



# GIS Data in the iTwin.js ecosystem

Best Practices Guide

**Bentley**<sup>®</sup>  
*Advancing* Infrastructure

[www.bentley.com](http://www.bentley.com)

Contents

- GIS Data in iTwin.js ecosystem.....3
- Two solutions - Referencing or Synchronizing.....3
- Level of Interactions*.....3
  - Level 1 - Visual context + Properties Access*.....3
  - Level 2 – Visual context + Properties access + Interaction with the elements in the iModel*.....3
- GIS Data formats .....4
- Referencing GIS data to an iModel.....8
  - Strengths and limitations.....8
  - Supported Services*.....9
    - ArcGIS REST Map Service*.....9
    - Open Geospatial Consortium WMS and WMTS Services can also be attached to the iModel, through the Map Layers widget.....10
  - Adding ArcGIS/WMS/WMTS layers in the Map Layers widget.....10
- Synchronizing GIS data to an iModel.....11
  - Supported Formats .....11
  - Strengths and Limitations.....11

## GIS Data in iTwin.js ecosystem

Do you see value in integrating your GIS data into the iTwin.js ecosystem, but not know where to start? This user guide will help you to choose the best solution for your needs. It contains general information on what can be achieved with GIS data in the iTwin.js ecosystem.

## Two solutions - Referencing or Synchronizing

There are two ways to bring GIS data into the iTwin.js ecosystem. GIS data can be referenced to an iModel or synchronized (copied) to an iModel. Both solutions meet different needs and support different formats. The choice of one or the other depends on what you want to achieve with your GIS data and the type of data that is available for your project.

### *Level of Interactions*

The level of interactions required in your project will have an impact on the choice you will make. There are two levels of interactions that can be reached with GIS data in the iTwin.js ecosystem:

Level 1 - Visual context + Properties Access (when selecting an element or location)

Level 2 – Visual context + Properties Access + Interaction with the properties (queries & searches, Design Validation, etc.)

#### *Level 1 - Visual context + Properties Access*

In this case, the user wants a visual context as well as accessing the GIS properties. For example, you want to see the land use of the region where your project is planned, and you also want to access the properties of that land use by clicking an area of interest in the view. Depending on the type of GIS data that is available to you, you have two choices. You can choose to reference your GIS data through the Map Layers widget or, you can choose to synchronize your GIS data to an iModel with the GeoConnector, a product specifically designed to integrate and share geospatial data from a variety of sources.

*Reference through the Map Layers* – If you choose this option, you will be able to reference the ArcGIS Map Service, the WMS and WMTS. However, the properties will be available only for the ArcGIS Map Service. Access to the properties for WMS and WMTS may be added at a later date.

To access the properties from an ArcGIS Map Service, the server must be configured to return information with the GetFeatureInfo feature request. Contact your GIS administrator for more details. A complete description of the referencing option is available further in this document.

*Synchronize with the GeoConnector* – If you choose this option, the GIS data will be copied to your iModel by using the GeoConnector. You will have access to the properties by selecting geometries in the viewer, as with any other elements contained in the iModel. This solution does not support the same GIS formats as the referencing option. A table listing the different formats for both solutions is available in the next section. We will see the synchronizing option in more detail subsequently.

#### *Level 2 – Visual context + Properties access + Interaction with the elements in the iModel*

In this case, your requirements are focused on more advanced needs, such as Design Validation or searches and queries on GIS data. To satisfy those requirements, the option of synchronizing GIS data to iModel is the way to accomplish this. Using this method, the GIS data is part of the iModel like any other

Engineering data. This option provides the means to perform different interactions with the data that is inside the iModel, such as queries and searches, Design Validation, etc. We will see the Synchronization tool in more details further in this guide.

## GIS Data formats

Each level of solution supports different formats. One of the first things to check is the format of the data source you want to interact with. In some cases, this will dictate your choice of referencing or synchronizing GIS data to the iModel. The grid below lists all supported formats combined with GIS environment criteria. Consulting this grid will help you to find the proper solution. For example, find the data type, the data location, the authentication method if any, etc. You will find the recommended solution for this specific case.

