



Using Spatial NI Web Services and APIs

A Developer's Guide



Land & Property Services
Seirbhísí Talún agus Maoine



Department of
Finance | An Roinn
Airgeadais

Table of Contents

1.	Introduction	3
2.	Understanding Web Services, APIs, and Endpoints	3
3.	Spatial NI Read-Only Web Service Model.....	3
4.	Spatial NI Cached Services vs. Dynamic Services	4
5.	Spatial NI Web Map Service Types.....	4
6.	Spatial NI Locator Service	10
7.	Spatial NI Hosting Platform	10
8.	Spatial NI Service Authentication	10
9.	Spatial NI Architecture Overview	11
10.	Spatial NI Integration Options	12
11.	Spatial NI Web Service Catalogue	13
12.	Spatial NI Web Service Access Request Workflow	13
13.	Conclusion	14

1. Introduction

This guide provides technical direction for NIMA participants and their sub-contractors on integrating authoritative geospatial data from Ordnance Survey of Northern Ireland (OSNI) using Spatial NI web map services. Aimed at developers and GIS professionals, it introduces key concepts such as web services, APIs, and endpoints. It also explains the Spatial NI service model and outlines the steps for requesting access.

Spatial NI offers both OGC-compliant and ArcGIS REST services, supporting broad interoperability and vendor-neutral integration. The platform operates a read-only service model, prioritising performance and data integrity, with authentication managed through username and password credentials. All services are hosted on Esri ArcGIS Server Version 11.3.

2. Understanding Web Services, APIs, and Endpoints

Understanding the roles and differences between web services, APIs, and endpoints helps clarify how systems and applications connect to exchange geospatial data. For example, when you check the rainfall in your area using a weather app, the app uses a web service and an API to get map data from another system. The endpoint is the specific address the app connects to so it can request that data. Together, these tools make it easy for information to move between different systems and applications.

Web services are network-based services that enable systems to communicate over the internet, typically using standards or protocols such as OGC WMS, WFS or ArcGIS REST. In this context, the term "web services" commonly refers to the actual endpoints that deliver data to client applications.

An **endpoint** is a specific URL that represents a function or service within an API or web service. It is the location where a service can be accessed by a client and where requests are sent to interact with that service.

APIs (Application Programming Interfaces) are a broader concept; they define how software components interact. A web service is a type of API, but not all APIs are web services. For example, the ArcGIS REST API is an API that exposes web services. People often use the term "APIs" when referring to the interface or specification used to interact with those services.

Throughout this document, we use the term "web service" to refer collectively to both OGC-compliant services and RESTful endpoints. Although these are technically types of APIs, "web service" is used here as a simplified and consistent reference.

3. Spatial NI Read-Only Web Service Model

Spatial NI web services are delivered as Map Services rather than Feature Services, which limits advanced capabilities such as feature editing, syncing, and transactional updates. This design choice prioritises performance and stability for viewing and querying spatial data, rather than enabling direct data manipulation.

Map services are designed primarily for read-only access, allowing users to visualise spatial data without modifying the underlying datasets. Depending on the service type and how it has been configured, querying the data may also be supported. This approach is especially valuable in environments where maintaining data integrity and consistency is critical.

4. Spatial NI Cached Services vs. Dynamic Services

Spatial NI map services are delivered as either cached tiles or dynamically rendered maps, depending on their intended use. The choice between cached and dynamic rendering affects both performance and functionality.

Cached services are optimised for fast and efficient visualisation of geographic data. They use pre-rendered image tiles across various zoom levels, enabling smooth navigation and high performance. Designed primarily for display purposes, cached services do not support data querying.

Spatial NI's main cached service, the OSNI Fusion Basemap, support zoom scales ranging from highly detailed views at 1:500 to broad overviews at 1:2 million. This allows users to explore geographic data at varying levels of detail, from granular urban landscapes to wide regional perspectives.

Dynamic services, by contrast, are rendered on-the-fly in response to user requests. This means map images are generated in real time, enabling interactive operations such as querying, feature selection, and geometry-based analysis.

In summary, cached services are ideal for speed and visualisation, while dynamic services offer more advanced capabilities for data interaction and analysis.

5. Spatial NI Web Map Service Types

Spatial NI map services are available in both OGC-compliant and ArcGIS REST. The following section outlines the core operations supported by each service type.

OGC-compliant Services

These services adhere to Open Geospatial Consortium (OGC) standards, ensuring broad interoperability across platforms and applications that support OGC protocols. Spatial NI provides two primary types of OGC-compliant services: WMS (Web Map Service) and WMTS (Web Map Tile Service). Both services are read-only by design, meaning they do not expose or allow direct interaction with the underlying feature data.

