Final Report of CSIT 6910D

Pervasive Positioning Standard for Fingerprint-based and Proximity-based Systems

Yuan Fangxu, 20799126, fyuanad@connect.ust.hk

1. Abstract

Pervasive positioning is to locate an object anywhere seamlessly on a country scale. Current positioning technologies are mature enough to support both indoor and outdoor environments, using site signal survey and indoor localization algorithms for indoor and GNSS for outdoor. Pervasive positioning can be realized if application request's location services from the correct parties using the correct formats. Pervasive Positioning Standard aims to bridge every party together. This standard specifies a set of communication protocols, data organization and format definitions for site signals, and data organization for maps using existing map standards. With this standard, existing location-based service (LBS) applications can operate anywhere in Hong Kong including their original supporting zone, and many novel LBS applications can be developed.

2. Introduction

The goal of the Pervasive Positioning Standard is to enable any applications to locate their users anywhere to provide LBS potentially on a country scale. It is important to be able to locate users in both outdoor and indoor environments. In outdoor open areas, localization can be done with GNSS using satellite signals. In outdoor and indoor enclosed environments, we carry out site surveys and use the site signals to locate users. However, there are no agreed formats of site signals, so applications cannot share their site signals with others easily to enlarge their supported zone. In addition, pervasive positioning on a country scale requires the management of a huge size of site signal and map data. This standard aims to provide communication protocols, data formats, and organization required for an application to locate a user anywhere efficiently.

This standard consists of three components:
1) A set of communication protocols,
2) data organization and format definitions for site signals,
3) data organization for maps using existing map standards.

3. Overview

Communication protocols describe the necessary components for applications to acquire locations, that is, what should be done for all stakeholders. The following diagram shows a general workflow of the protocols:

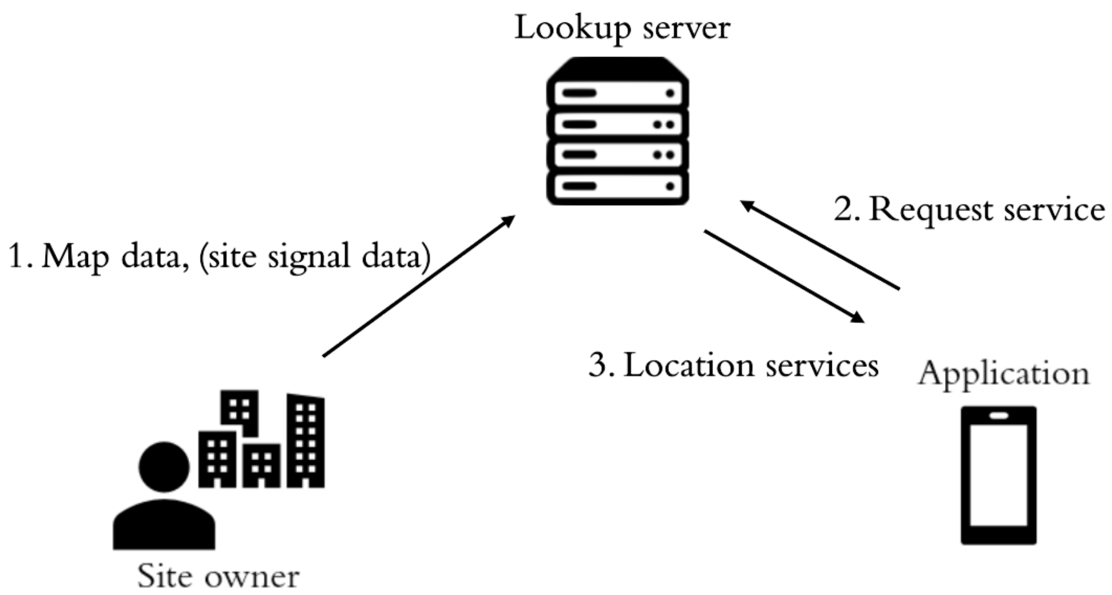

Figure 1. A general workflow of communication protocols

Site owners first upload map data and/or site signal data to lookup server. When applications request locations, they contact the lookup server, and the lookup server offers the location service to them, for example, sending nearby site signals to them or redirecting them to site owner’s server.

