Karnataka GIS

Standards

Volume-II : Data Exchange Protocol

KSRSAC

Karnataka State Remote Sensing Applications Centre

Dept. of IT, BT and S&T, Government of Karnataka

April 2017

KARNATAKA GIS STANDARDS

VOLUME-II: Data Exchange Protocol

KARNATAKA STATE REMOTE SENSING APPLICATIONS CENTRE

Dept. of IT, BT and S&T, Government of Karnataka

April, 2017

Contributors:

Ms R Shilpa

Mr Vilas H Chavan

Mr Madhu Kumar

Mr.M.Jayachadran

Version Controlled by:

Ms R Shilpa

Document Control Sheet

Document Number	KSRSAC/K-GIS/STANDARDS/03/APRIL-2017
Title	Karnataka GIS Standards, Volume- II: Data Exchange Protocol
Type of Document	Technical Report – Ver 3.1
Number of pages	**40**
Publisher	Karnataka State Remote Sensing Applications Centre
Prepared by	K-GIS Technical Team
Reviewed by	Mission Director, Chief Technical Officer & Technical Officer
Approved by	Technical Committee, KGIS
Abstract	Karnataka State Remote Sensing Applications Centre is implementing Karnataka- GIS which envisions maintaining a State-wide, Standardized, seamless and most current GIS asset and providing GIS based Decision Support System (DSS) for governance, private enterprise and citizen. Data being critical part for the DSS applications, Sharing of data across the departments is one of the important aspects. This document defines the protocol for exchange of data.
Distribution	Unrestricted
Reproduction Rights	This report and its contents are the property of KSRSAC

This page is intentionally left blank

Contents

List of Tables

List of Figures

This page is intentionally left blank

1. Background

1. Government of Karnataka is implementing Karnataka GIS- a comprehensive GIS based Decision Support System for good governance, sustainable development and citizen empowerment. KSRSAC an autonomous agency working under IT/BT, GoK is recognized as implementing agency for K-GIS.

2. K-GIS envisions to maintaining a state-wide, standardised, seamless and most current GIS asset for Karnataka through GIS Decision Support System for Government, Enterprise and Citizen.

3. KSRSAC has published K-GIS Standards – a suite of technical standards and protocols for K- GIS that will allow easy GIS Asset organisation/maintenance, GIS services of Data and Applications on a standards-based K-GIS Portal.

4. K-GIS Content standards have more than 200 layers classified under different categories. The summary of the content is shown the **Table-1.1**. The list is tentative and is expected to increase over period of time.

Table 1.1: Summary of Content Standards

NO	CATEGORY	NO OF ASSETS
1	Boundary	43
2	Cultural Features	8
3	Hydrology	7
4	Urban and Settlement	2
5	Industrial content	2
6	Forest Environment & Ecology	6
7	Geological	5
8	Land Information	2
9	Land Ownership Information	3
10	Terrain Information	2
11	Soil Information	1
12	Public Assets/ Amenities	132
13	Point Of Interest	5
14	Graticules	1
15	Raster Data	3
TOTAL		**222**

5. All the GIS layers will be having corresponding MIS data coming from different sources. MIS data resides and maintained in each department in one or the other form. In order to address the queries in the GIS application, MIS need to be linked with GIS information.

6. KGIS is a platform for all departments under GoK to collaborate and share the GIS and associated MIS. The departments which are responsible for creating and managing the physical assets shall also be the owners of the GIS data. The maintenance and management of the GIS / MIS data shall reside with the respective department.

7. A data exchange is the sharing of a specific type of data between two or more partners. In this document we refer to data exchange between KGIS KSRSAC and other GoK departments.

8. This document explains the different approaches of sharing / exchanging data (spatial (GIS) and attribute (MIS)) between KGIS and respective department / applications.

9. Data Exchange of attribute info can happen through the following modes
    - sharing soft copies of the data maintained in flat files (Excel, Comma Separated Values etc.) or
    - direct one to one access to the data maintained in the enterprise database or
    - data exchanged through web services.

10. The most appropriate and accepted form of exchange happens through service oriented architecture (SoA); which is more secured and reliable for exchange of dynamic voluminous data from difference sources. The concept of Service Oriented Architecture is explained in Chapter 3.

2. Data Exchange – Type and Mode

1. There are 2 types of data which are generally shared between the departments as envisaged in KGIS
    - Spatial Data (maps, locations, coordinates, geo-images etc)
    - Non-spatial / Attribute Data (departmental MIS data)

2. The Geospatial data available with the departments can be shared / exchanged in the following modes for both spatial and non-Spatial data
    - Offline mode – sharing softcopies of data in CD / DVD media or through emails
    - Online mode
        i. Through direct database connectivity to the database
        ii. Through web services

3. All data exchanges are done for making the available spatial data with its attributes for use of its internal department use / for sharing data to other departments and public through KGIS platform.

4. The data exchange between KGIS database / applications and the MIS database of the respective departments are done through a unique code established between the spatial database available within KGIS and the MIS database.

5. KGIS shall generate a unique spatial ID for every entity object created, layerwise. This is generally referred as KGIS-ID. This unique number shall be a running number for every object that is created, in a layer. KGIS ID – the unique spatial object identifier, once generated will never be regenerated so that the option / possibility of creating the same ID shall never happen. KGIS ID shall be used as a unique identifier within KGIS applications whenever a query or application requirements need to know or track the history of spatial changes happened.

6. The spatial changes on the various layers also shall be tracked through the versioning option of all layers. All layers shall have the versioning @ 2 levels- by default. More version layers shall be planned based on the application requirement for hierarchical approvals. The base version of database shall always be used for display, generic query and applications requirements under KGIS. However, the departmental specific applications shall be used to validate the changes made on the versioned database and on approval, the data shall be merged with the base version of layers.

7. Along with the KGIS-ID, every point asset / spatial object of an asset layer shall also have KGIS-code. KGIS code is the code generated to identify any asset based on the

geographical position and the category of asset and its count in a village / ward. The KGIS-code pattern and standard is explained in the database design document and also on the Asset design document of Version 1.0 and 2.0.

8. The KGIS database schema document (Volume 1- Part 2 of standards document) explains and mentions the availability of KGIS-ID and / KGIS-Code in every layer along with the schema.

9. Apart, from the KGIS-ID / KGIS Code, provision is made in the KGIS database to establish an additional link field (department code). The department code is an unique field which is already available in an existing MIS database of the department. This linkage is proposed to enable the departments to continue to follow the existing departmental code to link and extract the related spatial data.

10. In nutshell, KGIS ID, KGIS Code and the departmental code are unique in its nature and design. One-to-one relationship is established among these 3 codes for each layer. Thus any one of these codes shall be available between the MIS database and KGIS spatial database.

11. This unique linking key shall be used for data exchange / data updation between the applications / database. It is very important that every MIS record having the characteristics of spatial objects should be uniquely coded. A one-time integration / correlation of Spatial data with the MIS data based on unique coding shall be carried out. A sample list of unique link code for few asset layers and its corresponding field name containing the department code for MIS data linkage is shown in the Annexure 8.

12. For addition / deletion of any new spatial objects and attributes, the departments shall have to work and establish a mechanism of simple web based application for GIS data update @ KGIS.

13. Departments shall mandatorily ensure that new MIS shall not be created without creating a spatial object and having a unique reference key for integration with its associated spatial data.

14. The preferred mode of data exchange is by having **direct online connectivity** to the database or to applications explained as below:

- The simplest way of exchanging data (spatial / MIS) data is by having direct connectivity to the spatial / MIS database.

- All such access should be given on read access mode by the publishing / owning agency.
- If needed, to avoid performance burden to the existing database tables, it is recommended that publishing agency shall create separate copy / view tables which are synchronized with the actual / main databases and given access to the consuming agency.
- The owners (publishing departments) of the database need to share a clear description of various database / table / fields available which are shared with the consuming agency.
- In nutshell, the database design document with clear alias names and description need to be shared. This is important for extracting the relevant data in an application by KGIS.
- It is to be noted that any online transaction database/tables shall not be given direct access to the consuming agency.
- The consuming agency shall ensure that no complex queries are generated on a large transaction table.
- Most of the GIS queries shall be based on the statistics / summary. ETL tools / data warehousing should be used to create such tables. KGIS shall access the summary statistics, for queries related to statistics / spatial spread / charts / graph preparation etc.
- Any query which needs specific details for any individual or a small set of spatial obejects shall be extracted from the live transactional database – on need basis. For ex : owner name of a specific property shall only be executed on the large transaction tables of Bhoomi

15. The process of **Offline Mode of Data Exchange** is explained as below:

- Offline mode of data exchange happens whenever the department does not have an existing application or database
- The data exchange can happen by offline mode by sharing soft copies of the data maintained in flat files (Excel, Comma Separated Values etc.) for attributes and CAD / GIS formats for spatial data.
- The CAD / GIS formats or any other format of location details (X,Y / Lat, Long etc) of assets shall be shared with a unique reference or identifier for each asset
- In the absence of a unique identifier, the unique code for spatial objects generated by KGIS (KGIS ID and/or KGIS code) shall be used for all future updation / transactions.
- The MIS data available with the departments should be shared in the form of flat files (Excel / database etc) and establishing a unique reference identifier for the MIS data and the spatial data.

- Any departmental MIS data without reference to spatial data shall not be used in KGIS (ex: details of employees, details of projects / programmes, facilities etc)
- All data exchange of flat files can happen through emails or CD / DVD media.

- The sharing of spatial and attribute info through flat files is the least recommended option due to the inherent gaps and complications in terms of maintaining periodic updates.

Data Updation

- After the first time data creation (either by department or KGIS), the departments shall use the KGIS portal for continuous monitoring, data updation and maintenance.
- Requisite workflow process or data updation mechanism shall be discussed with the KGIS team as per the operational needs of department.
- It is strongly recommended that all updates of attributes from departments shall be done through online mode.
- It is also visualized that some of the departmental process may generate new spatial data outside KGIS environment. (ex: creation of as-built / planned utility network in a department through its contractors)
- In such cases, the departments shall coordinate with KSRSAC to establish a mechanism of data update in terms of format and content standard, update frequency etc. The mode of data update into the central server is preferred to be carried out in online / automatic mode, as much as possible.

16. It is highly recommended that workflow based data updation is established and the same is used for updating the spatial data in KGIS.

