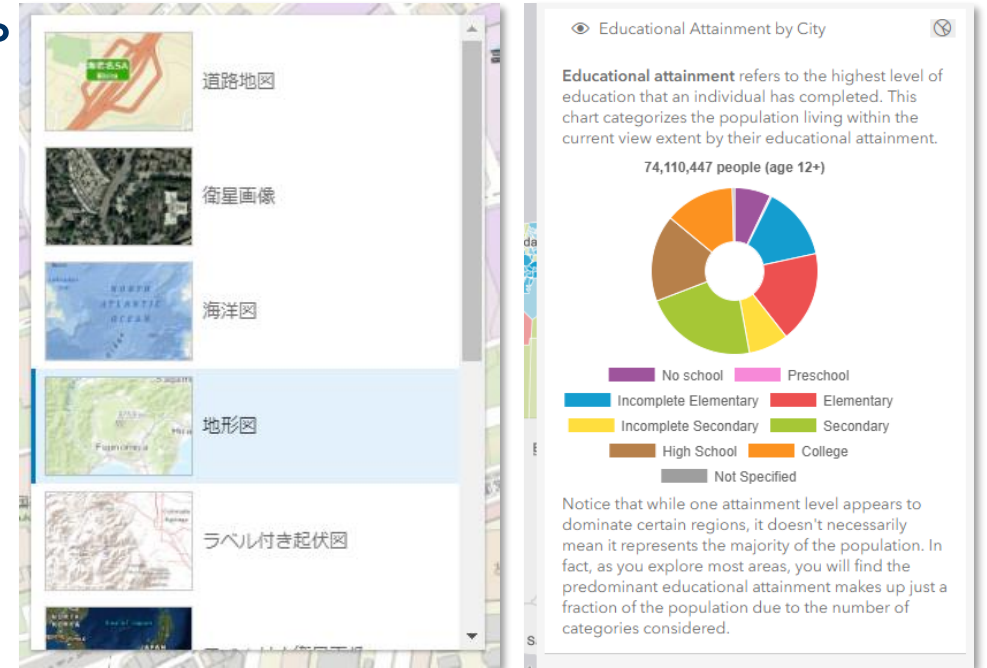
● Calcite Mapとは？

- Esri がオープンソースとして提供しており、Bootstrap を使用しているため、レスポンシブでモダンなマップアプリを開発することができる。
 - <http://esri.github.io/calcite-maps/samples/index.html>



ArcGIS API for JavaScript が提供するウィジェット

- ウィジェットとは、ArcGIS API for JavaScript が提供している GIS 機能を兼ね備えた UI コンポーネント
- ウィジェットを活用することで GIS の機能と同時にリッチな UI を Web マッピングアプリケーションに簡単に組み込むことができる。
- 独自ウィジェットの作成も可能