When Site owner sends data to Lookup Server, it is very important to verify the data and make sure it meets the requirements. I am mainly responsible for building the data validation part of the platform.

4 Architecture

To enable pervasive positioning, the platform requires site owners to upload a data package including site information, spatial representation, maps, site signals and metadata for the grid. Contents in the data package should be arranged according to the standard specification, including file structure, file naming principle, file contents format and its values.

4.1 Data package structure

Since indoor positioning data and outdoor positioning data use different positioning methods and techniques, we designed the format of indoor data and outdoor data separately.

4.1.1 Indoor site data

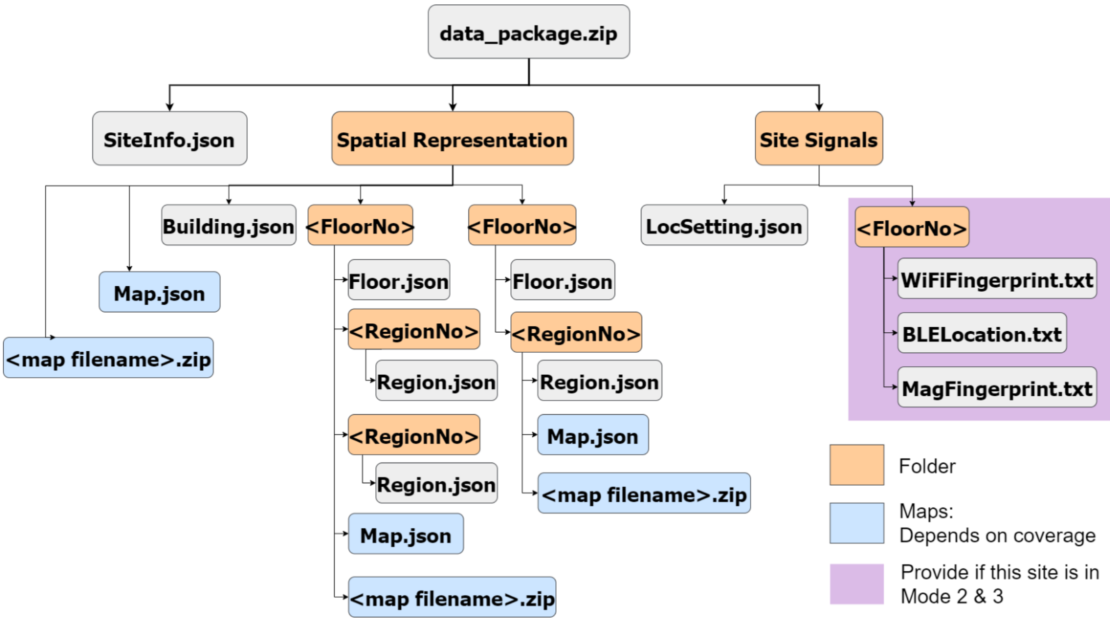

Figure 2. The file system tree for indoor data validation.

All blocks are named by their filename or folder name. Blocks colored with orange are folders in data package, they named by categories or tags (FloorNo or RegionNo). Blocks colored with blue are zipped map files and metadata of map files, they should be arranged according to map coverage. Blocks colored with purple are site signals file of this building, site owner who wants to share site signal to platform should include this part.