17. The best approach of data exchange is by having web services developed at both ends of application / database which can be used for sharing data through XML standards of Service Oriented Architecture (SOA). This is the best approach of sharing data (spatial and non-spatial), online. As it is the most preferred approach, detailed description of service oriented architecture and standards adopted for the sharing are explained in Chapter 3 and Chapter 4, respectively.

3. Service Oriented Architecture

1. K-GIS propose to have Service oriented architecture (SOA); which facilitates heterogeneous application development, integration and sharing. Main aim of the SOA is to provide reusable services which can be consumed from different kinds of user applications.

2. SOA is a set of architectural concepts used for the development and integration of services. SOA deals with distributed computing, in which, consumers consume a set of interoperable services. Multiple consumers can consume a single service and vice versa. Web services can be used by anybody without being concerned about the platforms or the programming languages used for developing them. Components or elements of SOA is shown in the **Fig -3.1**

   - **Application frontend** are web application, they initiate and control all the enterprise system.

   - **Service** is a software component that encapsulates high business concepts.

     - **Contract** provides a specification of the purpose, functionality, constraints and usage of services.
     - **Interface** contains functionality/ operations which have a mechanism to accept input and give output as well as handle errors/ exceptions.
     - **Implementation** provides the required business logic and appropriate data.

   - **Service Repository** registers the service and their attributes to facilitate the discovery of the service.

   - **Service Bus** is an infrastructure setup for integrating applications and services by routing, transforming protocols, handling business events, security and managing interactions among services.

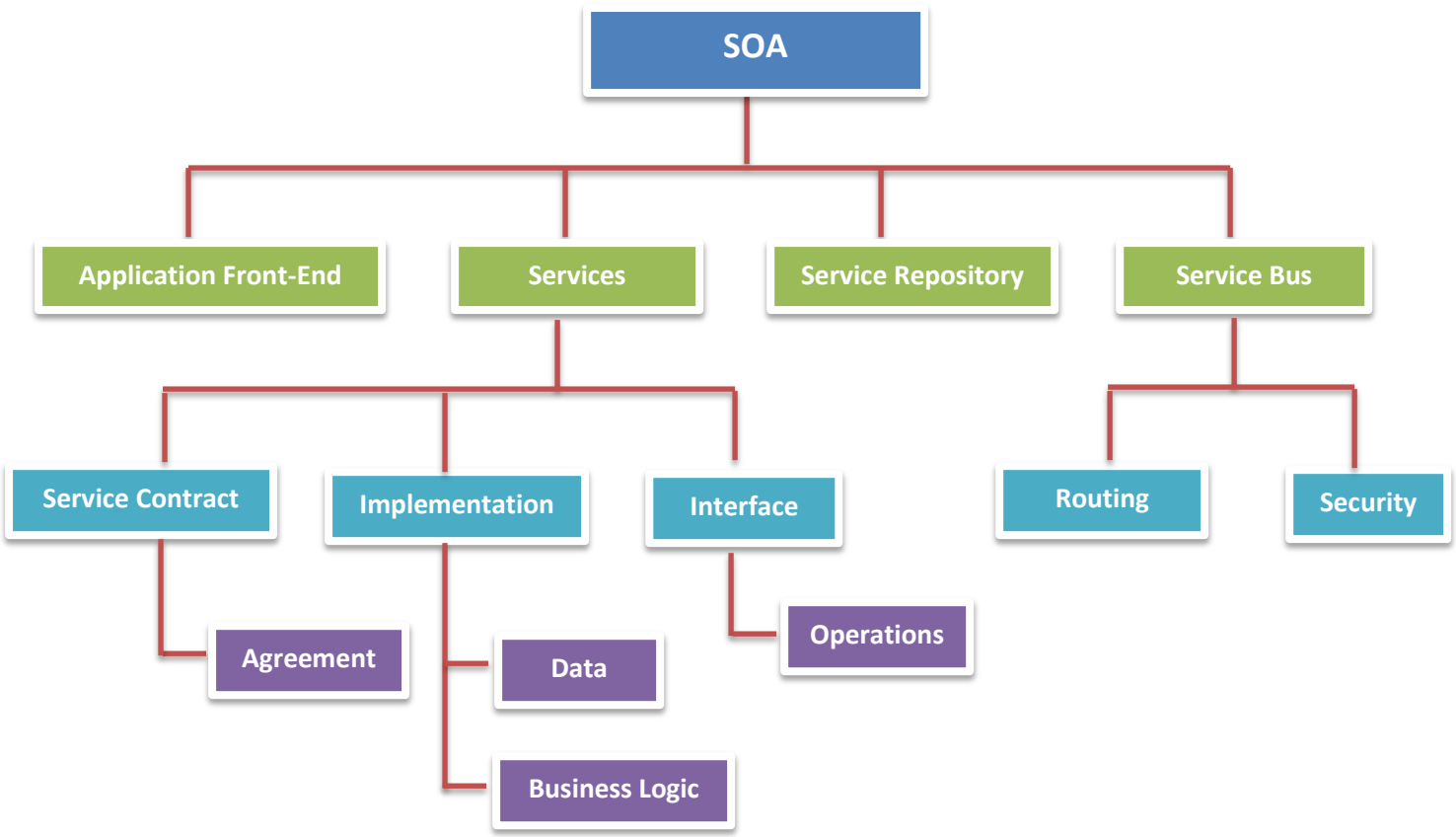

Fig 3.1: Elements of Service Oriented Architecture

3. Process flow of SOA can be best shown in **Fig- 3.2**

- **Service Repository/ Registry** – Is a resource which sets the access rights for data that is necessary for SOA. Registry allows service providers to discover and communicate with consumers efficiently. Contains Service Contract which helps **Service provider** in describing the services and fulfilling the request from Service Consumer. **Service Consumer** gets the results in the form of API for the consumption through Service Stub.

- **Service Contract** – Consists of service description document and its technical interfaces which establish an API into the functionality offered by the service. When services are implemented as Web services, the most common service description documents are the WSDL definition, XML schema definition, and WS-Policy definition.

- **Service Stub** – Represents the service for a client. It provides API that client can easily use. It includes information like network protocol, marshalling and unmarshalling of data, standard security mechanisms. The service dispatcher is the counterpart of the service stub. It receives incoming network requests that are generated by the service stub. It analyses the requests technically and invokes the request operations with appropriate data.

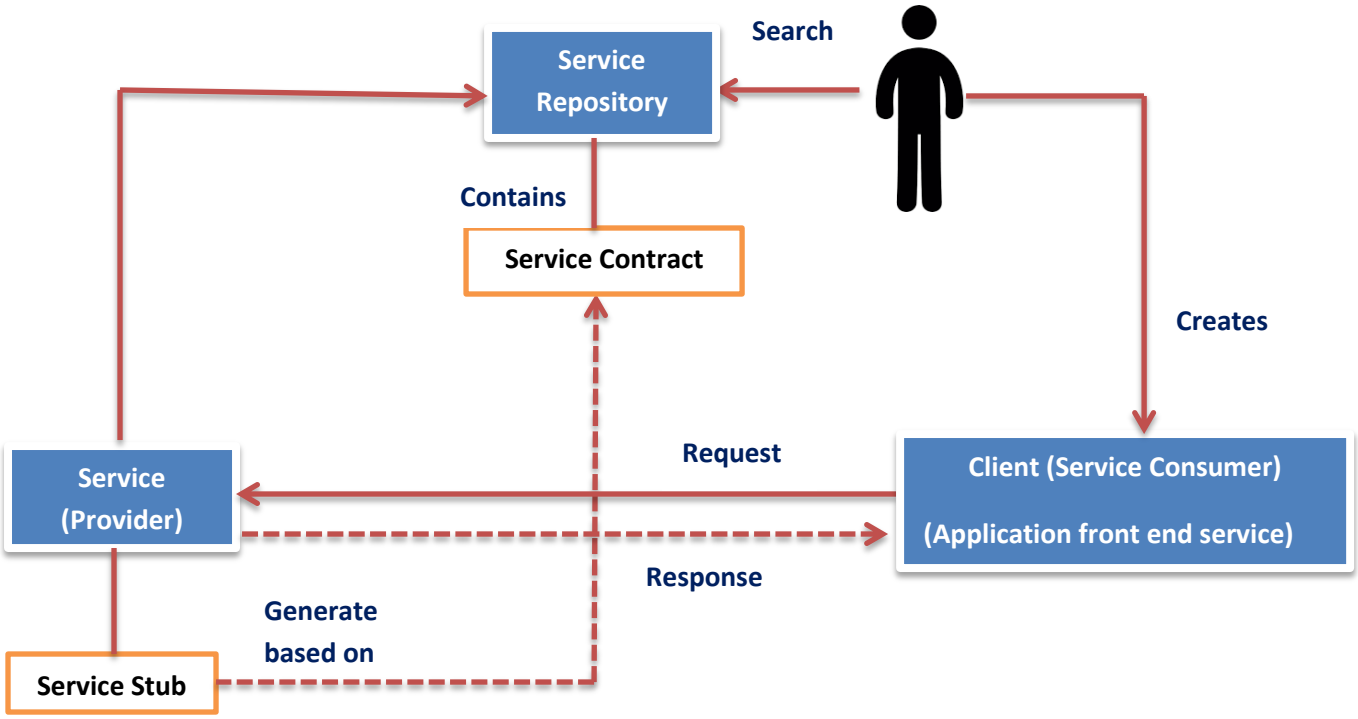

Fig 3.2: Process flow of Service Oriented Architecture

4. Definitions and explanation of key technologies used in Service oriented architecture are given below:

- **Web Services** – A Web service is a method of communication over the network. According to W3C, a Web service is a system dedicated for supporting machine-to-machine transactions over a network. It is a Web API described in Web Service Description Language (**WSDL**) and Web services are usually self-contained and self-describing. Web services can be discovered using Universal Description, Discovery and Integration (**UDDI**) protocol. By exchanging Simple Object Access Protocol (**SOAP**) messages typically over **HTTP** (with **XML**), other systems can interact with Web services. Web services are used in number of ways such as Remote Procedure Calls (**RPC)**, Service Oriented Architecture and Representational State Transfer (**REST**). **Fig – 3.3** represents the web service architecture.