- GeoNet ブログ「続編-はじめてのweb-マッピングアプリケーション開発-ウィジェット編」でも公開中。

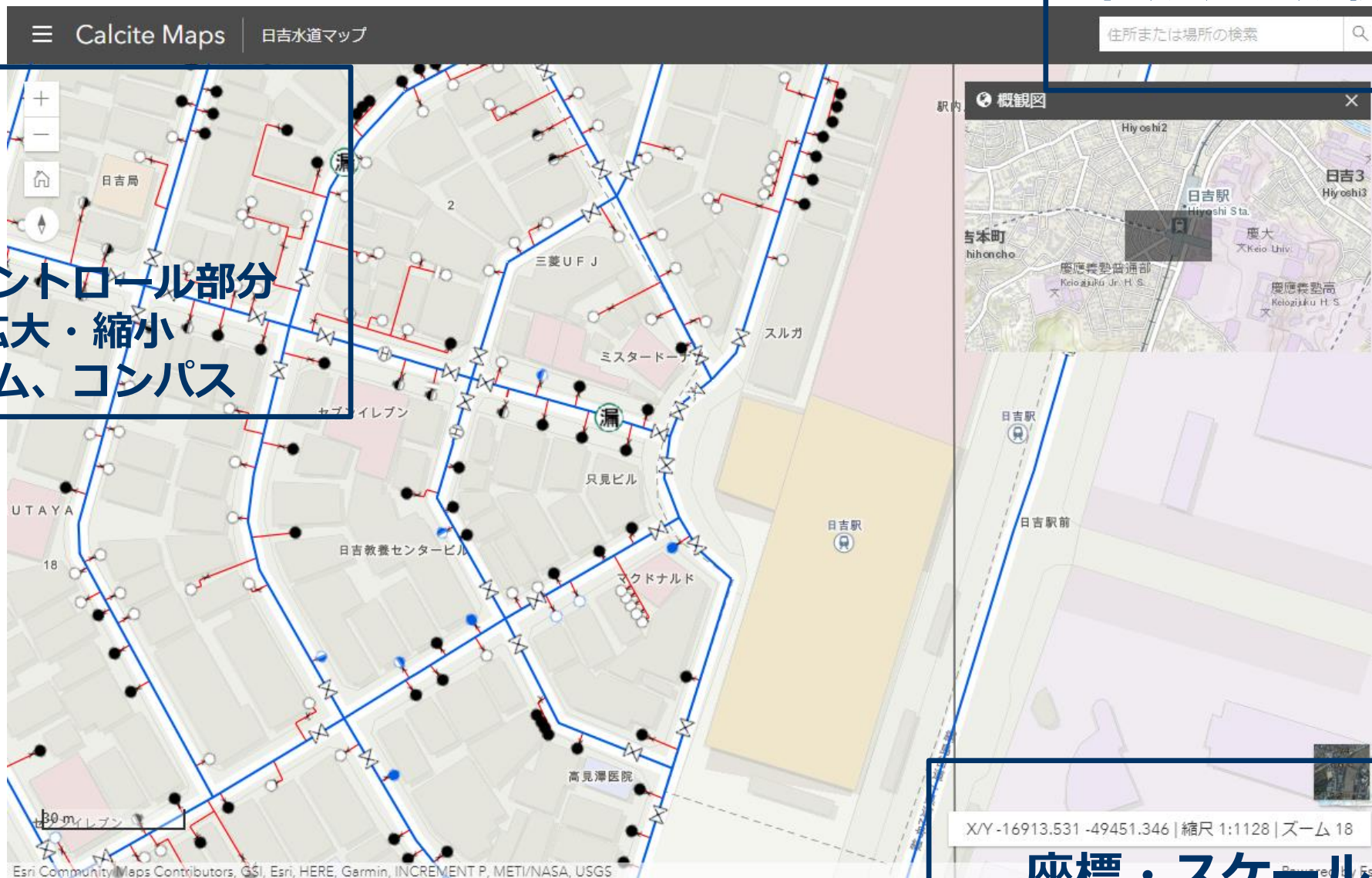
➤ <https://community.esri.com/docs/DOC-14692>