- **WMS** delivers dynamically rendered map images (e.g., PNG, JPEG) generated on-the-fly in response to user requests. This enables flexible styling and layer composition but may introduce latency under high-demand conditions.

- **WMTS**, in contrast, serves pre-rendered map tiles that are cached and optimized for rapid delivery. This makes WMTS highly scalable and well-suited for performance-critical environments such as web and mobile applications.

Example Endpoint Formats

The following two examples illustrate typical endpoint formats used to access Spatial NI web services that conform to OGC standards.

- A **WMS Endpoint** is a specific URL that allows users or applications to request map images from a WMS server.

```
https://services.spatialni.gov.uk/ogc/services/ServiceName/MapServer/WMSServer
```

- A **GetCapabilities** request is sent to a WMS endpoint to retrieve details about the service, including available map layers, supported formats, and operations. For instance, when a GIS application queries the Spatial NI WMS, it receives metadata that helps users understand how to interact with and make use of the service.

Key parameters:

SERVICE=WMS tells the server you're using the Web Map Service protocol.

REQUEST=GetCapabilities asks the server to return metadata about the service.

```
https://services.spatialni.gov.uk/ogc/services/ServiceName/MapServer/WMSServer
?SERVICE=WMS&REQUEST=GetCapabilities
```

Overview of Supported Operations and Functional Constraints

The following tables summarise the core operations supported by WMS and WMTS.

Supported Operations – WMS		
Operation	Purpose	What it returns
GetCapabilities	Provides metadata about the service and available data layers. Typically, the first request a client makes.	XML document detailing: • Available map layers • Supported coordinate reference systems (CRS) • Image formats • Supported operations and constraints
GetMap	Retrieves a map image of a specific geographic area.	Georeferenced image (e.g., PNG, JPEG) based on parameters like:

		• LAYERS • BBOX • CRS • WIDTH, HEIGHT • FORMAT
GetFeatureInfo	Retrieves attribute information about features at a specific location. It does not support full feature-level querying like filtering by attribute values or spatial queries.	Attribute data (e.g., name, type, value) using parameters like: • QUERY_LAYERS • I, J (pixel coordinates) • INFO_FORMAT

Supported Operations – WMTS		
Operation	Purpose	What it returns
GetCapabilities	Provides metadata about the WMTS service, including available layers, supported formats, tile matrix sets, and service constraints.	XML document detailing: • Available map layers • Supported CRS • Tile matrix sets • Image formats • Supported operations
GetTile	Retrieves a specific map tile based on zoom level and tile coordinates.	Static image tile (e.g., PNG, JPEG) using parameters like: • LAYER • STYLE • TILEMATRIX • TILEROW, TILECOL • FORMAT

OGC services such as WMS and WMTS are primarily designed for visualization rather than direct interaction with spatial data. While they adhere to open standards and ensure broad compatibility, they also impose functional limitations that users should understand.

Functional Constraints (WMS/WMTS)		
Limitation	Description	Impact
Read-Only Access	WMS and WMTS are read-only services, they do not support editing or exporting feature-level data.	Users can view map layers but cannot modify or extract underlying spatial features.
Limited Attribute Querying	WMS supports only basic attribute access through the <i>GetFeatureInfo</i> request, which returns information for a single pixel location on the map. WMTS does not support attribute querying at all.	Both WMTS and WMS are unsuitable for searching, filtering, or analysing data based on attribute values across multiple features.

No Spatial or Analytical Interactions	WMS and WMTS do not support geometry-based operations such as select by location, buffer, or intersect, nor do they enable image or vector analysis.	These limitations restrict workflows that require spatial analysis, raster classification, feature overlays, or precise interaction with geographic features (e.g. snapping to features).
Performance Trade-offs	WMS generates images on-the-fly; WMTS uses cached tiles.	WMS may be slower under load; WMTS is faster but less flexible in styling and updates.

In summary, OGC services are excellent for interoperability and visualisation, especially in vendor-neutral environments. However, for tasks requiring interactive querying, data analysis, or feature-level access, Esri ArcGIS REST services offer more robust capabilities.

ArcGIS REST Services

Spatial NI provides proprietary map services through the ArcGIS Server REST API. While these services are optimised for use within Esri platforms, they are also accessible in custom or non-Esri applications using standard web development tools. Spatial NI supports two main types of REST services:

- **MapServer**, which delivers dynamic, vector-based maps and supports interactive operations such as querying, feature selection, and geometry-based analysis.
- **ImageServer**, which provides access to raster datasets (e.g., aerial or satellite imagery), delivered as tiled image services for fast rendering and analysis. These raster tiles are typically generated from underlying vector or raster data, depending on the source.