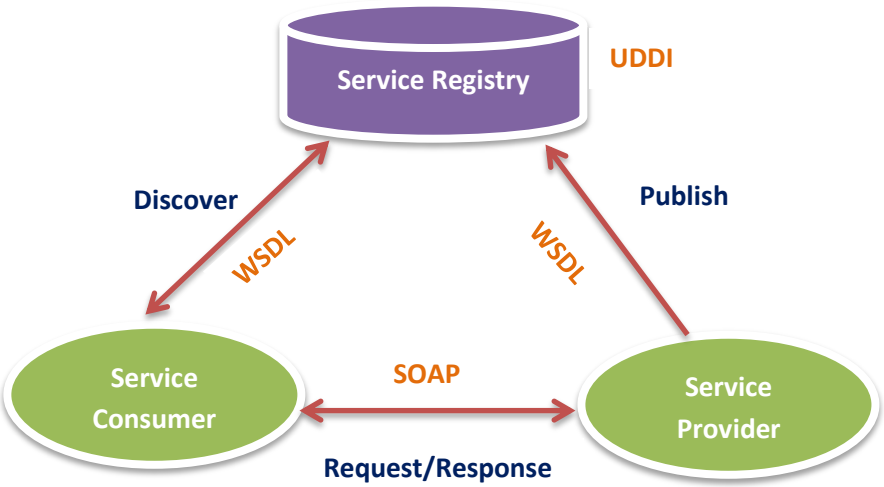

Fig 3.3: Web service Architecture

- **SOAP** – Simple Object Access Protocol (SOAP) defines a standard communication protocol (set of rules) specification for XML-based message exchange. SOAP uses different transport protocols, such as Hyper Text Transfer Protocol (**HTTP)** and Simple Mail Transfer Protocol (**SMTP**).

- **REST** –Representational State Transfer (REST) is an architectural style, and an approach to communications that is often used in the development of Web services. The use of REST is often preferred over the more heavyweight SOAP (Simple Object Access Protocol) style because REST does not leverage as much bandwidth, which makes it a better fit for use over the Internet. The SOAP approach requires writing or using a provided server program (to serve data) and a client program (to request data).

4. K-GIS Data Exchange Services

18. GIS and MIS data will be sourced from different departments. KSRSAC will be validating agency for ingest. Seamless data content of the state including the more detailed urban content data will reside at K-GIS infrastructure. **Fig- 4.1** depicts the schematic of centralized data exchange between KGIS and GoK departments.

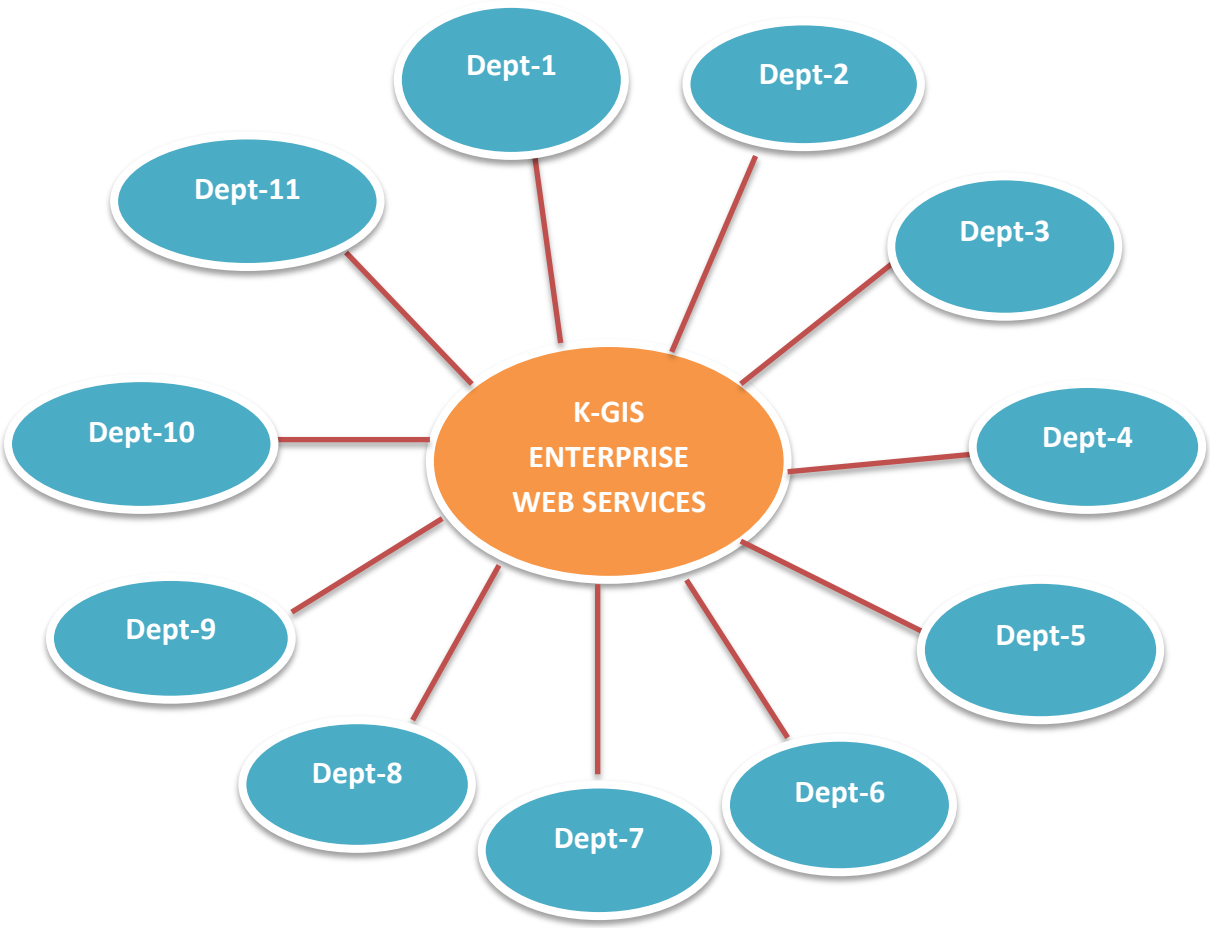

Fig 4.1: Data sharing through K-GIS

1. Exchange of data through web services between two or more applications is the professional and most appropriate and accepted form in a service oriented architecture; which is more secured and reliable for exchange of dynamic voluminous data from different sources.

2. Sharing of the web service through K-GIS happens in two different kinds of services; GIS and MIS services. Client can be requesting any GIS or MIS web services, which will be published through K-GIS. Data sharing across the department will be through web services for dynamic data updates. **Fig- 4.2** shows the K-GIS proposed Web Data Service:

- Spatial (GIS) Data service.

- MIS Data service.

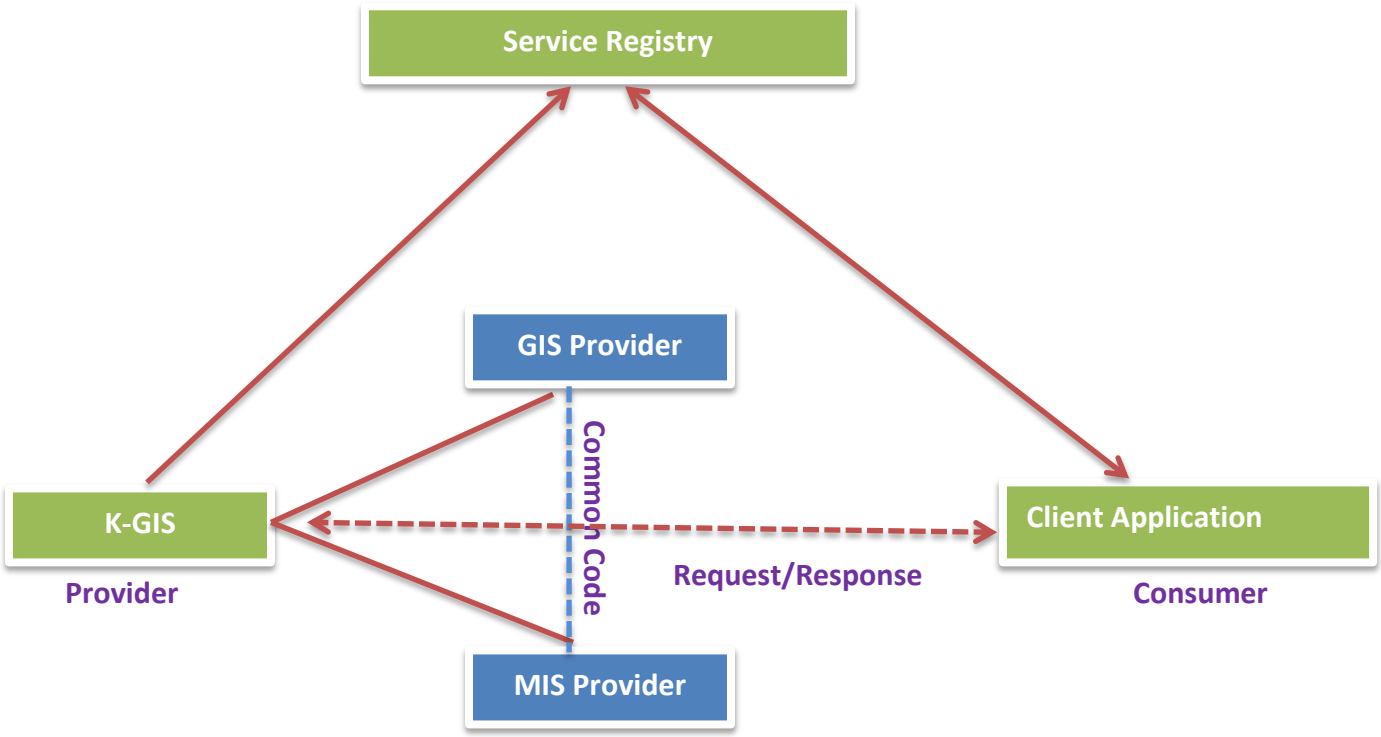

Fig 4.2: K-GIS Web service

1. Spatial Data are generally shared through REST services and non-spatial attribute data are shared through SOAP services

2. Spatial Data services will have the set of layers which will be shared for viewing, querying and editing operations. In general, for any web applications, the spatial data is accessed from a published service of maps / images. In KGIS, the spatial data shall be published through OGC standards as per various service needs, explained below. The similar data services are expected from departments to be published which shall be used by KGIS applications.

3. The departments which are having an existing application / portal can publish the spatial data for access in KGIS or other application. KGIS shall also validate / help in publishing spatial services from any dept. The Idealistic approach is to have all spatial data available in KGIS.