Web アプリケーションの機能

住所、場所検索

住所または場所の検索

地図コントロール部分 拡大・縮小 ホーム、コンパス

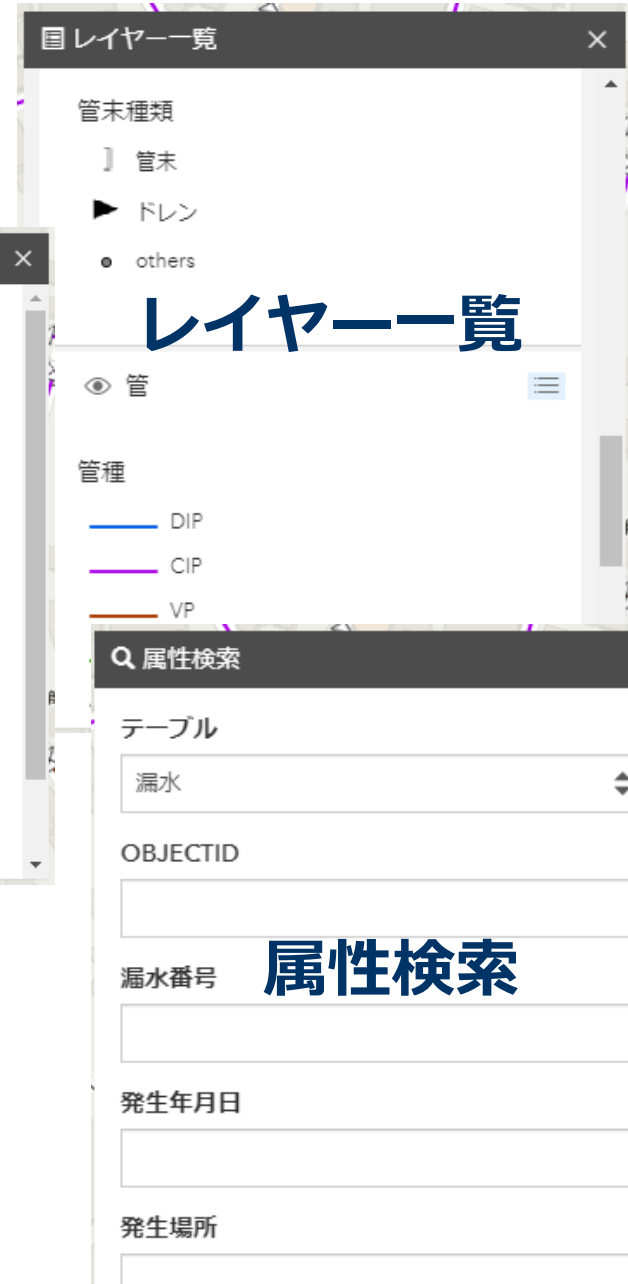


座標・スケール表示

Web アプリケーションの機能

● ウィジェット一覧

- 概観図
- レイヤー一覧
- ブックマーク
- 印刷
- 属性検索



Web アプリケーションのデモ

Web アプリケーションの構成

	ディレクトリ名/ファイル名		内容
1	css		bootstrap を定義した CSS ファイル、本アプリで使用するために定義した CSS ファイル
2	fonts		フォントファイル
3	js		
4		jquery	bootstrap コンポーネント群 (jquery を使用)
5		main.js	地図メイン部 (地図表示、標準ウィジェットの作成)
6		over-view-map.js	概観図の設定と表示
7		query-task.js	属性検索用のウィジェットを作成
8		search-feature-table.js	属性検索の実行と結果をフィーチャテーブル ウィジェットに表示
9		setting-popup-template.js	ポップアップの設定
10	index.html		bootstrap の要素を適用した HTML ファイル

ハンズオン対象

API、CSS の利用 (index.html)

```
<!-- Calcite Maps Bootstrap -->
<link rel="stylesheet" href="/css/calcite-maps-bootstrap.min-v0.10.css">
<!-- Calcite Maps -->
<link rel="stylesheet" href="/css/calcite-maps-arcgis-4.x.min-v0.10.css">

<!-- ArcGIS JS 4.x -->
<link rel="stylesheet" href="https://js.arcgis.com/4.16/esri/css/main.css">
<!-- 日吉水道マップ CSS ファイル -->
<link rel="stylesheet" href="/css/index.css">

<!-- ArcGIS JS 4.x -->
<script src="https://js.arcgis.com/4.16/"></script>

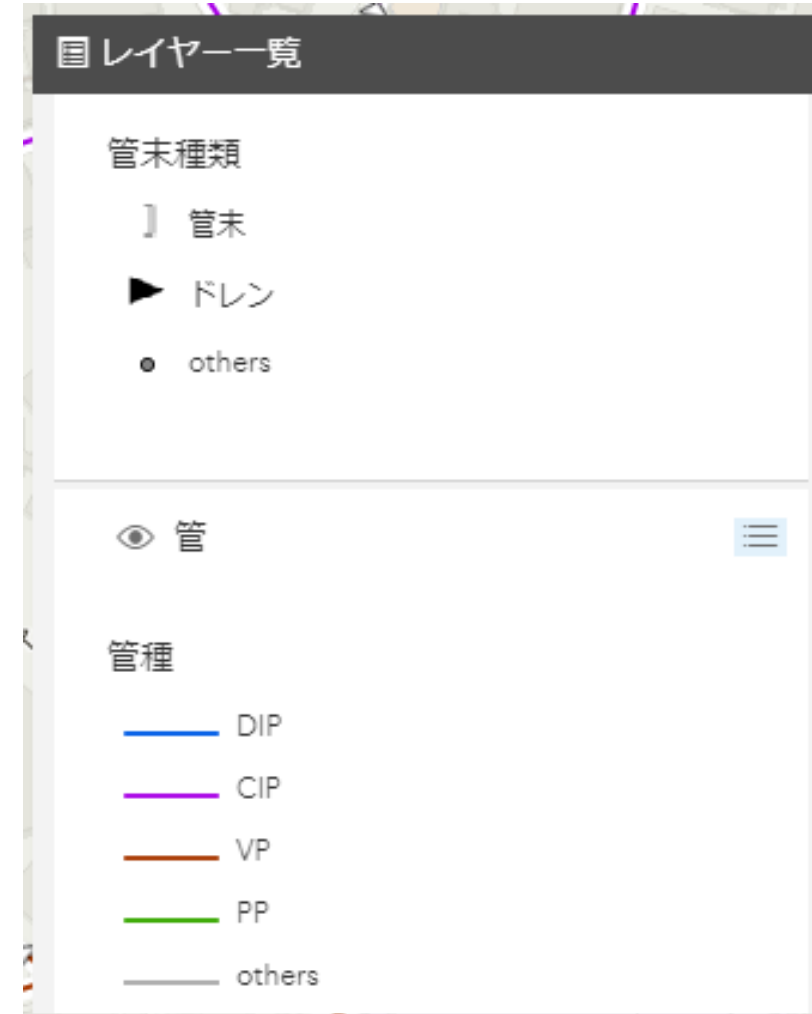
<!-- 日吉水道マップ JS ファイル -->
<script src="/js/main.js"></script>
<!-- 概観図 -->
<script src="/js/over-view-map.js"></script>
<!-- 属性検索 -->
<script src="/js/query-task.js"></script>
<!-- 検索結果の表示（フィーチャ テーブル） -->
<script src="/js/search-feature-table.js"></script>
<!-- ポップアップの設定 -->
<script src="/js/setting-popup-template.js"></script>
```