4.1.2 Outdoor site data

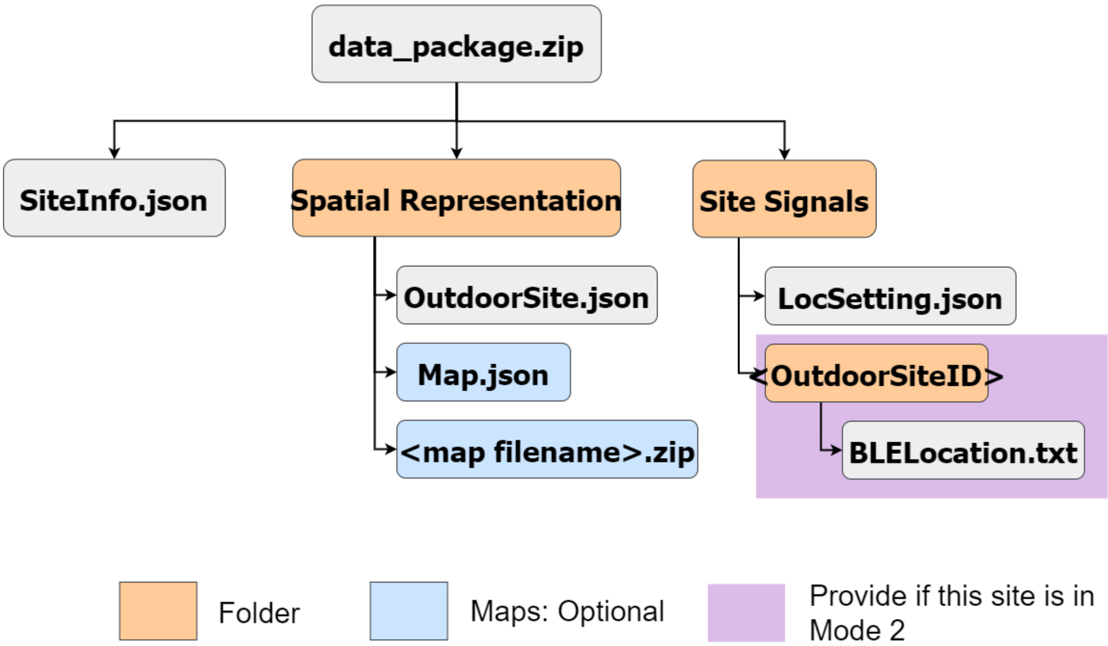

Figure 3. The file system tree for outdoor data validation

Outdoor data package contains less information, as outdoor space is less complex. It contains SiteInfo.json, OutdoorSite.json, OutdoorLocSetting.json and some optional data: map.json, map file and outdoor site signals.

Optional map in outdoor data package allow site owner provides customize outdoor map instead of global map on the internet. The map file should be zipped.

4 Methodology

In this Independent project, I provide two ways to verify data, using the AWS tool Glue and scripting for verification, each of which has its own advantages. The advantage of Glue is that it automates the process and is inexpensive to deploy on Amazon Web Service in a low-code manner. The advantage of validation by script is that it does not take much time to deploy and can be executed by a simple compilation platform.

4.1 Method based on Amazon Web Service (AWS) tools

AWS Glue is a fully managed ETL (extract, transform, and load) service that makes it simple and cost-effective to categorize the data, clean it, enrich it, and move it reliably between various data stores and data streams. AWS Glue consists of a central metadata repository known as the AWS Glue Data Catalog, an ETL engine that automatically generates Python or Scala code, and a flexible scheduler that handles dependency resolution, job monitoring, and retries. AWS Glue is serverless, so there's no infrastructure to set up or manage.

In the method, there are three step to validate data from site owners:
1. Data Load
2. Data Crawler
3. Data Transform

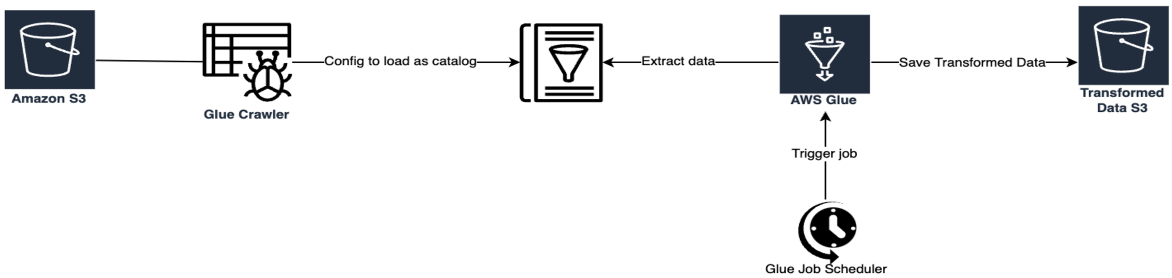

Figure 4. The workflow of data validation using AWS tools

Details will be elaborated in the following.

