A Hybrid Peer-to-Peer Architecture for Global Geospatial Web Service Discovery

Shawn Chen[1], Steve Liang[2]

1 Geomatics, University of Calgary, hschen@ucalgary.ca
2 Geomatics, University of Calgary, steve.liang@ucalgary.ca

Abstract

For any large-scale distributed system, both communication and data management distill down to the problem of resource discovery. Similarly, the geoweb needs a resource discovery service for users to find the relevant web services that providing data of interest. The Open GIS Consortium (OGC) recommends tackling the question by using Web Catalog Service (CS/W). However, the system provides only local knowledge. In this work we propose and implement a locality-aware peer-to-peer (P2P) based system for global geospatial web service discovery. The system is scalable because it operates on a cooperative model and has no single point of failure. We use spatial hash indexing to preserve spatial locality information while retaining the load balancing properties of the underlay P2P networks. We evaluate our implementation with both simulated and real world data sets. Our experiments show promising potential of the architecture in both performing spatial and keyword query for discovering geoweb services.

Background and Relevance

Efficient methods for geospatial data discovery and exchange still pose a major challenge for scientists. It is not unusual that scientists working on multidisciplinary Earth Science research have to spend more than 50% time and resources on locating and acquiring data and information, pre-processing and assembling them into analysis-ready form (Di & McDonald, 1999). The technology for information extraction and knowledge discovery is accordingly considered far behind the technology for data collection (Di et al., 2008). Open Geospatial Consortium (OGC) recommended the Web Catalog Service (CS/W) to tackle the issue (OGC, 2007). However, existing OGC architectures and implementations have the following issues:

1) **Single points of failure**: CS/W becomes a system performance bottleneck. When a CS/W portal ceases to function, users are not able to search for services in interest even though the services are functioning;
2) ***A priori* knowledge of CS/W locations**: users have to know the location of existing CS/W servers *a priori* to search for OGC web services (OWS).
3) **Difficult to maintain and manage**: data publishers have to find one or multiple CS/W portals to publish their resources. Without proper tools, management and maintenance can be difficult and challenging.

We argue that using a peer-to-peer (P2P) approach can address the above-mentioned problems and enhance the existing CS/W systems. However, building a P2P system for geospatial web service discovery needs to consider the following unique settings:

1) An OGC server (mostly hosted by large organizations, e.g., Natural Resource Canada, NOAA, and NASA) is a *stable peer* in the system that would not join and leave randomly; in most cases these servers are made accessible 24-7.
2) The volume of data served by OGC servers is huge in general while the number of servers is considerably smaller.
3) There are a greater number of dynamic and transient users (compared to the number of servers). In other words, users are *dynamic peers* in the system that join and leave frequently.

P2P systems normally implement an abstract overlay network built on top of the psychical network. Such overlays are for peer indexing and discovery. According to how peers are organized, generally P2P overlay network is categorized into two paradigms:

1) Structured system: In the system peers are organized and optimized by algorithms and specific criteria such as CAN (Ratnasamy et al., 2001), Pastry (Rowstron & Druschel, 2001), Chord (Stoica et al., 2001). As a result, peers are connected with specific topologies and properties. It offers a scalable solution for exact-match queries.
2) Un-structured system: Peer connections are randomly created so that the overlay network is not optimized by any specific algorithm. Such a system (e.g. Gnutella (Doyle et al., 2001)) is generally more appropriate for accommodating highly transient peer populations (Androutsellis-Theotokis & Spinellis, 2004).

In a geospatial web service discovery context, extension and customization are required for existing architectures to enhance the system stability and reliability. Considering the above-described settings (i.e., a mixture of stable and dynamic peers), the unnecessary overhead required to maintain the structured overlay network makes use of a structured design impractical. On the other hand, an unstructured design diminishes the system stability from stable peers.

As a result, we propose a hybrid P2P approach for geospatial web services discovery. The goal is to build a dynamic, scalable, and decentralized OGC CS/W with flexible spatial and keyword querying capabilities. The proposed system is also unique in that it is a locality-aware system, that is the system is able to exploit the locality information between peers in order to deliver the query results quickly and efficiently.

Methods and Data

Figure 1 depicts the core elements of this distributed geospatial service discovery system. A quadkey (i.e., a quadtree key) is a unique geographical identifier adopted in the system. A geographical location (i.e., latitude and longitude) can be converted from two-dimensional coordinates into a one-dimensional string (i.e., quadkey) using Peano Space-Filling Curves a particular level of detail. Instance here refers to the application running on the local machine. Specifically, in the system only the Hybrid P2P instance runs on machine of a dynamic peer while both instances are executed on a stable peer (i.e., an OGC server). The Hybrid P2P instance is composed of two engines:

1) The **communication manager** provides the interface to users and peers (i.e., neighbors in Figure 1). It is responsible for following tasks:

   i. Determining a physical location of the peer (through IP geolocator) and its location quadkey based on Peano SFC.

   ii. Network setup and cooperative discovery.

   iii. Receiving queries from users. It further resolves the queries by using the local query engine. If no local match found, it sends requests to other peers and merges the received responses as response to the original query.

   iv. Receiving queries from peers in the system. Similar to task iii, it first resolves the local query. Then it sends the response to the requester peer if a local match found. Otherwise, it forwards the query to other peers if the query is still alive (i.e., Time To Life (TTL)>0), merges received responses from forwarded peers as the response to the requester peer.