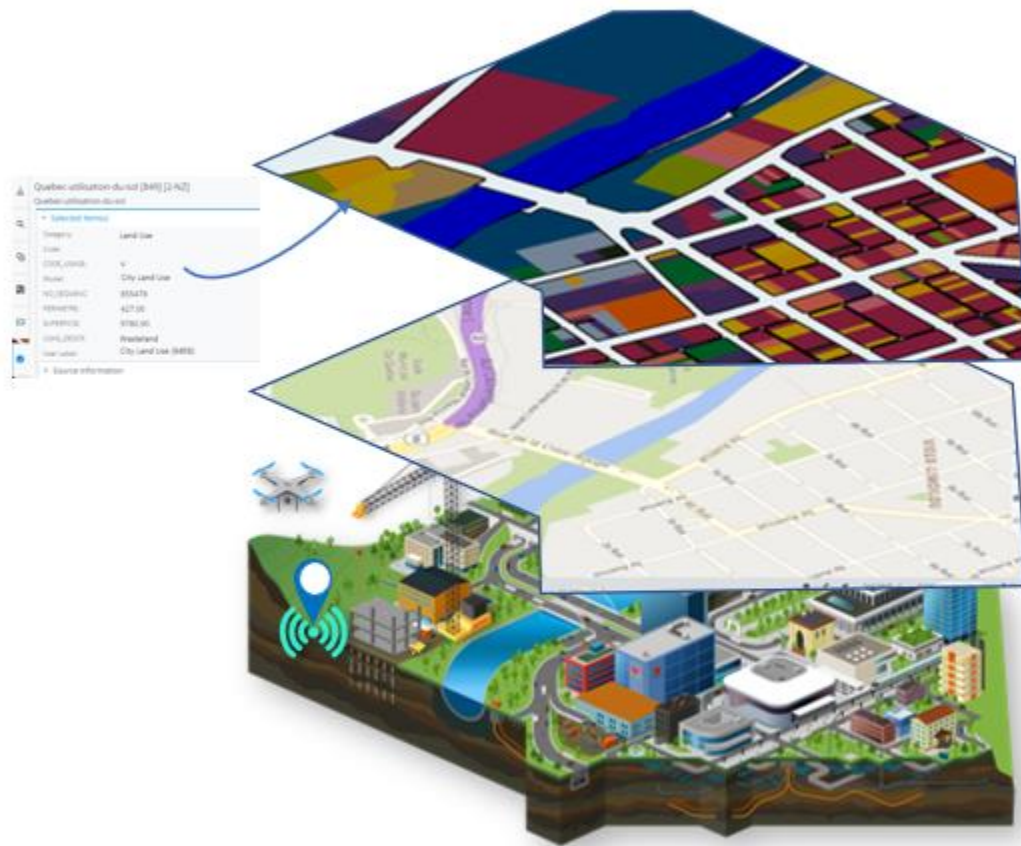
It is important to know that GIS data may be available in multiple formats. It is the case with ArcGIS Server. For example, a server can be configured to publish in Feature Service or Map Service modes, simultaneously. Contact your GIS Administrator for more information on your server's configuration.

Supported Data Types	Authentication	Data Location	Proposed solution	Properties	Symbology	Comments
Shapefile	N/A	ProjectWise Design Integration or ProjectWise Share	Synchronizing iTwin Synchronizer Portal (cloud)	Yes	Basic symbology defined in the configuration file: color, fill color, transparency, and line weight  For more details on the configuration file, consult the <a href="#">GeoConnector wiki page</a>	Use Shapefile or File Geodatabase when no live access to the ArcGIS server is possible
		Local folder	Synchronizing iTwin Synchronizer Client (desktop)			
ESRI File Geodatabase	N/A	ProjectWise Design Integration	Synchronizing iTwin Synchronizer Client (desktop) – The files must be downloaded with ProjectWise Explorer and copied locally.	Yes	Basic symbology defined in the configuration file: color, fill color, transparency, and line weight  For more details on the configuration file, consult the <a href="#">GeoConnector wiki page</a>	Not possible in iTwin Synchronizer portal as the UI prevents folder selection.  Use Shapefile or File Geodatabase when no live access to the ArcGIS server is possible
		Local folder	Synchronizing iTwin Synchronizer Client (desktop)			
ArcGIS Feature Service  (.geodb file with URL to the FeatureServer)	Public	.geodb in a local folder	Synchronizing iTwin Synchronizer Client (desktop)	Yes	Basic symbology defined in the configuration file: color, fill color, transparency, and line weight  For more details on the configuration file, consult the <a href="#">GeoConnector wiki page</a>	
		.geodb in ProjectWise Share or ProjectWise Design Integration	Synchronizing iTwin Synchronizer Portal (cloud)			
	Token authentication (Username & Password)	.geodb in a local folder	Synchronizing iTwin Synchronizer Client (desktop)			