4.1.1 Data load

In order to be able to process data in AWS services, Site owners first need to place the data package in data storage provided by AWS, in this project we use Simple Storage Service (S3) as our choice, because S3 supports multiple data types, is easy to store, and works well with our data processing tool Glue.

4.1.2 Data crawler

Data Crawler is one of the most important components of Glue, through Data Crawler we can get the schema of the data stored in S3 bucket, which is a kind of meta data, it is very helpful for us to confirm the format and content of data in the data package.

4.1.3 Data transform

The second Glue application is the Glue studio, where we can clean and transform data, such as dropping null values and converting string data to date data, etc.

After processing the data from site owners to meet the corresponding requirements, it is finally placed in the result S3 bucket for use by the Lookup server.

4.1.4 Result

Name	Database	Location	Classification	Last updated	Deprecated
data_package_80_8f848882803a3d694391ddd16725be4b	gluetest	s3://fangxuy-s3/data_package/Spatial R...	json	10 May 2022 10:19 AM UTC+8	Yes
data_package_site_signal	gluetest	s3://fangxuy-s3/data_package/Site Signal/	ion	10 May 2022 10:19 AM UTC+8	Yes
data_package_80_23cc265f855ee7d6689d2b92f4ee9d80	gluetest	s3://fangxuy-s3/CYT_indoor_2.0/Spatial ...	json	10 May 2022 10:19 AM UTC+8	
data_package_80_6be3e1745b512cd07d32dfc42b30eb0f	gluetest	s3://fangxuy-s3/CYT_indoor_2.0/Spatial ...	json	10 May 2022 10:19 AM UTC+8	
data_package_map_json_a690e962d23b63fc90b849a145dafa1c	gluetest	s3://fangxuy-s3/data_package/Spatial R...	json	10 May 2022 10:19 AM UTC+8	Yes
data_package_map_json_31377665ad2b37088310c928660e42...	gluetest	s3://fangxuy-s3/data_package/Spatial R...	json	10 May 2022 10:19 AM UTC+8	Yes
data_package_80_b30f58e1bbc772286a579a7fdeceb627	gluetest	s3://fangxuy-s3/data_package/Spatial R...	json	10 May 2022 10:19 AM UTC+8	Yes
data_package_siteinfo_json_d6f47131b3a9b68febc2687839a2...	gluetest	s3://fangxuy-s3/HKSTP_outdoor_2.0/Sit...	json	10 May 2022 10:19 AM UTC+8	
test_fangxu_vpc_bucket	gluetest	s3://test-fangxu-vpc-bucket/	Unknown	6 April 2022 3:38 AM UTC+8	
data_package_siteinfo_json_392a29ebaf1e5a0851665e93a8aa...	gluetest	s3://fangxuy-s3/CYT_indoor_2.0/SiteInfo...	json	10 May 2022 10:19 AM UTC+8	
data_package_map_jpg_c2a291120a98020916eac15c993ecfd0	gluetest	s3://fangxuy-s3/data_package/Spatial R...	Unknown	10 May 2022 10:19 AM UTC+8	Yes
data_package_floor_json_1e3d35e8705ba9f8f25cf574b647ed30	gluetest	s3://fangxuy-s3/CYT_indoor_2.0/Spatial ...	json	10 May 2022 10:19 AM UTC+8	
data_package_80_69a9804b972a4b8dc30923627174c920	gluetest	s3://fangxuy-s3/CYT_indoor_2.0/Spatial ...	json	10 May 2022 10:19 AM UTC+8	
data_package_map_json_2b9104d55682ecd06cdd5a31ede209...	gluetest	s3://fangxuy-s3/data_package/Spatial R...	json	10 May 2022 10:19 AM UTC+8	Yes
data_package_mapfile_zip_83d4149c3cfb48aafbb0b35c847d0...	gluetest	s3://fangxuy-s3/CYT_indoor_2.0/Spatial ...	Unknown	10 May 2022 10:19 AM UTC+8	
data_package_map_json	gluetest	s3://fangxuy-s3/data_package/Spatial R...	json	10 May 2022 10:19 AM UTC+8	Yes
data_package_map_json_0111be9d1e90b8feac71737b5ee8ceb9	gluetest	s3://fangxuy-s3/data_package/Spatial R...	json	10 May 2022 10:19 AM UTC+8	Yes
data_package_80_38478e2cd0c88e6cc3c419f60ce4447c	gluetest	s3://fangxuy-s3/CYT_indoor_2.0/Spatial ...	json	10 May 2022 10:19 AM UTC+8	
data_package_site_signal_954c440f66e9662add46ec33607695...	gluetest	s3://fangxuy-s3/CYT_indoor_2.0/Site Sig...	json	10 May 2022 10:19 AM UTC+8	