4. In case of spatial data being managed by respective department, a mechanism of data synchronization between databases, periodically shall be established.

5. All spatial data shall be published as WMS and WFS data. Details are explained, below.

6. The generic spatial data publishing services which will be shared are as follows:

- **WMS (Version 1.3.0)** - A Web Map Service (WMS) produces maps of spatially referenced data dynamically from geographic information. WMS-produced maps are generally rendered in a pictorial format such as PNG, GIF or JPEG, or occasionally as vector-based graphical elements in Scalable Vector Graphics (SVG) or Web Computer Graphics Metafile (WebCGM) formats. Sample response is added in **Annexures**. The operations supported by WMS are:
    - **GetCapabilities** - Retrieves metadata about the service, including supported operations and parameters, and a list of the available layers.
    - **GetMap** - Retrieves a map image for a specified area and content.
    - **GetFeatureInfo** - Retrieves the underlying data, including geometry and attribute values, for a pixel location on a map.
    - **GetLegendGraphic** - Retrieves a generated legend for a map.

- **WMTS (Version 1.0)** - A WMTS enabled server application can serve map tiles of spatially referenced data using tile images with predefined content, extent, and resolution. WMTS includes both resource (RESTful approach) and procedure oriented architectural styles (KVP and SOAP encoding). Sample response is added in **Annexures**. The operations supported by WMTS are:
    - **GetCapabilities** – It describes the abilities and information holdings of the specific server implementation.
    - **GetTile** – shows a fragment of a map representation of a layer.
    - **GetFeatureInfo** – It provides information about the features located at a particular pixel of a tile map, in a similar way to the WMS GetFeatureInfo operation.

- **WFS & WFS-T (Version 2.0)** - Web Feature Service allows a client to retrieve and update geospatial data encoded in Geography Markup Language (GML) from multiple Web Feature Services. The WFS –T supports operations like INSERT, UPDATE, DELETE, LOCK, QUERY and DISCOVERY operations on geographic features using HTTP as the distributed computing platform. Sample response is added in **Annexures**. The operations supported by WFS & WFS-T are:

- **GetCapabilities** – Generates a metadata document describing a WFS service provided by server as well as valid WFS operations and parameters.
- **DescribeFeatureType** – Returns a description of feature types supported by a WFS service.
- **GetFeature** – Returns a selection of features from a data source including geometry and attribute values.
- **LockFeature (WFS-T)** – Prevents a feature from being edited through a persistent feature lock
- **Transaction (WFS-T)** – Edits existing feature types by creating, updating, and deleting

- **WCS (Version 2.0.1)** - The Web Coverage Service (WCS) is digital geospatial information representing space-varying phenomena. One can think of it as Web Feature Service (WFS) for raster data. It gets the 'source code' of the map, but in this case it's not raw vectors but raw imagery. The results of a WCS can be used for complex modeling and analysis. It also allows more complex querying - clients can extract just the portion of the coverage that they need. WCS format encodings allow delivering coverages in various data formats, such as GML, GeoTIFF, HDF-EOS, CF-netCDF or NITF, JPEG etc. WCS requests and responses can be made using GET/KVP, POST/XML & SOAP/XML protocols. Sample response is added in **Annexures**. The operations supported by WCS are:
  - **GetCapabilities** – Retrieves a list of the server's data, as well as valid WCS operations and parameters
  - **DescribeCoverage** – Retrieves an XML document that fully describes the request coverages.
  - **GetCoverage** – Returns coverage in a well-known format. Like a WMS GetMap request, but with several extensions to support the retrieval of coverages.

- **WPS (Version 1.0)** - Web Processing Service (WPS) defines a standardized interface that facilitates the publishing of geospatial processes, and the discovery of and binding to those processes by clients. Processes include any algorithm, calculation or model that operates on spatially referenced data. A WPS can be configured to offer any sort of GIS functionality to clients across a network, including access to pre-programmed calculations and/or computation models that operate on spatially referenced data. WPS defines three operations for the discovery and execution of geospatial processes. Sample response is added in Annexures. The operations are:

- **GetCapabilities** – Requests details of the service offering, including service metadata and metadata describing the available processes.
- **DescribeProcess** – This operation allows a client to request and receive back detailed information about the processes that can be run on the service instance, including the inputs required, their allowable formats, and the outputs that can be produced.
- **Execute** – This operation allows a client to run a specified process implemented by the WPS, using provided input parameter values and returning the outputs produced.

- **City GML (Version 2.0)** - CityGML is an open data model and XML-based format for the storage and exchange of virtual 3D city models. It is an application schema for the Geography Markup Language version 3.1.1 (GML3), the extendible international standard for spatial data exchange. It defines the classes and relations for the most relevant topographic objects in cities and regional models with respect to their geometrical, topological, semantical and appearance properties. Included are generalization hierarchies between thematic classes, aggregations, relations between objects, and spatial properties. In contrast to other 3D vector formats, CityGML is based on a rich, general purpose information model in addition to geometry and graphics content that allows retaining virtual 3D city models for sophisticated analysis tasks in different application domains like simulations, urban data mining, facility management, and thematic inquiries.

7. KSRSAC will take the responsibility of publishing the seamless GIS layers services. Source of each layer will come from different departments but KSRSAC would be the validating and ingest agency for those layers. Access rights for each layer will be defined. Sharing of the data will be based upon **Karnataka Geospatial policy,** which is yet to define**.** All the GIS layers will be shared based on the kind of operations it supports; Viewing, Querying and Editing. Layers published can serve 2 different kinds of audience; Departmental and the general public. Anyone can use these services and build an application on top of it.

8. Security of data and the database shall be the responsibility of the agency publishing the services. The access right to view, download, edit / update shall be defined by the publishing agency.

9. K-GIS propose to have department MIS data shared through services. Sample WSDL is added in **Annexures**. Each department should publish their data services. Few instructions for publishing MIS services include:

- There should be a common code between GIS and MIS data for the retrieval of information. GIS and MIS providers should agree for one common coding scheme. KGIS Code shall be maintained by the MIS database and shall be used for linking.

- Department should publish the Web Service Description file which contains definition of the web service; its function name, input and output parameters, datatypes. Its protocol information, access information and error handling mechanisms.

- All the transactions which happen for a Web services will be stored in database for Audit trails (User access, operations, IP information etc.).

- Based on the access constraints, department can publish what is sharable and what is secured data. Each department should define the kind of data/ fields they are going to share with inter- department and general public.

- Multilingual data should be published in Unicode format.

10. It is to be noted that KGIS shall maintain an extracted summary of various live database in its own system through ETL (extract – Transform – Load) tools. It is envisaged that the queries on various statistics of departmental assets through KGIS shall be performed from these extracted summary tables. This shall ensure that the existing live departmental databases are not burdened with many queries on statistics which generally slows down the performance. However, any query on lowest / finest information of transaction shall be extracted from the respective departmental database.

- Example Scenario: KGIS is having the cadastral maps of complete Karnataka. Bhoomi system in SSLR / Revenue department maintains and updates the ownership, extent and usage of land in its bhoomi database.

- Expected sample queries on cadastral database through KGIS are as follows
    i. Total no of agricultural properties in a village
    ii. Total extent of Government lands
    iii. Total no of government lands in a particular district / Taluk etc
    iv. Show the ownership details of a specific property

- In the above scenario, except for the query (iv), all queries shall be addressed by the database available within KGIS – as many are static in nature. The summary / general update of such cases to KGIS database can be done through ETL tools based on defined update interval, as needed.

- The data update for data through ETL tools can be once in a day (preferably off business hours).

- In case of update of important / critical data between KGIS and departmental database, the update shall be live through direct web services.

- However, for specific details of owner name and extent of single property, the data shall be extracted from the bhoomi database through web services, as needed.

5. Summary

1. The preferred mode of data exchange is by establishing direct connectivity between spatial and MIS databases. The database schema / design and explanation of tables need to be shared between KGIS and department teams.

2. Data warehousing / ETL tools is preferred to be adopted to extract the summary and thereby using it for more generalized / statistical queries.

3. The best technically accepted mode of data exchange is through adoption of service oriented architecture.

4. Data Exchange in K-GIS happen through REST / SOAP based web services in an Service Oriented Architecture.

5. Data Exchange across the department shall happen through K-GIS.

6. Common unique coding scheme has to be agreed based on K-GIS standards. Unique codes shall be established and linked between 2 or more databases of user departments and KGIS spatial database.

7. One to One interactions for the development of services shall be done by K-GIS team and Partner departments – as needed for the defined application

8. Service Description file for each of the department including all the details for the data exchange should be defined and shared.