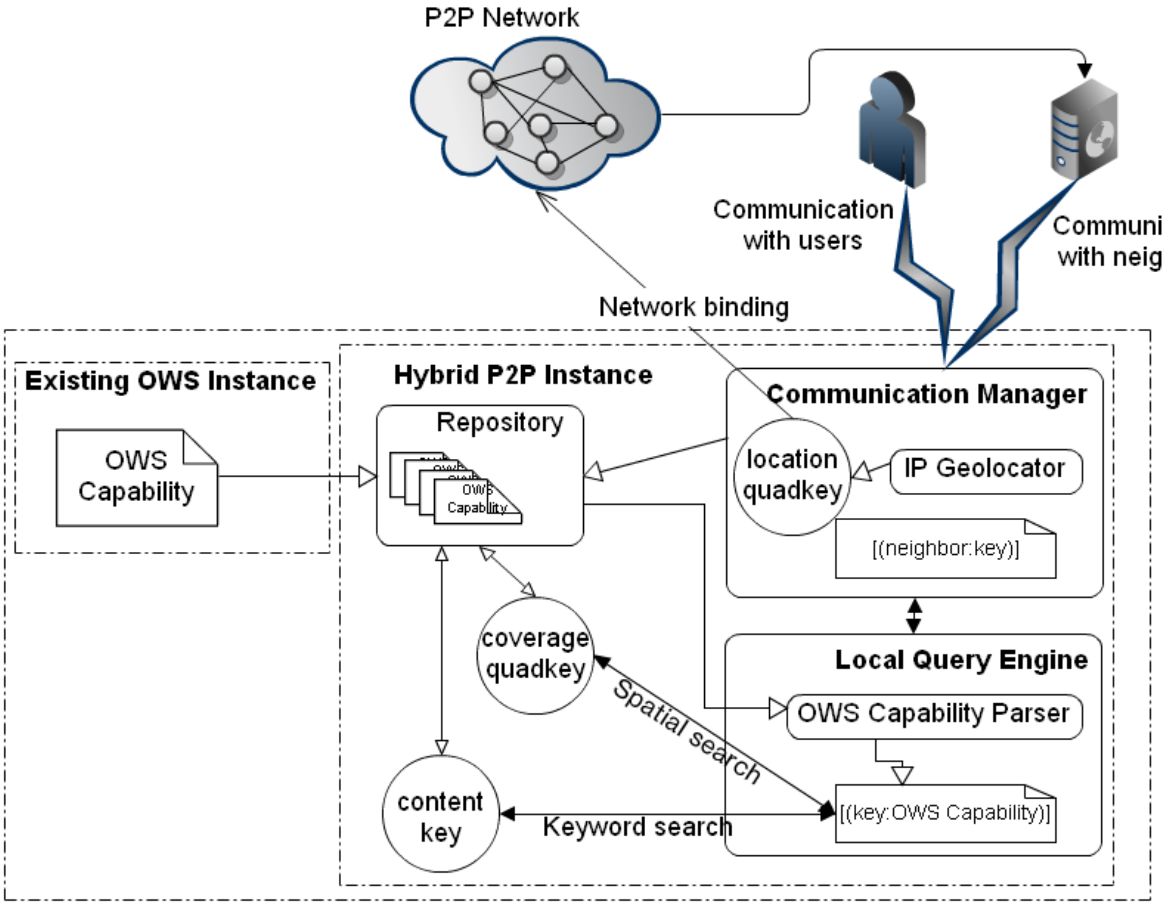

Figure 1 System Architecture (existing OWS is an optional component)

2) The **local query** engine handles following operations:
    i. Spatial indexing OWS capabilities XML document by determining its coverage quadkey.
    ii. Receiving queries from communication manager, searching the local repository for matching services and sending a response to the communication manager.

Results

In this work we present emulation results by using the following settings:

1) Network topology: a Bittorrent swarm replica
2) Search mechanism: *Random Walks* (Lv et al., 2002) and proposed *Locality-Aware Walks*. The former approach is essentially a blind search that a peer forwarding queries randomly chosen neighbors. On the contrary, a locality-aware walks directs queries based on the geographical location that a query is looking for.
3) Query: A Nearby Region-of-Interest (ROI) query is a query looking for services geographically closed to the requester peer. Remote ROI query refers to a query looking for services geographically far from the requester peer.