ウィジェットの作成例

```
<div id="widgetBodyDiv" class="panel-body">  
  <div id="layerListDiv"></div>  
</div>
```

```
// ウィジェットの作成  
const layerList = new LayerList({  
  selectionEnabled: true,  
  container: "layerListDiv",  
  view: mapView,  
  listItemCreatedFunction: function(event) {  
    const item = event.item;  
    if (item.layer.type !== "group") {  
      item.panel = {  
        content: "legend",  
        open: true  
      };  
    }  
  }  
});
```

- HTML ファイル
 - DIV の要素を指定 (layerListDiv)
- JS ファイル
 - LayerList クラスの container に DIV の要素を指定 (layerListDiv)



ハンズオンの流れ

ハンズオンの流れ

- Step1: ハンズオンの準備
 - 事前に準備した開発環境へのソースの配置
- Step2: 地図メイン部の実装
 - Web マップを使用 (Map と MapView)
- Step3: 標準ウィジェットの实装
 - 標準ウィジェットの作成
- Step4: 概観図の実装
 - 概観図を表示するウィジェットを作成
- Step5: 属性検索の実装
 - 属性検索用のウィジェットを作成
 - 検索結果の表示にはフィーチャ テーブル ウィジェットを使用

① ステップ1、2 : 20分
実装 (18分) + (解説2分)

② ステップ3、4 : 30分
実装(25分) + 解説(5分)

③ ステップ5 : 30分
実装(25分) + 解説(5分)

Step1: ハンズオンの準備

事前に準備した開発環境へのソースの配置

Step2: 地図メイン部の実装

Web マップを使用 (Map と MapView)