6. Annexures

1. Sample WDSL file for MIS data

```
<! -- <definition> defines the name, targetNamespace of the web service-->

<wsdl:definitions targetNamespace="http://tempuri.org/">

<! -- <type> defines the Function names used and the name of the parameter and its
datatype used in the function -->

<wsdl:types>
<s:schema elementFormDefault="qualified" targetNamespace="http://tempuri.org/">
<s:element name="get_grid_numbers_on_date">
<s:complexType>
<s:sequence>
<s:element minOccurs="0" maxOccurs="1" name="_date" type="s:string"/>
</s:sequence>
</s:complexType>
</s:element>
<s:element name="get_grid_numbers_on_dateResponse">
<s:complexType>
<s:sequence>
<s:element minOccurs="0" maxOccurs="1" name="get_grid_numbers_on_dateResult">
<s:complexType>
<s:sequence>
<s:element ref="s:schema"/>
<s:any/>
</s:sequence>
</s:complexType>
</s:element>
</s:sequence>
</s:complexType>
</s:element>
<s:element name="get_grid_numbers_between_date">
<s:complexType>
<s:sequence>
<s:element minOccurs="0" maxOccurs="1" name="_date1" type="s:string"/>
<s:element minOccurs="0" maxOccurs="1" name="_date2" type="s:string"/>
</s:sequence>
</s:complexType>
</s:element>
<s:element name="get_grid_numbers_between_dateResponse">
<s:complexType>
<s:sequence>
<s:element minOccurs="0" maxOccurs="1"
name="get_grid_numbers_between_dateResult">
<s:complexType>
<s:sequence>
<s:element ref="s:schema"/>
```

```
<s:any/>
</s:sequence>
</s:complexType>
</s:element>
</s:sequence>
</s:complexType>
</s:element>
<s:element name="get_data_from_inetermediate_table">
<s:complexType>
<s:sequence>
<s:element minOccurs="0" maxOccurs="1" name="grid_number" type="s:string"/>
</s:sequence>
</s:complexType>
</s:element>
<s:element name="get_data_from_inetermediate_tableResponse">
<s:complexType>
<s:sequence>
<s:element minOccurs="0" maxOccurs="1"
name="get_data_from_inetermediate_tableResult">
<s:complexType>
<s:sequence>
<s:element ref="s:schema"/>
<s:any/>
</s:sequence>
</s:complexType>
</s:element>
</s:sequence>
</s:complexType>
</s:element>
</s:schema>
</wsdl:types>

<! -- <message> has 2 messages; input (parameters) and output (return data). Part and type
are elements inside the message-->

<wsdl:message name="get_grid_numbers_on_dateSoapIn">
<wsdl:part name="parameters" element="tns:get_grid_numbers_on_date"/>
</wsdl:message>
<wsdl:message name="get_grid_numbers_on_dateSoapOut">
<wsdl:part name="parameters" element="tns:get_grid_numbers_on_dateResponse"/>
</wsdl:message>
<wsdl:message name="get_grid_numbers_between_dateSoapIn">
<wsdl:part name="parameters" element="tns:get_grid_numbers_between_date"/>
</wsdl:message>
<wsdl:message name="get_grid_numbers_between_dateSoapOut">
<wsdl:part name="parameters"
element="tns:get_grid_numbers_between_dateResponse"/>
</wsdl:message>
<wsdl:message name="get_data_from_inetermediate_tableSoapIn">
<wsdl:part name="parameters" element="tns:get_data_from_inetermediate_table"/>
</wsdl:message>
```

```
<wsdl:message name="get_data_from_inetermediate_tableSoapOut">
<wsdl:part name="parameters"
element="tns:get_data_from_inetermediate_tableResponse"/>
</wsdl:message>

<! -- <portType> combines one request and response into single operation -->


<wsdl:portType name="agriSoap">
<wsdl:operation name="get_grid_numbers_on_date">
<wsdl:input message="tns:get_grid_numbers_on_dateSoapIn"/>
<wsdl:output message="tns:get_grid_numbers_on_dateSoapOut"/>
</wsdl:operation>
<wsdl:operation name="get_grid_numbers_between_date">
<wsdl:input message="tns:get_grid_numbers_between_dateSoapIn"/>
<wsdl:output message="tns:get_grid_numbers_between_dateSoapOut"/>
</wsdl:operation>
<wsdl:operation name="get_data_from_inetermediate_table">
<wsdl:input message="tns:get_data_from_inetermediate_tableSoapIn"/>
<wsdl:output message="tns:get_data_from_inetermediate_tableSoapOut"/>
</wsdl:operation>
</wsdl:portType>


<! -- <binding> how portType operations will be transported over network. Binding can be
made available in HTTP GET, POST or SOAP. It contains soap:binding, soap:operation and
soapbody- which contains encoding style info.-->

<wsdl:binding name="agriSoap" type="tns:agriSoap">
<soap:binding transport="http://schemas.xmlsoap.org/soap/http"/>
<wsdl:operation name="get_grid_numbers_on_date">
<soap:operation soapAction="http://tempuri.org/get_grid_numbers_on_date"
style="document"/>
<wsdl:input>
<soap:body use="literal"/>
</wsdl:input>
<wsdl:output>
<soap:body use="literal"/>
</wsdl:output>
</wsdl:operation>
<wsdl:operation name="get_grid_numbers_between_date">
<soap:operation soapAction=http://tempuri.org/get_grid_numbers_between_date
style="document"/>
<wsdl:input>
<soap:body use="literal"/>
</wsdl:input>
<wsdl:output>
<soap:body use="literal"/>
</wsdl:output>
</wsdl:operation>
<wsdl:operation name="get_data_from_inetermediate_table">
```

```
<soap:operation soapAction="http://tempuri.org/get_data_from_inetermediate_table"
style="document"/>
<wsdl:input>
<soap:body use="literal"/>
</wsdl:input>
<wsdl:output>
<soap:body use="literal"/>
</wsdl:output>
</wsdl:operation>
</wsdl:binding>


<! -- <service> gives information about where to access the service (Physical endpoint),
through which port and how to communicate the message. -->

<wsdl:service name="agri">

<! -- <port> -->

<wsdl:port name="agriSoap" binding="tns:agriSoap">
<soap:address location="http://164.100.133.131/agri_ksrsac_services/agri.asmx"/>
</wsdl:port>
<wsdl:port name="agriSoap12" binding="tns:agriSoap12">
<soap12: address location="http://164.100.133.131/agri_ksrsac_services/agri.asmx"/>
</wsdl:port>

</wsdl:service>

</wsdl:definitions>
```


2. Sample Response for WMS request


```
<WMS_Capabilities version="1.3.0" xsi:schemaLocation="http://www.opengis.net/wms
http://schemas.opengis.net/wms/1.3.0/capabilities_1_3_0.xsd http://www.esri.com/wms
https://server:Port/GIS/services/Local/School/MapServer/WmsServer?version=1.3.0%26ser
vice=WMS%26request=GetSchemaExtension">

<! -- Service Metadata -->

<Service>
<Name>WMS</Name>
<Title>School Location</Title>
<Abstract>WMS</Abstract>
<KeywordList>
<Keyword></Keyword>
</KeywordList>
<OnlineResource xlink:type="simple"
xlink:href="https://server:Port/GIS/services/Local/School/MapServer/WmsServer?"/>
```

<! -- Contact information -->

<ContactInformation>
<ContactPersonPrimary>
<ContactPerson> Shilpa</ContactPerson>
<ContactOrganization>KSRSAC </ContactOrganization>
</ContactPersonPrimary>
<ContactPosition>Team Lead </ContactPosition>
<ContactAddress>
<AddressType>Email</AddressType>
<Address>shilpa.r@ksrsac.in </Address>
<City>Begaluru</City>
<StateOrProvince>Karnataka</StateOrProvince>
<PostCode>560097</PostCode>
<Country>India</Country>
</ContactAddress>
<ContactVoiceTelephone>+91-80-2972 0557 / 58</ContactVoiceTelephone>
<ContactFacsimileTelephone>+91-80- 2972 0556</ContactFacsimileTelephone>
<ContactElectronicMailAddress>ksrsac_gok@yahoo.co.in</ContactElectronicMailAddress>
</ContactInformation>

<! -- Fees or access constraints imposed -->

<Fees>none</Fees>
<AccessConstraints>none</AccessConstraints>
<MaxWidth>4096</MaxWidth>
<MaxHeight>4096</MaxHeight>
</Service>

<Capability>
<Request>

<! -- GetCapabilities-->

<GetCapabilities>
<Format>application/vnd.ogc.wms_xml</Format>
<Format>text/xml</Format>
<DCPType>
<HTTP>
<Get>
<OnlineResource xlink:type="simple"
xlink:href="https://server:Port/GIS/services/Local/School/MapServer/WmsServer?"/>
</Get>
</HTTP>
</DCPType>
</GetCapabilities>

<! -- GetMap-->

<GetMap>

```
<Format>image/bmp</Format>
<Format>image/jpeg</Format>
<Format>image/tiff</Format>
<Format>image/png</Format>
<Format>image/png8</Format>
<Format>image/png24</Format>
<Format>image/png32</Format>
<Format>image/gif</Format>
<Format>image/svg+xml</Format>
<DCPType>
<HTTP>
<Get>
<!-- The URL here for invoking GetCapabilities using HTTP GET is only a prefix to which a
query string is appended. -->
<OnlineResource xlink:type="simple"
xlink:href="https://server:Port/GIS/services/Local/School/MapServer/WmsServer?"/>
</Get>
</HTTP>
</DCPType>
</GetMap>

<! -- GetFeatureInfo-->

<GetFeatureInfo>
<Format>application/vnd.esri.wms_raw_xml</Format>
<Format>application/vnd.esri.wms_featureinfo_xml</Format>
<Format>application/vnd.ogc.wms_xml</Format>
<Format>application/geojson</Format>
<Format>text/xml</Format>
<Format>text/html</Format>
<Format>text/plain</Format>
<DCPType>
<HTTP>
<Get>
<OnlineResource xlink:type="simple"
xlink:href="https://server:Port/GIS/services/Local/School/MapServer/WmsServer?"/>
</Get>
</HTTP>
</DCPType>
</GetFeatureInfo>

<esri_wms:GetStyles>
<Format>application/vnd.ogc.sld+xml</Format>
<DCPType>
<HTTP>
<Get>
<OnlineResource xlink:type="simple"
xlink:href="https://server:Port/GIS/services/Local/School/MapServer/WmsServer?"/>
</Get>
</HTTP>
</DCPType>
```

```
</esri_wms:GetStyles>
</Request>

<! -- Exception-->

<Exception>
<Format>application/vnd.ogc.se_xml</Format>
<Format>application/vnd.ogc.se_inimage</Format>
<Format>application/vnd.ogc.se_blank</Format>
<Format>text/xml</Format>
<Format>XML</Format>
</Exception>

<Layer>
<Title>Layers</Title>
<! -- Availability of layer in following spatial reference -->
<CRS>CRS:84</CRS>
<CRS>EPSG:4326</CRS>
<CRS>EPSG:3857</CRS>
<CRS>EPSG:102100</CRS>

<EX_GeographicBoundingBox>
<westBoundLongitude>74.083697</westBoundLongitude>
<eastBoundLongitude>78.674250</eastBoundLongitude>
<southBoundLatitude>11.616253</southBoundLatitude>
<northBoundLatitude>18.483493</northBoundLatitude>
</EX_GeographicBoundingBox>
<BoundingBox CRS="CRS:84" minx="74.083697" miny="11.616253" maxx="78.674250"
maxy="18.483493"/>
<BoundingBox CRS="EPSG:4326" minx="11.616253" miny="74.083697" maxx="18.483493"
maxy="78.674250"/>
<BoundingBox CRS="EPSG:3857" minx="8246959.418699" miny="1302066.236906"
maxx="8757977.400318" maxy="2094218.943504"/>
<BoundingBox CRS="EPSG:102100" minx="8246959.418699" miny="1302066.236906"
maxx="8757977.400318" maxy="2094218.943504"/>
<Layer queryable="1">
<Name>0</Name>
<Title>School</Title>
<Abstract></Abstract>
<CRS>CRS:84</CRS>
<CRS>EPSG:4326</CRS>
<CRS>EPSG:3857</CRS> <!-- alias 3857 -->
<CRS>EPSG:102100</CRS>
<EX_GeographicBoundingBox>
<westBoundLongitude>74.083697</westBoundLongitude>
<eastBoundLongitude>78.674250</eastBoundLongitude>
<southBoundLatitude>11.616253</southBoundLatitude>
<northBoundLatitude>18.483493</northBoundLatitude>
</EX_GeographicBoundingBox>
<BoundingBox CRS="CRS:84" minx="74.083697" miny="11.616253" maxx="78.674250"
maxy="18.483493"/>
```

```
<BoundingBox CRS="EPSG:4326" minx="11.616253" miny="74.083697" maxx="18.483493"
maxy="78.674250"/>
<BoundingBox CRS="EPSG:3857" minx="8246959.418699" miny="1302066.236906"
maxx="8757977.400318" maxy="2094218.943504"/>
<BoundingBox CRS="EPSG:102100" minx="8246959.418699" miny="1302066.236906"
maxx="8757977.400318" maxy="2094218.943504"/>
<Style>
<Name>default</Name>
<Title>0</Title>
<LegendURL width="16" height="16">
<Format>image/png</Format>
<OnlineResource
xlink:href="https://server:Port/GIS/services/Local/School/MapServer/WmsServer?request=
GetLegendGraphic%26version=1.3.0%26format=image/png%26layer=0"
xlink:type="simple"/>
</LegendURL>
</Style>
</Layer>
</Layer>
</Capability>
</WMS_Capabilities>
```