As shown in Figure 2, *Locality-Aware Walks* design creates higher search hits than *Random Walks* with both types of query. Moreover, in the case that a user querying for dataset in his neighborhood, search hit rate of *Locality-Aware Walks* (0.919 in Table 1) is three times higher than *Random Walks* (0.296 in Table 1).

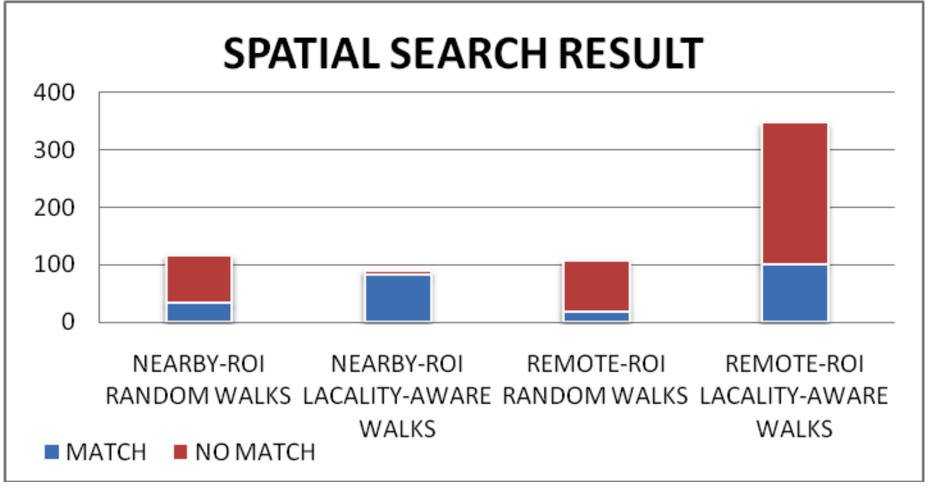

Figure 2 Average Message Productions

From Table 1 we notice that the number of hops is similar for two designs. This happens because, with small network size, a dynamic peer (i.e., a user) is easily neighbored to a stable peer (i.e., an OWS server). One interesting observation is how transmission delay is affected by geophysical distance. *Random Walks* design connects peers regardless of their geographical location and as a result tends to induce larger transmission delay.

Table 1 Statistics of Emulation Result

	Nearby-ROI Random Walks	Nearby-ROI Locality-Aware Walks	Remote-ROI Random Walks	Remote-ROI Locality-Aware Walks
Search Hit	0.296	0.919	0.203	0.290
Maximum Hops	2	3	3	4
Average Hops	0.935	0.739	1.223	1.830
Maximum Transmission Delay	1200	840	1400	1240
Average Transmission Delay	402.533	220.118	632.9577	322.994

Conclusions

We have presented and implemented new hybrid P2P architecture to enhance OGC CS/W for global geospatial web service discovery. The system provides two functions: (1) service discovery and (2) service publishing. That is, if a user decides not to publish to an OGC CS/W server, the service can still be accessible and discoverable with the proposed system.

Our preliminary experiments focus on three metrics, namely accuracy, the number of exchanged messages and the number of the discovered services. *Locality-Aware Walks* produce much higher accuracy results with less network transmission delay than *Random Walks*, but it may incur higher network traffic in extreme cases.

We are currently improving the primitive implementation to emulate a larger scale of network.

References

L. Di and K. McDonald, "Next generation data and information systems for Earth sciences research," in *Proceedings of the first international symposium on digital earth*, 1999, pp. 92–101.

L. Di, A. and W, Yang, and Y. Liu, and Y. Wei, and P. Mehrotra, and C. Hu, and D. Williams, "The development of a geospatial data Grid by integrating OGC Web services with Globus-based Grid technology", *Software Focus,* vol. 20, pp. 1617--1635, 2008.

OGC, OpenGIS Catalogue Service Implementation Specification Version 2.0.2, 2007. Retrieved November 01, 2010, from http://www.opengeospatial.org/standards/is

S. Androutsellis-Theotokis, and D. Spinellis, A survey of peer-to-peer content distribution technologies. *ACM Computing Surveys,* vol. 36 (4), pp. 335-371, 2004.

S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Schenker, "A scalable content-addressable network," in *Proceedings of the 2001 conference on Applications, technologies, architectures, and protocols for computer communications,* 2001, p. 172.

A. Rowstron and P. Druschel, "Pastry: Scalable, decentralized object location, and routing for large-scale peer-to-peer systems," *Lecture Notes in Computer Science,* pp. 329–350, 2001.

I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan, "Chord: A scalable peer-to-peer lookup service for internet applications," in *Proceedings of the 2001 conference on Applications, technologies, architectures, and protocols for computer communications*, 2001, p. 160.

A. Doyle, C. Reed, J. Harrison, and M. Reichardt, "Introduction to OGC web Services," in *White Paper*, 2001.

C. Lv, P. Cao, E. Cohen, K.Li, and S. Shenker. Search and Replication in Unstructured Peer-to-Peer Networks. *ICS*, 2002.