Example Endpoint Formats

The following examples illustrate typical endpoint formats used to access Spatial NI ArcGIS Rest services.

- An **ArcGIS REST API Endpoint** is a specific URL that provides access to map layers and spatial data from a Map Server.

```
https://utility.arcgis.com/usrvcs/servers/<uniqueid>/rest/services/ServiceName/MapServer
```

- A **Query** request is a type of request sent to a Map Server using the ArcGIS REST API to retrieve features based on location or specific attributes (such as names, types, or values). For example, you might use this endpoint to find all buildings within a certain area or all roads with a particular name. This is especially useful when you need to extract targeted data for analysis, mapping, or integration with other tools.

This endpoint corresponds to **Layer 2** of the example map service. To query features from this layer, you can use the /query operation.

<https://utility.arcgis.com/usrvcs/servers/<uniqueid>/rest/services/ServiceName/MapServer/2/query>

Overview of Supported Operations and Functional Constraints

The table below outlines the core operations supported by Spatial NI's MapServer and ImageServer services.

Supported Operations (MapServer/ImageServer)				
Layer Type	Rendering Type	Operations supported	Purpose	Underlying Server
Map Image Layer	Dynamically rendered image tiles based on vector data	View, identify, and limited query operations (depending on configuration)	Suitable for dynamic visualisation where real-time rendering is required. Supports up-to-date display of changing datasets and can accommodate scale-dependent rendering and symbology.	MapServer
Feature Layer (Map Service)	Dynamically rendered vector data (points, lines, polygons)	View, analyse, query, change symbology, snap to geographic features.	Supports interactive, vector-based visualisation and spatial analysis.	MapServer
Tile Layer	Pre-rendered raster tiles	Fast, cached map tiles for display only. No query or identify operations supported.	Ideal for basemaps and static visualisation where performance and scalability are prioritised over interactivity.	ImageServer
Imagery Layer	Dynamically rendered raster data	Visualisation of raster data, pixel-level querying and raster analysis (e.g. NDVI, NIR, Hillshade, Slope).	Supports imagery-based visualisation and analysis, particularly for environmental monitoring, terrain modelling, and remote sensing applications.	ImageServer

Important Usage Considerations for ArcGIS REST Services

In addition to understanding the operational capabilities of MapServer and ImageServer, it's equally important to recognise their limitations and behaviours that may vary depending on configuration. The following section highlights key considerations to keep in mind when using these services.

Feature Layers within Map Services

Spatial NI supports Feature Layers embedded within Map Services, but not standalone Feature Services. This distinction has several implications:

- **Editing operations** (e.g. add, update, delete) are **not supported**, as these require a dedicated Feature Service. Therefore, operations such as ApplyEdits, AddFeatures, or UpdateFeatures are unavailable.
- **Querying and data interaction** are supported depending on the Map Service configuration. Users can perform attribute and spatial queries, view feature details, and export selected records where permitted.

In summary, while direct data editing is not available, Feature Layers within Map Services still support meaningful spatial and attribute-based analysis.

Functional Constraints (MapServer/ImageServer)

These constraints are influenced by whether the service is configured as dynamic or cached.

Functional Constraints (MapServer/ImageServer)		
Limitation	Description	Impact
Maximum Record Count	Each request is limited to a maximum of 2,000 features.	Large datasets may require pagination or multiple requests to retrieve all records.
File Export Restrictions	Map Services do not support direct export to shapefile or file geodatabase formats, unlike Feature Services.	Limits offline data sharing and integration with non-Esri tools; alternative export methods may be needed.
Access to Attributes and Geometry	Access to feature attributes and geometry is typically available in dynamic services, whereas cached services are designed for fast visual rendering. Actual capabilities depend on how the service is configured.	Cached services do not support querying or detailed data extraction; dynamic services offer more flexibility for analysis.

6. Spatial NI Locator Service

Spatial NI also provides locator services - dynamic geocoding endpoints hosted on ArcGIS Server as GeocodeServer services. These endpoints use custom locators built from the OSNI Pointer Address dataset.

The service supports core operations such as *FindAddressCandidates*, *ReverseGeocode*, *Suggest*, and *GeocodeAddresses*, enabling both interactive and automated location search workflows.

While it integrates seamlessly with Esri platforms such as ArcGIS Pro and ArcGIS Online, the Locator Service can also be accessed from non-Esri clients and custom applications via the ArcGIS REST API. This allows developers to consume the service in other technologies, provided they handle authentication and RESTful requests appropriately.