3. Sample Response for WMTS request


```
<Capabilities xsi:schemaLocation="http://www.opengis.net/wmts/1.0
http://schemas.opengis.net/wmts/1.0/wmtsGetCapabilities_response.xsd" version="1.0.0">

<! -- Service Identification -->

<ows:ServiceIdentification>
<ows:Title>School Tile Service</ows:Title>
<ows:ServiceType>OGC WMTS</ows:ServiceType>
<ows:ServiceTypeVersion>1.0.0</ows:ServiceTypeVersion>
<ows:Fees>none</ows:Fees>
<ows:AccessConstraints>none</ows:AccessConstraints>
</ows:ServiceIdentification>

<! -- Metadata about organisation operating WMTS server -->

<ows:ServiceProvider>
<ows:ProviderName>KSRSAC<ows:ProviderName/>
<ows:ProviderSite
xlink:href="https://server:Port/GIS/rest/services/School1/MapServer/WMTS/tile/1.0.0">
<ows:ServiceContact>
<ows:IndividualName>Shilpa<ows:IndividualName/>
<ows:PositionName>Team Lead<ows:PositionName/>
<ows:ContactInfo>
<ows:Phone>
```

```
<ows:Voice>+91-80-2972 0557 / 58<ows:Voice/>
<ows:Facsimile>+91-80- 2972 0556<ows:Facsimile/>
</ows:Phone>
<ows:Address>
<ows:DeliveryPoint>KSRSAC<ows:DeliveryPoint/>
<ows:City>Bengaluru<ows:City/>
<ows:AdministrativeArea/>
<ows:PostalCode>560097<ows:PostalCode/>
<ows:Country>India<ows:Country/>
<ows:ElectronicMailAddress>ksrsac_gok@yahoo.co.in<ows:ElectronicMailAddress/>
</ows:Address>
</ows:ContactInfo>
</ows:ServiceContact>
</ows:ServiceProvider>


<! -- Operations Metadata -->

<ows:OperationsMetadata>
<ows:Operation name="GetCapabilities">
<ows:DCP>
<ows:HTTP>
<ows:Get
xlink:href="https://server:Port/GIS/rest/services/School1/MapServer/WMTS/1.0.0/WMTSCa
pabilities.xml">
<ows:Constraint name="GetEncoding">
<ows:AllowedValues>
<ows:Value>RESTful</ows:Value>
</ows:AllowedValues>
</ows:Constraint>
</ows:Get>

<! -- add KVP binding in 10.1 -->
<ows:Get xlink:href="https://server:Port/GIS/rest/services/School1/MapServer/WMTS?">
<ows:Constraint name="GetEncoding">
<ows:AllowedValues>
<ows:Value>KVP</ows:Value>
</ows:AllowedValues>
</ows:Constraint>
</ows:Get>
</ows:HTTP>
</ows:DCP>
</ows:Operation>

<ows:Operation name="GetTile">
<ows:DCP>
<ows:HTTP>
<ows:Get
xlink:href="https://server:Port/GIS/rest/services/School1/MapServer/WMTS/tile/1.0.0/">
<ows:Constraint name="GetEncoding">
<ows:AllowedValues>
```

```
<ows:Value>RESTful</ows:Value>
</ows:AllowedValues>
</ows:Constraint>
</ows:Get>
<ows:Get xlink:href="https://server:Port/GIS/rest/services/School1/MapServer/WMTS?">
<ows:Constraint name="GetEncoding">
<ows:AllowedValues><ows:Value>KVP</ows:Value>
</ows:AllowedValues>
</ows:Constraint>
</ows:Get>
</ows:HTTP>
</ows:DCP>
</ows:Operation>
</ows:OperationsMetadata>

<Contents>

<!--Layer-->

<Layer>
<ows:Title>School</ows:Title>
<ows:Identifier>School</ows:Identifier>
<ows:BoundingBox crs="urn:ogc:def:crs:EPSG::3857">
<ows:LowerCorner>8246959.418698623 1302066.2369062705</ows:LowerCorner>
<ows:UpperCorner>8757977.400317516 2094218.9435044602</ows:UpperCorner>
</ows:BoundingBox><ows:WGS84BoundingBox crs="urn:ogc:def:crs:OGC:2:84">
<ows:LowerCorner>74.08369693330415 11.616252777512281</ows:LowerCorner>
<ows:UpperCorner>78.67424956678578 18.483492956531286</ows:UpperCorner>
</ows:WGS84BoundingBox>
<Style isDefault="true">
<ows:Title>Default Style</ows:Title>
<ows:Identifier>default</ows:Identifier>
</Style><Format>image/png</Format>
<TileMatrixSetLink>
<TileMatrixSet>default028mm</TileMatrixSet>
</TileMatrixSetLink>
<TileMatrixSetLink>
<!--Only show this TileMatrixSet if the tiling scheme is compliant to Google Maps (and that
happens with tile width = 256 px)
-->
<TileMatrixSet>GoogleMapsCompatible</TileMatrixSet>
</TileMatrixSetLink>
<ResourceURL format="image/png" resourceType="tile"
template="https://server:Port/GIS/rest/services/School1/MapServer/WMTS/tile/1.0.0/Scho
ol1/{Style}/{TileMatrixSet}/{TileMatrix}/{TileRow}/{TileCol}.png"/>
</Layer>

<!--TileMatrixSet-->

<TileMatrixSet>
<ows:Title>TileMatrix using 0.28mm</ows:Title>
```

<ows:Abstract>
The tile matrix set that has scale values calculated based on the dpi defined by OGC specification (dpi assumes 0.28mm as the physical distance of a pixel).
</ows:Abstract>
<ows:Identifier>default028mm</ows:Identifier>
<ows:SupportedCRS>urn:ogc:def:crs:EPSG::3857</ows:SupportedCRS>

<TileMatrix>
<ows:Identifier>0</ows:Identifier>
<ScaleDenominator>5.590822640285016E8</ScaleDenominator>
<TopLeftCorner>-2.0037508342787E7 2.0037508342787E7</TopLeftCorner>
<TileWidth>256</TileWidth>
<TileHeight>256</TileHeight>
<MatrixWidth>1</MatrixWidth>
<MatrixHeight>1</MatrixHeight>
</TileMatrix>

<TileMatrix>
<ows:Identifier>1</ows:Identifier>
<ScaleDenominator>2.7954113201425034E8</ScaleDenominator>
<TopLeftCorner>-2.0037508342787E7 2.0037508342787E7</TopLeftCorner>
<TileWidth>256</TileWidth>
<TileHeight>256</TileHeight>
<MatrixWidth>2</MatrixWidth>
<MatrixHeight>1</MatrixHeight>
</TileMatrix>

<TileMatrix>
<ows:Identifier>2</ows:Identifier>
<ScaleDenominator>1.3977056600712562E8</ScaleDenominator>
<TopLeftCorner>-2.0037508342787E7 2.0037508342787E7</TopLeftCorner>
<TileWidth>256</TileWidth>
<TileHeight>256</TileHeight>
<MatrixWidth>3</MatrixWidth>
<MatrixHeight>2</MatrixHeight>
</TileMatrix>

<TileMatrix>
<ows:Identifier>3</ows:Identifier>
<ScaleDenominator>6.988528300356235E7</ScaleDenominator>
<TopLeftCorner>-2.0037508342787E7 2.0037508342787E7</TopLeftCorner>
<TileWidth>256</TileWidth>
<TileHeight>256</TileHeight>
<MatrixWidth>6</MatrixWidth>
<MatrixHeight>4</MatrixHeight>
</TileMatrix>

<TileMatrix>
<ows:Identifier>4</ows:Identifier>
<ScaleDenominator>3.494264150178117E7</ScaleDenominator>
<TopLeftCorner>-2.0037508342787E7 2.0037508342787E7</TopLeftCorner>

```
<TileWidth>256</TileWidth>
<TileHeight>256</TileHeight>
<MatrixWidth>12</MatrixWidth>
<MatrixHeight>8</MatrixHeight>
</TileMatrix>

<TileMatrix>
<ows:Identifier>5</ows:Identifier>
<ScaleDenominator>1.7471320750890587E7</ScaleDenominator>
<TopLeftCorner>-2.0037508342787E7 2.0037508342787E7</TopLeftCorner>
<TileWidth>256</TileWidth>
<TileHeight>256</TileHeight>
<MatrixWidth>23</MatrixWidth>
<MatrixHeight>15</MatrixHeight>
</TileMatrix>

<TileMatrix>
<ows:Identifier>6</ows:Identifier>
<ScaleDenominator>8735660.375445293</ScaleDenominator>
<TopLeftCorner>-2.0037508342787E7 2.0037508342787E7</TopLeftCorner>
<TileWidth>256</TileWidth>
<TileHeight>256</TileHeight>
<MatrixWidth>46</MatrixWidth>
<MatrixHeight>30</MatrixHeight>
</TileMatrix>
</TileMatrixSet>
</Contents>
<ServiceMetadataURL
xlink:href="https://server:Port/GIS/rest/services/School1/MapServer/WMTS/1.0.0/WMTSCapabilities.xml"/>
</Capabilities>
```