Supported Data Types	Authentication	Data Location	Proposed solution	Properties	Symbology	Comments
ArcGIS Map Service	Token authentication (Username & Password)	ArcGIS server	Referencing Map Layers widget	No (Not yet implemented)	Inherited from the existing symbology defined in the GIS system	The ArcGIS Identity (OAuth2) protocol authentication has been implemented in the Map Layers widget API. This functionality is not yet available in all Bentley Viewer Applications. The RealityData Web Viewer has this functionality. Please contact your Bentley representative for more information.
	Federated account (OAuth2)  (Support has been added in the Map Layers widget API, see in the comments)					
PostGIS database (.geodb file with URL to the database)	Basic credentials Username & Password	.geodb in a local folder	Synchronizing iTwin Synchronizer Client (desktop)	Yes	Basic symbology defined in the configuration file: color, fill color, transparency, and line weight  For more details on the configuration file, consult the <a href="#">GeoConnector wiki page</a>	Not possible in iTwin Synchronizer portal as credentials cannot be stored for security reason
SQL Spatial database (.geodb file with URL to the database)	Username & Password	.geodb in a local folder	Synchronizing iTwin Synchronizer Client (desktop)	Yes	Basic symbology defined in the configuration file: color, fill color, transparency, and line weight  For more details on the configuration file, consult the <a href="#">GeoConnector wiki page</a>	Not possible in iTwin Synchronizer portal as credentials cannot be stored for security reason
Oracle Spatial database	Username & Password	.geodb in a local folder	Synchronizing iTwin Synchronizer Client (desktop)	Yes	Basic symbology defined in the configuration file: color, fill color, transparency, and line weight. For more details on the configuration	Not possible in iTwin Synchronizer portal as credentials cannot be stored for security reason

Supported Data Types	Authentication	Data Location	Proposed solution	Properties	Symbology	Comments
(.geoddb file with URL to the database)					file, consult the <a href="#">GeoConnector wiki page</a>	
GeoJson file	N/A	ProjectWise Design Integration or ProjectWise Share	Synchronizing iTwin Synchronizer Portal (cloud)	Yes	Basic symbology defined in the configuration file: color, fill color, transparency, and line weight  For more details on the configuration file, consult the <a href="#">GeoConnector wiki page</a>	
		Local folder	Synchronizing iTwin Synchronizer Client (desktop)			
KML file	N/A	ProjectWise Design Integration or ProjectWise Share	Synchronizing iTwin Synchronizer Portal (cloud)	yes	Basic symbology defined in the configuration file: color, fill color, transparency, and line weight  For more details on the configuration file, consult the <a href="#">GeoConnector wiki page</a>	
		Local folder	Synchronizing iTwin Synchronizer Client (desktop)			
WMS (Web Map Service)	Public Basic Authentication	Map Service	Referencing Map Layers widget	No	Inherited from the existing symbology defined in the GIS system	
WMTS (Web Map Tile Service)	Public Basic Authentication	Map Service	Referencing Map Layers widget	No	Inherited from the existing symbology defined in the GIS system	

## Referencing GIS data to an iModel



GIS data is often available as digital maps that have been created in a GIS system. A map contains all the layers required for a specific project. Those maps can be referenced to an iModel and viewed in an iTwin viewer for adding contextual information. Existing maps are referenced via the Map Layers widget, three formats are supported: WMS, WMTS and ArcGIS Map Service. Referenced maps are published as raster data and it is not presently possible to retrieve information from such layers. Development is underway to permit properties retrieval from such layers.

### Strengths and limitations

A key benefit of referencing an existing map layer is the ability to retain and utilize the existing layer symbology, as defined in the original GIS system.

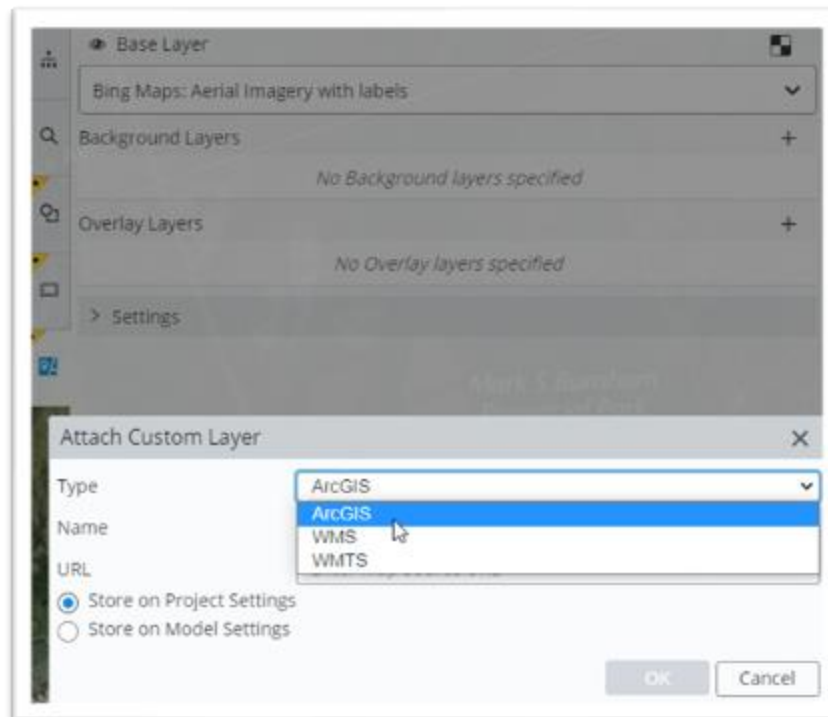
Another major advantage is the access to live data –Referencing GIS data in iTwin viewer ensures that it is always valid and up to date.



