A LOCALITY-AWARE PEER-TO-PEER APPROACH FOR GEOSPATIAL WEB SERVICES DISCOVERY

Shawn Chen[a], Steve Liang[a], Mea Wang[b]

[a] Department of Geomatics Engineering, University of Calgary, 2500 University Dr NW
Calgary, Alberta, T2N1N4 Canada - (steve.liang, hschen)@ucalgary.ca

[b] Department of Computer Science, University of Calgary, 2500 University Dr NW
Calgary, Alberta, T2N1N4 Canada - (meawang)@ucalgary.ca

Canada's Success with Spatial Data Infrastructure

KEY WORDS: Peer to Peer Network, Web Service Discovery

ABSTRACT:

Discovering appropriate services with related geospatial datasets among a large number of available ones is a key task in the Geographic Information Science domain. This paper proposes a locality-aware peer-to-peer (P2P) based approach for geospatial web services discovery. We present a locality-aware hybrid P2P system for geospatial web service discovery, based on spatial hash index to preserve spatial locality information while retaining local balancing properties of underlay networks.

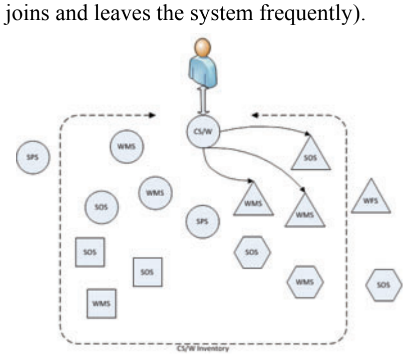
1. INTRODUCTION

While the rest of the world sharing data using state-of-art technologies such as peer-to-peer (P2P) distributed network, scientists are still struggling to find an efficient method for geospatial data discovery and exchange. 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 [1, 2]. The capability of information extraction and knowledge discovery is accordingly considered far behind the capability of data collection. Open Geospatial Consortium (OGC) [3] had proposed Web Catalog Service (CS/W) to solve the issue. However, three critical problems emerged under current circumstances:

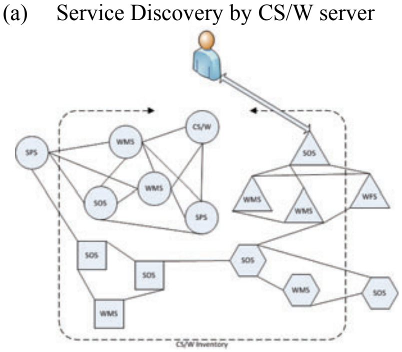
1) Single points of failure: if a CS/W portal out of function, users are not able to search for services in interest even though the services are functioning;
2) Prerequisite for usage: users have to locate the proper CS/W portal in order to search for services;
3) Lack of self-organization: data owners have to find a working CS/W portal to register to. Without proper tool, management and maintenance can be difficult and challenging.

We argue that P2P systems can be an effective approach to address the above-mentioned problems (see Figure1). 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) would not join and leave the network arbitrarily; in most cases these servers are made accessible 24-7.
2) The volume of dataset served by OGC servers is huge in general while the number of servers is considerably relatively small.
3) There are a greater number of users (than the number of servers) that are dynamic and transient (*i.e.*, imagine users are P2P nodes in the system and they joins and leaves the system frequently).



(a) Service Discovery by CS/W server



(b) Service Discovery in P2P System

Figure 1 Scenarios of service discovery systems

2. RELATED WORK

Distributed hash tables (DHT) based structured P2P systems including CAN, Pastry and Chord, are designed to support exact match lookups (*i.e.*, one needs to know the exact identifier of a data object in order to be able to locate the node that stores it) [4]. Such system is vulnerable to node dynamic and would incur significant higher overheads than unstructured P2P networks for popular data. However in an unstructured P2P system such as Gnutella, if a node wants to find desired data in the network, the query has to be *flooded* through the network to find as many nodes as possible that share the data [5]. The flooding approach can cause a high amount of signaling traffic in the network and hence such networks typically have poor search efficiency. In addition, since there is no correlation between a node and the content managed by it, there is no guarantee that flooding will find a node that has the desired data.