7. Spatial NI Hosting Platform

All Spatial NI map services, including both OGC-compliant services and those based on the ArcGIS REST API, are hosted by ArcGIS Server, part of the Esri ArcGIS platform. This hosting environment ensures high performance, scalability, and broad compatibility with a wide range of GIS applications and clients.

ArcGIS Server also enables seamless interoperability, allowing users and developers to integrate Spatial NI services into their own systems using standard tools, Esri-compatible clients, and custom workflows.

For further guidance, please refer to the relevant ArcGIS Technical Support article:

[Map Service](#) | [ArcGIS REST APIs](#) | [ArcGIS Developers](#)

8. Spatial NI Service Authentication

This section outlines how Spatial NI web services are secured and how authentication is managed across different service types. Authentication methods vary depending on whether the service is OGC-compliant or based on ArcGIS REST. All Spatial NI services use username and password-based authentication, rather than API keys. The handling of credentials depends on the service type and its integration method.

Authentication by Service Type		
Service Type	Authentication Method	Details
OGC Services (WMS/WMTS)	Web-Tier Digest Authentication	Credentials (username and password) must be embedded in the client or application configuration. Digest Authentication follows HTTP RFC 7616 and must be used over HTTPS to ensure encrypted communication. OGC services are deployed using Apache Tomcat as the web server and Active Directory for user authentication.
ArcGIS REST Services (MapServer/ImageServer)	GIS-Tier Authentication via ArcGIS Online Proxy	Credentials are securely stored in ArcGIS Online and managed through a proxy URL (e.g. utility.arcgis.com). This proxy handles authentication behind the scenes, allowing services to be used in web maps and applications without exposing credentials. Access is restricted to whitelisted application domains provided by customers via the Spatial NI web service request form.

For further guidance on authentication in ArcGIS environments, refer to the following Esri technical support articles:

- [Types of authentication | ArcGIS Maps SDK for .NET | Esri Developer](#)
- [FAQ: What is the Difference between Web-Tier vs. GIS-Tier Functionality in ArcGIS Server?](#)
- [FAQ: Why Are Secured Services Added to ArcGIS Online with Stored Credentials Remapped to utility.arcgis.com](#)

9. Spatial NI Architecture Overview

The following section provides a high-level overview of the Spatial NI IT architecture, illustrating how requests are routed and secured for both OGC-compliant and ArcGIS REST services.

- **OGC-compliant Services**

Spatial NI OGC map services are hosted within the Spatial NI infrastructure. Incoming requests are routed through the IT Assist Firewall and the IT Assist Application Security Manager (ASM) before reaching the ArcGIS Enterprise environment located in the DMZ (Demilitarized Zone). Figure 1 illustrates a simplified view of the request path.

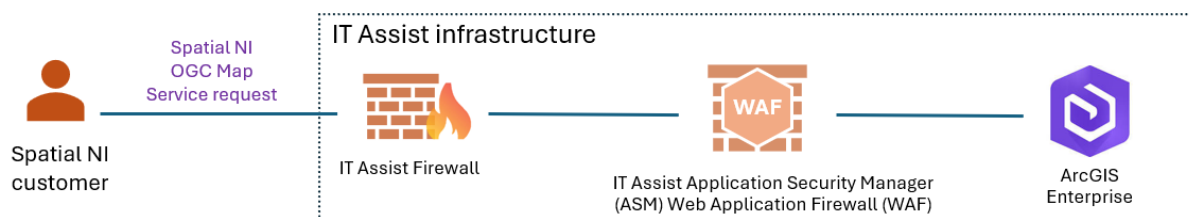


Figure 1: Spatial NI OGC service request path

- **ArcGIS REST Services**

Spatial NI ArcGIS REST services are proxied through ArcGIS Online, then routed through the IT Assist infrastructure, including the ASM, and finally to ArcGIS Enterprise in the DMZ.

Figure 2 illustrates a simplified view of the request path.

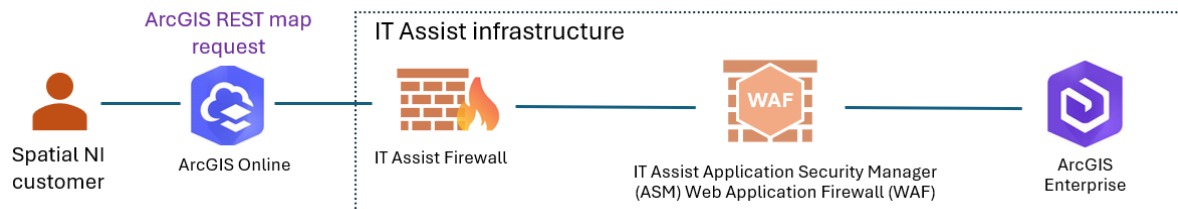


Figure 2: Spatial NI ArcGIS REST service request path

10. Spatial NI Integration Options

Spatial NI web services can be integrated into a wide range of desktop GIS and web-based applications. These services support flexible access to geospatial data, enabling use across various platforms, development environments, and analytical workflows.