## Supported Services

The iTwin.js application allows users to attach diverse types of layers through the “Map Layers” widget. The following types are supported:

- ArcGIS Map Service
- WMS
- WMTS



These types of services are rendered as raster layers and can be attached to the iModel through the Map Layers widget.

Chances are your GIS department already has existing maps available and ready to use in your project. Alternatively, you can use any GIS server (ArcGIS, open-source Map Service, GeoServer, etc.) to compose your map, add symbology and publish it to the format that best suits your needs. Contact your GIS Administrator for more information.

### ArcGIS REST Map Service

ArcGIS can serve data through multiple services like Map Service, ImageService and Feature Service. However, not all types of services can be attached to the iModel through the Map Layers widget. Only the *Map Service* can be attached and displayed in iTwin Viewer through the Map Layers widget. The support of Feature Service will be added shortly. In the meantime, if you want to display data coming from ArcGIS Feature Service, please consult the section “*Synchronizing GIS data to an iModel*”.

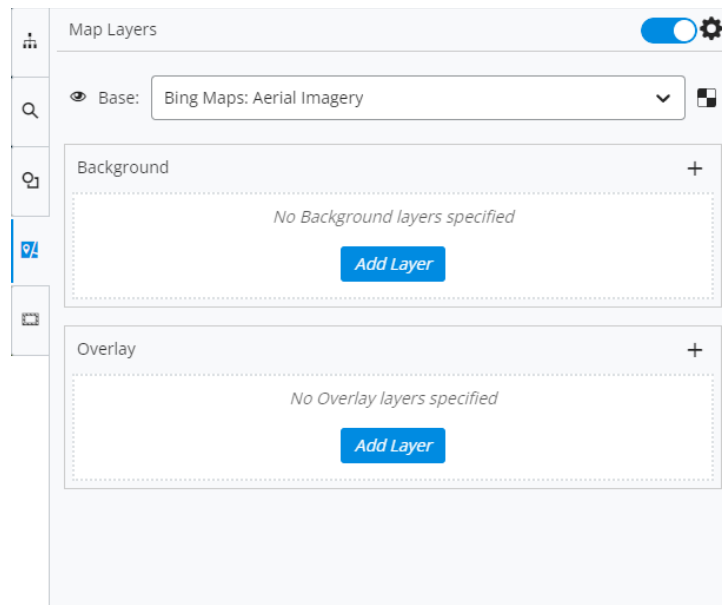
A good indicator to find out if a layer can be referenced to the iModel is the presence of “MapServer” at the end of the ArcGIS REST URL that is added in the Map Layers dialog. See the figure below.

Open Geospatial Consortium WMS and WMTS Services can also be attached to the iModel, through the Map Layers widget.

Service Type	Server URL Examples
ArcGIS MapServer	https://<host>/<site>/rest/services/<folder>/<ServiceName>/<MapServer>
WMS	https://<host>/<site>/<folder>?service=WMS https://<host>/<site>/rest/services/<folder>/<ServiceName>/<MapServer>/WMSServer
WMTS	https://<host>/<site>/<folder>?service=WMTS https://<host>/<site>/rest/services/<folder>/<ServiceName>/<MapServer>/WMTS