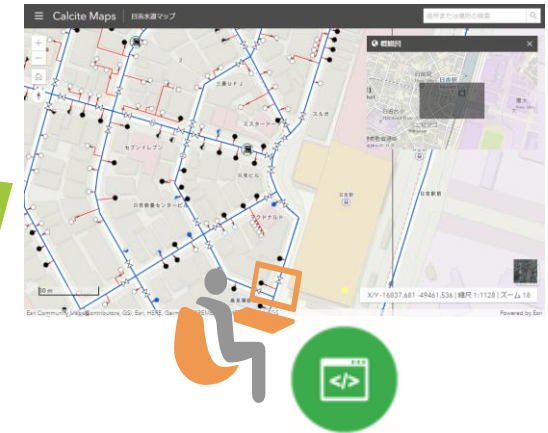
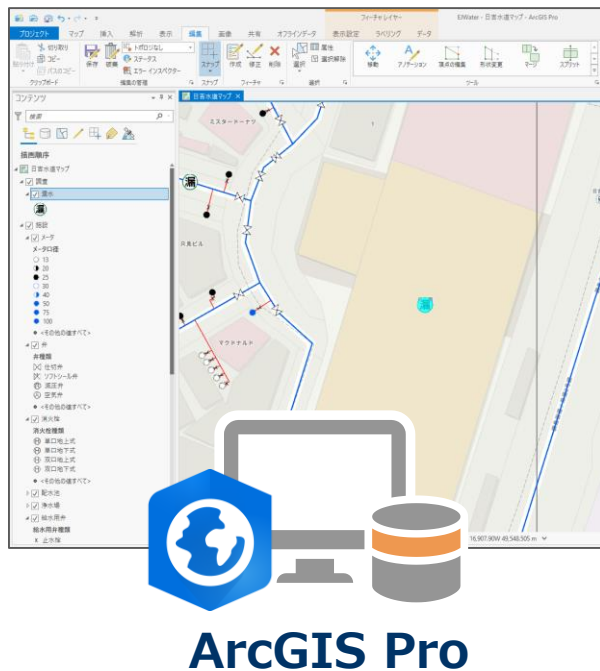
● 演習

➤ <https://bit.ly/34wQ9uw>



Web マップの活用

- ArcGIS Pro からローカルの FGDB を編集し、ArcGIS Online 上のホスト フィーチャ レイヤーをアップデート



ArcGIS API for JavaScript

Web マップの ID を指定

- ・ 参照/分析するデータは最新の状態

スクリプトによるデータ更新の自動化・定期実行でデータ完全性を保つ

Step3: 標準ウィジェットの実装 標準ウィジェットの作成

Step4: 概観図の実装 概観図を表示するウィジェットを作成

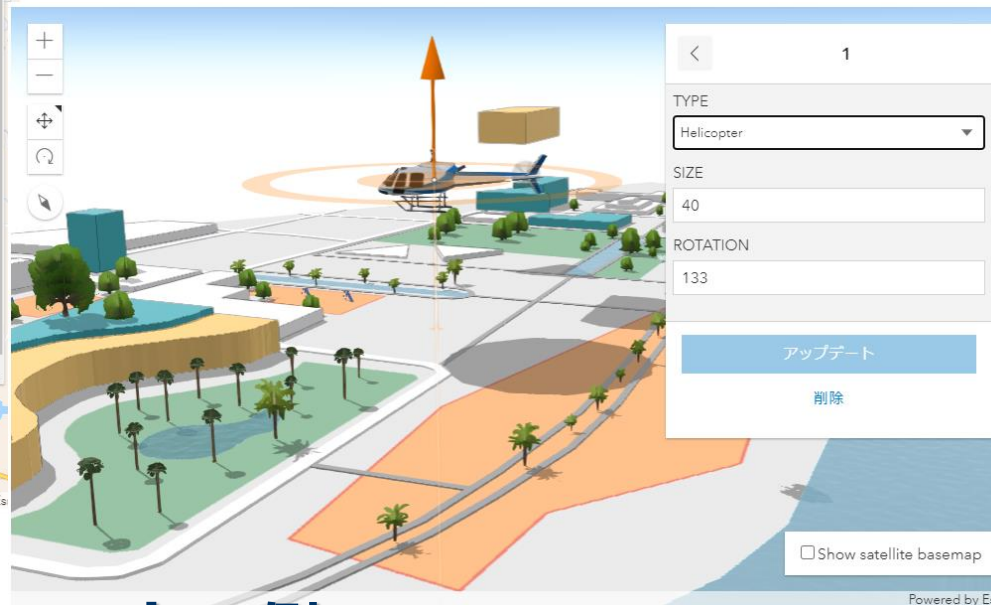
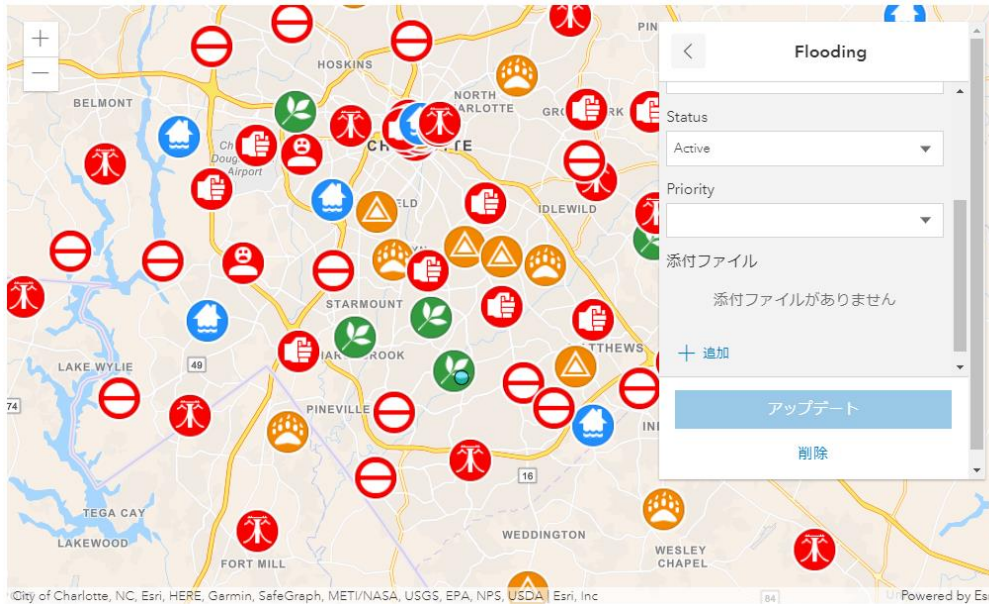
● 演習