Below is a representative list of commonly used applications, frameworks, and programming languages that support integration with OGC-compliant and/or ArcGIS REST services.

Desktop GIS Applications

- QGIS – Supports WMS, WFS, WMTS, and ArcGIS REST.
- ArcGIS Pro – Fully supports ArcGIS REST and OGC services.
- AutoCAD Map 3D – Can connect to WMS/WFS services and (ArcGIS REST services via ArcGIS for AutoCAD plugin).
- MapInfo Pro – Supports OGC services via plugins.

Web-Based Applications

- JavaScript Mapping Libraries:
 - Leaflet – Lightweight, open-source library for interactive maps.
 - OpenLayers – Robust library for working with OGC standards like WMS/WFS.
 - ArcGIS API for JavaScript – Esri’s official SDK for building web mapping applications.
 - MapLibre GL JS – Open-source alternative to Mapbox GL JS, supports vector tiles and WebGL rendering.
 - CesiumJS – Open-source library for 3D geospatial visualization; supports WMS/WFS via plugins.
- Web Frameworks
 - React, Vue.js, Angular – Often used with mapping libraries to build interactive apps.

Programming Languages

- Python:
 - Libraries like OWSLib, geopandas, and requests can consume OGC services.
 - ArcGIS Python API for working with ArcGIS REST services.
- R:
 - Packages like sf, leaflet, and mapview support WMS/WFS integration.

11. Spatial NI Web Service Catalogue

Spatial NI provides secure access to OSNI's authoritative geospatial data through a diverse range of web services. These services are available to NIMA participants and their sub-contractors via OGC-compliant interfaces or ArcGIS REST endpoints.

In addition, Spatial NI offers a small set of **developer-focused web map services** that are exempt from referrer URL restrictions to support testing and development. These simplified versions of OSNI's data are strictly intended for non-production use.

If you require access to a specific Spatial NI web service for development purposes that is **not listed in the catalogue**, please contact the Spatial NI team to discuss your requirements. The catalogue is updated regularly as new services become available.

Use the link below to explore the full catalogue of available web services:

[Spatial NI Web Service Catalogue](#)

12. Spatial NI Web Service Access Request Workflow

To ensure secure and appropriate use of Spatial NI web services, access is managed through a formal request process (including requests for test services). This section explains the steps to obtain access, covering licensing prerequisites and service delivery details. The guidance below will help you navigate the process efficiently.

Step 1 – Submit a Request via the Spatial NI Web Service Request Form

To integrate with Spatial NI web services, customers must complete the **Spatial NI Web Service Request Form**. Before submitting, ensure the form is fully completed and that a valid OSNI licence agreement is in place.

If you need help completing the form or have questions about service suitability or licensing requirements, please contact your NIMA or LGMA support officer, or the LPS Copyright team.

- NIMA/LGMA support team: nima.support@finance-ni.gov.uk
- LPS Copyright Team: copyright@finance-ni.gov.uk (licensing enquiries only)

To access the form:

Press **CTRL + Click** on the link below, then select **Download > Download file**. The form will be saved to your **Downloads** folder. [Spatial NI Web Service Request Form](#)

Please note: the form is only available as a downloadable file and cannot be viewed online.

Step 2 – Submit the Completed Form to the Spatial NI Team

Once the form is complete and all licensing requirements are met, email the form and any supporting documentation to the Spatial NI team using the following link:

[Submit Spatial NI Web Service Request Form](#).

Important: Spatial NI cannot process request forms that do not meet submission requirements. To ensure timely handling, all required fields must be completed accurately and all necessary documentation included.

Step 3 – Receive Access Credentials and Service Details

After your request is processed, you will receive an email from the Spatial NI team:

- **OGC Services:** The email will include a username and password, which must be configured within your application to authenticate access.
- **ArcGIS REST Services:** You will receive a utility URL with embedded credentials, allowing secure access without manually entering or exposing your username and password.

13. Conclusion

This guide has outlined the core technical requirements for the secure and efficient integration of Spatial NI web services. It is intended to help developers, GIS analysts, and public sector users seamlessly access and embed OSNI's authoritative geospatial data into business systems and workflows, ultimately enabling better decision-making for the benefit of the public.