4. Sample Response for WFS request


```
<wfs:WFS_Capabilities version="1.1.0" xsi:schemaLocation="http://www.opengis.net/gml
http://schemas.opengis.net/gml/3.1.1/base/gml.xsd http://www.opengis.net/ogc
http://schemas.opengis.net/filter/1.1.0/filter.xsd http://www.opengis.net/ows
http://schemas.opengis.net/ows/1.0.0/owsAll.xsd http://www.opengis.net/wfs
http://schemas.opengis.net/wfs/1.1.0/wfs.xsd">

<! -- Information about WFS service-->

<ows:ServiceIdentification>
<ows:Title>School</ows:Title>
<ows:Abstract> School Location service maintained by KSRSAC <ows:Abstract/>
<ows:Keywords>
<ows:Keyword>School<ows:Keyword/>
</ows:Keywords>
<ows:ServiceType>WFS</ows:ServiceType>
```

```
<ows:ServiceTypeVersion>1.1.0</ows:ServiceTypeVersion>
<ows:Fees>None<ows:Fees/>
<ows:AccessConstraints>None <ows:AccessConstraints/>
</ows:ServiceIdentification>

<! -- Metadata about organisation operating WFS server -->

<ows:ServiceProvider>
<ows:ProviderName>KSRSAC<ows:ProviderName/>
<ows:ServiceContact>
<ows:IndividualName>Shilpa<ows:IndividualName/>
<ows:PositionName>Team Lead<ows:PositionName/>
<ows:ContactInfo>
<ows:Phone>
<ows:Voice>+91-80-2972 0557 / 58<ows:Voice/>
<ows:Facsimile>+91-80- 2972 0556<ows:Facsimile/>
</ows:Phone>
<ows:Address>
<ows:DeliveryPoint>KSRSAC<ows:DeliveryPoint/>
<ows:City>Bengaluru<ows:City/>
<ows:AdministrativeArea/>
<ows:PostalCode>560097<ows:PostalCode/>
<ows:Country>India<ows:Country/>
<ows:ElectronicMailAddress>ksrsac_gok@yahoo.co.in<ows:ElectronicMailAddress/>
</ows:Address>
<ows:OnlineResource
xlink:href="http://192.168.1.6:6080/GIS/services/School1/MapServer/WFSServer"/>
<ows:HoursOfService>24X7<ows:HoursOfService/>
<ows:ContactInstructions>Direct contact to KSRSAC through Visit or E-
mail<ows:ContactInstructions/>
</ows:ContactInfo>
</ows:ServiceContact>
</ows:ServiceProvider>


<ows:OperationsMetadata>

<! -- GetCapabilities-->

<ows:Operation name="GetCapabilities">
<ows:DCP>
<ows:HTTP>
<ows:Get xlink:href="https://server:Port/GIS/services/School1/MapServer/WFSServer?"/>
<ows:Post xlink:href="https://server:Port/GIS/services/School1/MapServer/WFSServer"/>
</ows:HTTP>
</ows:DCP>
<ows:Parameter name="AcceptVersions">
<ows:Value>1.1.0</ows:Value>
<ows:Value>1.0.0</ows:Value>
</ows:Parameter>
<ows:Parameter name="AcceptFormats">
```

```
<ows:Value>text/xml</ows:Value>
</ows:Parameter>
</ows:Operation>

<! -- DescribeFeatureType-->

<ows:Operation name="DescribeFeatureType">
<ows:DCP>
<ows:HTTP>
<ows:Get xlink:href="https://server:Port/GIS/services/School1/MapServer/WFSServer?"/>
<ows:Post xlink:href="https://server:Port/GIS/services/School1/MapServer/WFSServer"/>
</ows:HTTP>
</ows:DCP>
<ows:Parameter name="outputFormat">
<ows:Value>text/xml; subType=gml/3.1.1/profiles/gmlsf/1.0.0/0</ows:Value>
</ows:Parameter>
</ows:Operation>

<! -- GetFeature-->

<ows:Operation name="GetFeature">
<ows:DCP>
<ows:HTTP>
<ows:Get xlink:href="https://server:Port/GIS/services/School1/MapServer/WFSServer?"/>
<ows:Post xlink:href="https://server:Port/GIS/services/School1/MapServer/WFSServer"/>
</ows:HTTP>
</ows:DCP>
<ows:Parameter name="resultType">
<ows:Value>results</ows:Value>
<ows:Value>hits</ows:Value>
</ows:Parameter>
<ows:Parameter name="outputFormat">
<ows:Value>text/xml; subType=gml/3.1.1/profiles/gmlsf/1.0.0/0</ows:Value>
</ows:Parameter>
</ows:Operation>

<! -- ExtendedCapabilities-->

<ows:ExtendedCapabilities>
<ows:Constraint name="serviceAxisOrderForSwappableSRS">
<ows:Value>latitude,longitude</ows:Value>
</ows:Constraint>
</ows:ExtendedCapabilities>
</ows:OperationsMetadata>

<!--Defines the list of feature types and operations that are available from a web feature
service and SRS details-->

<wfs:FeatureTypeList>
<wfs:FeatureType>
<wfs:Name>School1:School</wfs:Name>
```

```
<wfs:Title>School Locations</wfs:Title>
<wfs:DefaultSRS>urn:ogc:def:crs:EPSG:6.9:32643</wfs:DefaultSRS>
<wfs:OtherSRS>urn:ogc:def:crs:EPSG:6.9:4326</wfs:OtherSRS>
<wfs:OutputFormats>
<wfs:Format>text/xml; subType=gml/3.1.1/profiles/gmlsf/1.0.0/0</wfs:Format>
</wfs:OutputFormats>
<ows:WGS84BoundingBox>
<ows:LowerCorner>74.083696933304168 11.616252777512292</ows:LowerCorner>
<ows:UpperCorner>78.674249566785775 18.483492956531293</ows:UpperCorner>
</ows:WGS84BoundingBox>
</wfs:FeatureType>
</wfs:FeatureTypeList>

<!-- Filter Encoding specification-->

<ogc:Filter_Capabilities>

<! -- Spatial Capability-->

<ogc:Spatial_Capabilities>
<ogc:GeometryOperands>
<ogc:GeometryOperand>gml:Envelope</ogc:GeometryOperand>
<ogc:GeometryOperand>gml:Point</ogc:GeometryOperand>
<ogc:GeometryOperand>gml:Polygon</ogc:GeometryOperand>
<ogc:GeometryOperand>gml:LineString</ogc:GeometryOperand>
</ogc:GeometryOperands>
<ogc:SpatialOperators>
<ogc:SpatialOperator name="BBOX"/>
<ogc:SpatialOperator name="Equals"/>
<ogc:SpatialOperator name="Disjoint"/>
<ogc:SpatialOperator name="Intersects"/>
<ogc:SpatialOperator name="Crosses"/>
<ogc:SpatialOperator name="Touches"/>
<ogc:SpatialOperator name="Within"/>
<ogc:SpatialOperator name="Contains"/>
<ogc:SpatialOperator name="Overlaps"/>
</ogc:SpatialOperators>
</ogc:Spatial_Capabilities>

<! -- Scalar Capability-->

<ogc:Scalar_Capabilities>
<ogc:LogicalOperators/>
<ogc:ComparisonOperators>
<ogc:ComparisonOperator>EqualTo</ogc:ComparisonOperator>
<ogc:ComparisonOperator>NotEqualTo</ogc:ComparisonOperator>
<ogc:ComparisonOperator>LessThan</ogc:ComparisonOperator>
<ogc:ComparisonOperator>GreaterThan</ogc:ComparisonOperator>
<ogc:ComparisonOperator>LessThanEqualTo</ogc:ComparisonOperator>
<ogc:ComparisonOperator>GreaterThanEqualTo</ogc:ComparisonOperator>
<ogc:ComparisonOperator>Like</ogc:ComparisonOperator>
```

```
<ogc:ComparisonOperator>Between</ogc:ComparisonOperator>
<ogc:ComparisonOperator>NullCheck</ogc:ComparisonOperator>
</ogc:ComparisonOperators>
</ogc:Scalar_Capabilities>

<ogc:Id_Capabilities>
<ogc:EID/>
<ogc:FID/>
</ogc:Id_Capabilities>
</ogc:Filter_Capabilities>
</wfs:WFS_Capabilities>
```


5. Sample Response for WPS request


```
<wps:Capabilities version="1.0.0" service="WPS" xml:lang="en-US" updateSequence="1"
xsi:schemaLocation="http://www.opengis.net/wps/1.0.0
http://schemas.opengis.net/wps/1.0.0/wpsGetCapabilities_response.xsd">

<! -- Service Identification -->

<ows:ServiceIdentification>
<ows:Abstract/>
<ows:Keywords>
<ows:Keyword>Buffer</ows:Keyword>
</ows:Keywords>
<ows:ServiceType>WPS</ows:ServiceType>
<ows:ServiceTypeVersion>1.0.0</ows:ServiceTypeVersion>
<ows:Fees>None<ows:Fees/>
<ows:AccessConstraints>None</ows:AccessConstraints>
</ows:ServiceIdentification>

<! -- Metadata about organisation operating WPS server -->

<ows:ServiceProvider>
<ows:ProviderName>KSRSAC</ows:ProviderName>
<ows:ServiceContact>
<ows:IndividualName>Shilpa<ows:IndividualName/>
<ows:PositionName>Team Lead<ows:PositionName/>
<ows:ContactInfo>
<ows:Phone>+91-80-2972 0557 / 58<ows:Phone/>
<ows:Address>KSRSAC<ows:Address/>
<ows:OnlineResource
xlink:href="https:server:Port/GIS/services/Buffer/GPServer/WPSServer"/>
<ows:HoursOfService>24X7</ows:HoursOfService>
<ows:ContactInstructions>Direct contact to KSRSAC through Visit or E-
mail<ows:ContactInstructions/>
</ows:ContactInfo>
</ows:ServiceContact>
</ows:ServiceProvider>
```

```
<ows:OperationsMetadata>

<! -- General descriptions of each of the processes offered by a WPS instance-->

<ows:Operation name="GetCapabilities">
<ows:DCP>
<ows:HTTP>
<ows:Get xlink:href="https://server:Port/GIS/services/Buffer/GPServer/WPSServer?"/>
<ows:Post xlink:href="https://server:Port/GIS/services/Buffer/GPServer/WPSServer"/>
</ows:HTTP>
</ows:DCP>
</ows:Operation>

<! -- allows a client to request and receive back detailed information about the processes
that can be run on the service instance-->

<ows:Operation name="DescribeProcess">
<ows:DCP>
<ows:HTTP>
<ows:Get xlink:href="https://server:Port/GIS/services/Buffer/GPServer/WPSServer?"/>
<ows:Post xlink:href="https://server:Port/GIS/services/Buffer/GPServer/WPSServer"/>
</ows:HTTP>
</ows:DCP>
</ows:Operation>

<! -- allows a client to run a specified process implemented by WPS-->

<ows:Operation name="Execute">
<ows:DCP>
<ows:HTTP>
<ows:Get xlink:href="https://server:Port/GIS/services/Buffer/GPServer/WPSServer?"/>
<ows:Post
xlink:href="https://server:Port/GIS/services/Buffer/GPServer/WPSServer"/></ows:HTTP>
</ows:DCP>
</ows:Operation>

</ows:OperationsMetadata>

<! -- brief description of each of the processes offered by the service -->

<wps:ProcessOfferings>
<wps:Process wps:processVersion="1">
<ows:Identifier>Buffer</ows:Identifier>
<ows:Title>Buffer</ows:Title>
<ows:Abstract/>
</wps:Process>
</wps:ProcessOfferings>

<! -- list of the default and optional languages offered by the service -->

<wps:Languages>
```

```
<wps:Default>
<ows:Language>en-US</ows:Language>
</wps:Default>
<wps:Supported>
<ows:Language>en-US</ows:Language>
</wps:Supported>
</wps:Languages>
</wps:Capabilities>
```