➤ <https://bit.ly/2EuRpDH>



ウィジェットの活用

- 標準ウィジェットや独自ウィジェットは、作成して地図に配置したり、ウィジェットを目的に応じて呼び出して使用することができます。
- 編集ウィジェットなど、20を超える標準ウィジェットを提供しています。



● 2D、3Dに対して編集用のウィジェットも提供しています。
● データの編集に加えて、添付ファイルの追加や削除も可能です。

編集ウィジェットの例

Step5: 属性検索の実装

属性検索用のウィジェットを作成

検索結果の表示にはフィーチャ テーブル ウィジェットを使用

● 演習

➤ <https://bit.ly/32FXMMP>



属性検索 (フィーチャの検索)

フィーチャ レイヤー

- **FeatureLayer**
 - **queryFeatures()**
 - フィーチャ レイヤーに対してクエリを実行
 - **createQuery()**
 - パラメーターの作成
 - Query オブジェクトが返る

```
const featureLayer = new FeatureLayer({
  // ...
});

const queryParams = featureLayer.createQuery();
queryParams.where = "Population > 1000000";
queryParams.geometry = geometry;
queryParams.spatialRelationship = "intersects";

featureLayer.queryFeatures(queryParams)
  .then(function(featureSet) {
    console.log(featureSet.features);
  });
```

属性検索（検索結果の表示） フィーチャ テーブルウィジェット

● FeatureTable

● Layer

- ウィジェット内に表示するフィールドと属性を含んだ FeatureLayer を指定

● fieldConfigs

- 個々に設定したフィールドの配列を指定

● editingEnabled

- editingEnabled を true にすることで編集が可能

● container

- ウィジェットを配置する DOM 要素を表す ID またはノード

```
const featureTable = new FeatureTable({  
  view: MapView,  
  featureLayer,  
  fieldConfigs: fieldConfigs,  
  editingEnabled: true,  
  container: "tableDiv"  
});
```

最新情報

最新情報

ArcGIS API for JavaScript 4.16

- 7月9日米国リリース
- 主な新機能等
 - レイヤー ブレンディングの追加
 - 2つ以上のレイヤーを合成することでさまざまな見せ方が可能
 - クラスターのラベル対応
 - OGCFeatureLayer の追加
 - OGC API Feature サービスのデータを可視化
 - 3D でのリアルな水面の可視化
 - WaterSymbol3DLayer を使用することでリアルな外観を実現
 - 統合メッシュ レイヤーの形状を変更
 - 新しいクラス SceneModifications を使用することで対応
 - ビルディング エクスプローラーウィジェットの追加
 - BIM モデルを使用することで建物の特定カテゴリや建設段階などを把握
 - CIMSymbol ラインとポリゴンのサポート
 - CIMSymbol が CIMLineSymbol と CIMPolygonSymbol をサポート
 - フィーチャ テーブルウィジェットによる編集機能のサポート
- 8月19日に国内サポートを開始

sri ジャパン



ロードマップ