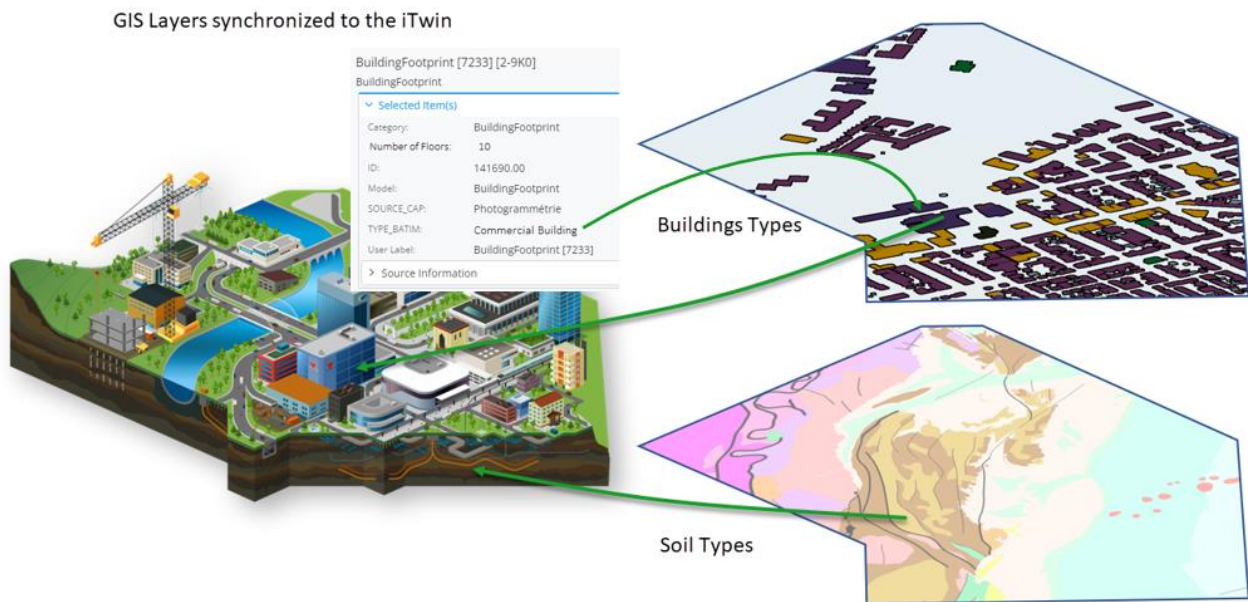
Adding ArcGIS/WMS/WMTS layers in the Map Layers widget

The iTwin Viewer Map Layers widget manages base layer selection and display settings for all layers.



For more information on the different options and settings, please refer to the [Map Layers Wiki page](#).

## Synchronizing GIS data to an iModel



Synchronizing data copies GIS data to the iModel, and ensures that you can Search, perform Design Validation, and offer formats not supported by the Reference option.

GIS data can be brought into an iModel using the Geospatial iModel connector. The GeoConnector is one of the multiple 'connectors' Bentley offers to synchronize data to an iModel. It supports many different GIS formats, and it allows full business property access, versioning (provided by iModels) etc. The data can be synchronized whenever required, as often as necessary, giving access to the latest updates made to the data.

The GeoConnector is being developed to support geospatial data from multiple sources.

### Supported Formats

Presently, the following formats are supported by the GeoConnector:

- Shapefile
- PostGIS database - through a *.geodb* file
- WFS (Web Feature Service) - through a *.geodb* file
- ArcGIS FeatureService - through a *.geodb* file
- ESRI File Geodatabase
- Kml
- GeoJSON
- Oracle Spatial database - through a *.geodb* file
- SQL Spatial database - through a *.geodb* file

### Strengths and Limitations

This iTwin.js Synchronize solution has its strengths and its limitations.

The access to the feature properties is one of the most valuable benefits. When GIS data is synchronized, the geometries and their properties are all copied into the iModel, allowing it to perform operations like queries, searches, and Design validation.

The principal limitation is maintaining symbology. When processing GIS data into the GeoConnector, the existing symbology created in your GIS system is lost. It must be re-created from scratch, with the help of a configuration file that is edited manually, with basic options like the color outlines and fills, the line weight and the transparency values. Investigation to support the ArcGIS Feature Service symbology is underway.

Another limitation could be that some formats cannot be processed on the Portal synchronizer due to security reasons. For example, if you want to synchronize a PostGIS database, you need to provide credentials. The credentials are stored in a file and this file must remain on a local machine for security reasons. So, there is no choice but to process this type of GIS data on the desktop client synchronizer.

Please refer to Data Location column in the chart shown earlier in this document for more details on which formats can be processed where and how.

If you need the GIS properties but do not need to do operations like Design Validation, queries, and searches, we recommend using the Map Layers tool with the possibility to display the properties when this functionality is available in a coming version iTwin.js.

For more details on the Synchronizer GeoConnector, please refer to the [GeoConnector wiki page](#).