Figure 5. The crawler result of Glue

Figure 5 shows the result of crawling the packets uploaded by the Site owner with Glue crawler, and the schema of each file in the packet will be stored in the glue database as a table.

If we look at the table of one of the files, we can see the keys, values and value format of the file in real time, Figure 6 shows the schema table of one of the files

Name	data_package_80_8f848882803a3d694391ddd16725be4b
Description	
Database	gluetest
Classification	json
Location	s3://fangxuy-s3/data_package/Spatial Representation/84/80/
Connection	
Deprecated	Yes
Last updated	Tue May 10 22:19:59 GMT+800 2022
Input format	org.apache.hadoop.mapred.TextInputFormat
Output format	org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat
Serde serialization lib	org.openx.data.jsonserde.JsonSerDe
Serde parameters	paths **ConnectedRegions,Name,ParentID,RegionNo**
Table properties	sizeKey **4690** objectCount **1** UPDATED_BY_CRAWLER **hkust** DEPRECATED_BY_CRAWLER **1652192399671** CrawlerSchemaSerializerVersion **1.0** recordCount **1** averageRecordSize **4690** CrawlerSchemaDeserializerVersion **1.0** compressionType **none** typeOfData **file**

Schema

Showing: 1 - 4 of 4

	Column name	Data type	Partition key	Comment
1	regionno	string		
2	name	string		
3	parentid	string		
4	connectedregions	array		

Figure 6. The schema table of a file

4.2 Method based on python script

If we don't use AWS tools, the best way to validate data is to use script to iterate through each file in data package and determine whether the format of the key and value in the file is compliant, and we can use the corresponding function to determine whether the value is within the specified range, for example, we can use simple logic to determine whether the value of whether the latitude and longitude in the file is within a reasonable range (Hong Kong latitude and longitude)