6. Sample web services developed between KGIS and KIADB team for sharing the Plot details and priority definition

```
**********************************************************************
Title : PLOT Priority
URL : 115.124.123.47:8080/kiadb_services/webapi/kumresource/kiadbGIS
Method : POST
Content type :application/XML
URL Params :  none
Data Params :
<priorityGisBean>
        <uniqueId> [Long] </uniqueId>
        <idenFlag> [String] </idenFlag>
        <plotPriorityBean>
                <plotPriority> [int] </plotPriority>
                <districtName> [String] </districtName>
                <industryName> [String]</industryName>
                <plotNo> [String] </plotNo>
        </plotPriorityBean>
        <plotPriorityBean>
                <plotPriority>[int]</plotPriority>
                <districtName>[String] </districtName>
                <industryName> [String] </industryName>
                <plotNo> [String] </plotNo>
        </plotPriorityBean>
</priorityGisBean>

**********************************************************************
Example:
**********************************************************************
<plotPriorityBeans>
        <uniqueId>12345</uniqueId>
        <idenFlag>KIADB</idenFlag>
        <plotPriorityBean>
                <plotPriority>1</plotPriority>
```

```
            <districtName>Bangalore</districtName>
            <industryName>Bangalore Limited</industryName>
            <plotNo>ABC1,BCD2</plotNo>
        </plotPriorityBean>
        <plotPriorityBean>
            <plotPriority>2</plotPriority>
            <districtName>Mysore</districtName>
            <industryName>Mysore Limited</industryName>
            <plotNo>ABC11,BCD22</plotNo>
        </plotPriorityBean>
    </plotPriorityBeans>
```


7. Sample Web service published by NIC to KSRSAC for sharing the details of ration shop

Food dept, intended to provide access to KSRSAC, for collecting the coordinates of fair price shops across the state along with photograph.

Following web service is intended to pull and push the data from KSRSAC.

URL: http://xx.xx.xx.xx/fcsdataoth_ws/WebService.asmx
Method Name: **shopList_ds**.

Description: The mentioned service will return the the list of shops based on the District code,taluk code and Food Inspectror mobile number.
INPUT PARAMETER

- username
- password
- distcode
- talcode
- fi_mob_no

NOTE:THE USERNAME AND PASSWORD REQUIRED TO ACCESS THE SERVICE WILL BE PROVIDED BY THE FCS DEPARTMENT.
OUTPUT PARAMETER

- DISTRICT_CODE
- SHOP_NAME_IN_ENGLISH
- SHOP_NAME_IN_KAN
- SHOP_AREA
- SHOP_ADDRE
- OWNER_NAME_ENG
- OWNER_NAME_KAN
- OWNER_MOBILE_NO

- TALUK_CODE
- FPD_ID
- FPD_SHOP_NO

SAMPLE OUTPUT:
THE OUTPUT OF THE ABOVE METHOD WILL BE A DATSET.

- <SHOPLIST msdata:rowOrder="0">
<DISTRICT_CODE>1501</DISTRICT_CODE>
<SHOP_NAME_IN_ENGLISH>V.S.S. KERUR</SHOP_NAME_IN_ENGLISH>
<SHOP_NAME_IN_KAN>ವಿ.ಎಸ್ .ಎಸ್ .ಕೆರೂರ</SHOP_NAME_IN_KAN>
<SHOP_AREA>U</SHOP_AREA>
<SHOP_ADDRESS>ವಾರ್ಡ್ ನಂ 2&3 ಬಸ್ ಸಾಟಂರ್ಡ ಹತ್ತಿರ ಕೆರೂರ.</SHOP_ADDRESS>
<OWNER_NAME_ENG>V.S.S. KERUR</OWNER_NAME_ENG>
<OWNER_NAME_KAN>ವಿ.ಎಸ್ .ಎಸ್ .ಕೆರೂರ</OWNER_NAME_KAN>
<OWNER_MOBILE_NO>9972301729</OWNER_MOBILE_NO>
<TALUK_CODE>1501001</TALUK_CODE>
<FPD_ID>4</FPD_ID>
<FPD_SHOP_NO>4</FPD_SHOP_NO>
</SHOPLIST>

URL: http://xxx.xx.xx.xx/fcsdataoth_ws/WebService.asmx
Method Name: **insertShopCor**.
Description: This service will store the the gps coordinate details and the photo sent by the FI.
INPUT PARAMETER

- username
- password
- dist_code
- tal_code
- shop_id
- shop_no
- shop_lat (latitude)
- shop_lon (longitude)
- photo
- fi_mob_no

ON SUCCESSFUL OPERATION THE WEB METHOD WILL RETURN “0”.
SAMPLE DATA
LATITUDE : 16.07250
LONGITUDE : 75.64133
SAMPLE PHOTO :

0xFFD8FFE000104A46494600010100000100010000FFDB004300100B0C0E0C0A100E0D0E12111
01318281A181616183123251D283A333D3C3933383740485C4E404457453738506D51575F626
768673E4D71797064785C656763FFDB00430111112121815182F1A1A2F63423842636363636363
636363636363636363636363636363636363636363636363636363636363636363636363636A8
A9AAB2B3B4B5B6B7B8B9BAC2C3C4C5C6C7C8C9CAD2D3D4D5D6D7D8D9DAE1E2E3E4

8. Sample List of Unique codes for various assets

The field names of MIS database having the unique code of department MIS database is given below. In KGIS database, the field names "KGIS ID" and "KGIS Code" shall be constant / standardized. One of these key unique fields shall be used for linking between spatial and non-spatial data.

S.No	Category	SubCategory	LayerName	Alias	Field Name linked to dept MIS database having unique code
1.	Education	Higher Education	University	University	UniversityCode
2.	Education	Higher Education	StateColleges	StateColleges	CollegeCode
3.	Education	Higher Education	PolytechnicInstitute	PolytechnicInstitute	InstitutionCode
4.	Education	Higher Education	OtherColleges	OtherColleges	CollegeCode
5.	Education	Higher Education	JuniorTechnicalSchool	JuniorTechnicalSchool	InstituteCode
6.	Education	Higher Education	EngineeringCollege	EngineeringCollege	CollegeCode
7.	Education	Higher Education	PreUniversityCollege	PreUniversityCollege	NICODE
8.	Education	Primary Education	PrimarySchool	PrimarySchool	SCHCD
9.	Education	Technical Education	ITI	ITI	
10.	Education	Medical Education	AyushColleges	AyushColleges	Code
11.	Education	Medical Education	AyushHospitalDispensaries	AyushHospitalDispensaries	Code
12.	Education	Medical Education	DentalCollege	DentalCollege	
13.	Education	Medical Education	MedicalCollege	MedicalCollege	
14.	Education	Medical Education	NursingSchool	NursingSchool	SchoolCode
15.	Education	Medical Education	ParamedicalCollege	ParamedicalCollege	Code
16.	Health	TraumaCentre	TraumaCentre	TraumaCentre	
17.	Health	PrivateHospital	PrivateHospital	PrivateHospital	Code
18.	Health	GovtHospital	GovtHospital	GovtHospital	Facility Code
19.	Health	DiagnosticCentersGeo	DiagnosticCentersGeo	DiagnosticCentersGeo	
20.	Health	ArogyaKavachaAmbulance	ArogyaKavachaAmbulance	ArogyaKavachaAmbulance	
21.	Health	StateAmbulance	StateAmbulance	StateAmbulance	Code

S.No	Category	SubCategory	LayerName	Alias	Field Name linked to dept MIS database having unique code
22.	Health	NagumaguAmbulance	NagumaguAmbulance	NagumaguAmbulance	Code
23.	Health	BloodBank	BloodBank	BloodBank	
24.	Health	ScanningCenter	ScanningCenter	ScanningCenter	
25.	Health	SpecialityHospital	SpecialityHospital	SpecialityHospital	
26.	Health	Storage	Storage	Storage	
27.	Mines, Industry & Allied	CheckPost	CheckPost	CheckPost	Code
28.	Mines, Industry & Allied	ChemicalLaboratory	ChemicalLaboratory	ChemicalLaboratory	Id
29.	Mines, Industry & Allied	MineralLocation	MineralLocation	MineralLocation	
30.	Mines, Industry & Allied	GWFObservationWell	GWFObservationWell	GWFObservationWell	
31.	Mines, Industry & Allied	Godown	Godown	Godown	Id
32.	Mines, Industry & Allied	IndustrialArea	IndustrialArea	IndustrialArea	Qid
33.	Social Welfare	BackwardClasses	BCResidentialSchool	BCResidentialSchool	ResidentialSchoolCode
34.	Social Welfare	BackwardClasses	BCHostel	BCHostel	HostelCode