4.2.1 Result

We used a python script to scan both the indoor and outdoor datasets provided, and if there was an error in one of the files, it would report the error directly in the returned results, and if there was no error, it would show that the scan passed the file. Figure 7 and Figure 8 show the scan results for indoor and outdoor data package respectively.

```
Scanning: CYT_indoor_2.0/SiteInfo.json ...
CYT_indoor_2.0/SiteInfo.json Pass
Scanning: CYT_indoor_2.0/Spatial Representation/Building.json ...
CYT_indoor_2.0/Spatial Representation/Building.json Pass
Scanning: CYT_indoor_2.0/Spatial Representation/87/Floor.json ...
CYT_indoor_2.0/Spatial Representation/87/Floor.json Pass
Scanning: CYT_indoor_2.0/Spatial Representation/87/Map.json ...
CYT_indoor_2.0/Spatial Representation/87/Map.json Pass
Scanning: CYT_indoor_2.0/Spatial Representation/87/80/Region.json ...
CYT_indoor_2.0/Spatial Representation/87/80/Region.json Pass
Scanning: CYT_indoor_2.0/Spatial Representation/80/Floor.json ...
CYT_indoor_2.0/Spatial Representation/80/Floor.json Pass
Scanning: CYT_indoor_2.0/Spatial Representation/80/Map.json ...
CYT_indoor_2.0/Spatial Representation/80/Map.json Pass
Scanning: CYT_indoor_2.0/Spatial Representation/80/80/Region.json ...
CYT_indoor_2.0/Spatial Representation/80/80/Region.json Pass
Scanning: CYT_indoor_2.0/Spatial Representation/81/Floor.json ...
CYT_indoor_2.0/Spatial Representation/81/Floor.json Pass
Scanning: CYT_indoor_2.0/Spatial Representation/81/Map.json ...
CYT_indoor_2.0/Spatial Representation/81/Map.json Pass
Scanning: CYT_indoor_2.0/Spatial Representation/81/80/Region.json ...
CYT_indoor_2.0/Spatial Representation/81/80/Region.json Pass
Scanning: CYT_indoor_2.0/Spatial Representation/86/Floor.json ...
CYT_indoor_2.0/Spatial Representation/86/Floor.json Pass
Scanning: CYT_indoor_2.0/Spatial Representation/86/Map.json ...
CYT_indoor_2.0/Spatial Representation/86/Map.json Pass
Scanning: CYT_indoor_2.0/Spatial Representation/86/80/Region.json ...
CYT_indoor_2.0/Spatial Representation/86/80/Region.json Pass
Scanning: CYT_indoor_2.0/Spatial Representation/88/Floor.json ...
CYT_indoor_2.0/Spatial Representation/88/Floor.json Pass
Scanning: CYT_indoor_2.0/Spatial Representation/88/Map.json ...
CYT_indoor_2.0/Spatial Representation/88/Map.json Pass
Scanning: CYT_indoor_2.0/Spatial Representation/88/80/Region.json ...
CYT_indoor_2.0/Spatial Representation/88/80/Region.json Pass
Scanning: CYT_indoor_2.0/Spatial Representation/83/Floor.json ...
CYT_indoor_2.0/Spatial Representation/83/Floor.json Pass
Scanning: CYT_indoor_2.0/Spatial Representation/83/Map.json ...
CYT_indoor_2.0/Spatial Representation/83/Map.json Pass
Scanning: CYT_indoor_2.0/Spatial Representation/83/80/Region.json ...
CYT_indoor_2.0/Spatial Representation/83/80/Region.json Pass
Scanning: CYT_indoor_2.0/Spatial Representation/84/Floor.json ...
CYT_indoor_2.0/Spatial Representation/84/Floor.json Pass
```

Figure 7. The scanning result of indoor data

```
/Users/fangxuy/.conda/envs/pyfile/bin/python /Users/fangxuy/Desktop/pyfile/ip.py
Scanning: HKSTP_outdoor_2.0/SiteInfo.json ...
HKSTP_outdoor_2.0/SiteInfo.json Pass
Scanning: HKSTP_outdoor_2.0/Spatial Representation/OutdoorSite.json ...
HKSTP_outdoor_2.0/Spatial Representation/OutdoorSite.json Pass
Scanning: HKSTP_outdoor_2.0/Site Signal/LocSetting.json ...
HKSTP_outdoor_2.0/Site Signal/LocSetting.json Pass

Process finished with exit code 0
```

Figure 8. The scanning result of outdoor data

5 Future work

In my future work, I will intend to automate the data validation based on these two methods. There are different situations when site owners update their data to our system, the data may be streaming or batch data, and for our data validation system, the logic of handling the data will change, so it will require more effort to adapt to different data scenarios. In addition, integration testing of the Pervasive Positioning Standard for Fingerprint-based and Proximity-based Systems is in process, and I need to continue to integrate the data validation portion of the work with the rest of the system.

Acknowledge

I would like to thank Professor Chan for his careful guidance in this Independent Project, where I learned how to understand the work done by others in an efficient way, extract key information, and also learn how to present my work in the right way. His guidance not only gave me a deeper understanding in the field of localization, but also inspired my research. I would also like to thank my partner Peter for his guidance in the work process, he gave me a lot of advice on my implementation work and report writing.