Hybrid solutions such as JXTA [6] and Kazaa [7, 8] try to reduce network traffic by establishing a second routing hierarchy, the Super Nodes layer. A significant reduction of the data rate consumption can be achieved by the differentiation of nodes in Supernodes and Leafnodes, without loosing the network's complete self-organization [9, 10]. In this paper, we also use a hybrid design in order to achieve good load balancing in the system. Our system is different in that supernodes are not for serving rare data as in general hybrid system; they are designed to improve the accessibility of existing services.

In this paper, a locality-aware P2P approach for geospatial web services discovery is proposed in order to build a dynamic, scalable, and decentralized geospatial web service registry with flexible spatial query capabilities. In the proposed system, two identifiers, 1) a globally unique identifier (**GUID**) and 2) a **location quadkey***, are assigned to each node. The system uses GUID to manage user information, and the location quadkey to represent each node's physical location. In addition, two types of neighbour information, 1) each node's physical nearest neighbours and 2) **coverage quadkey**'s nearest neighbours, are cached in nodes. A geospatial web services is represented by a coverage quadkey describing its geospatial coverage. A node would keep a lookup table recording the service with its coverage quadkey.

Figure 2 depicts the core elements of this distributed geospatial service discovery system with hybrid P2P architecture. The node functions with and without an OWS implementation. It can be consider as a plug-in service if the node originally plays the role as OWS provider. Each node is composed of two engines: communication manager and local query engine.

The exchanged messages composed of binary packets of information and text strings represents requests.

3.1 Communication manager

The system consists of both structured and unstructured networks. New nodes join the unstructured network at first; it will be moved to structured network only when identified in stable state (*i.e.*, the node is up and running in the system more than a predefined period of time). In this way, the system would

* A reference that combines two dimensional coordinates into one-dimensional string by using Peano Space-Filling Curve [11] mapping.

3. SYSTEM DESIGN

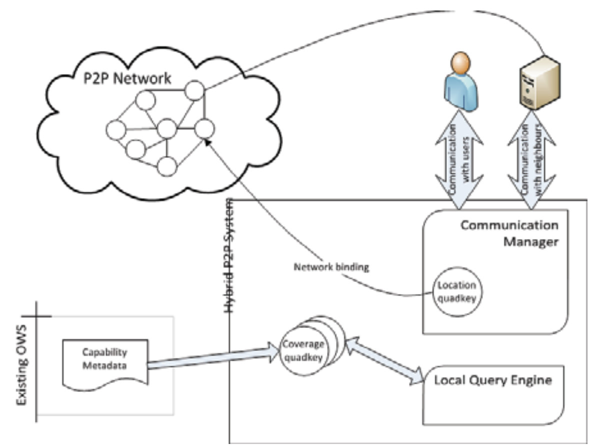


Figure 2 System Architecture (existing OWS is an optional component)

In this paper, a locality-aware P2P approach for geospatial web services discovery is proposed in order to build a dynamic, scalable, and decentralized geospatial web service registry with flexible spatial query capabilities. In the proposed system, two identifiers, 1) a globally unique identifier (**GUID**) and 2) a **location quadkey**

be less vulnerable to peer churn while best utilize the reliable and stable nodes.

Bootstrapping: Bootstrap nodes in the system would be geospatially sparsely distributed (preferably at least one node per continent). Each node has its location quadkey just as a general node in the system, and maintains a list recording all the active nodes in the system. When a new node joins the system, it contacts a bootstrap node whose location quadkey is closest to its own location quadkey. The contacted bootstrap node would reply with a node list based on geospatial existence of the node. Furthermore, the bootstrap node would cache and broadcast node information to other bootstrap nodes in case of any single point failure.

Node joining: A new node links to its neighbours in a mesh fashion, that is, a node would only have reach to its immediate neighbours. It is possible that a later joined node is assigned to an existing node. When it happens, a relation between two nodes may be established accordingly after first succeed transaction. The later joined node would become a neighbour within the reach to the existing node.

Node transition: A timer would track the uptime of a node as stability measure; it shall start after first succeed transaction the node made. When a node considered stable (i.e., up for a predefined period of time), it would temporarily do parts in two networks until there are enough members in DHT network. The first mesh node sustained to transition would create a new local DHT network. Later transformed node would join this network after consulting bootstrap node. In any case, a transformed (hybrid) node should always be accessible to mesh node even if it stops to maintain the links with immediate mesh node neighbours.

Node departure or failure: A node must notify the bootstrap node it boots from and its immediate neighbours when leaving the network. The bootstrap node would broadcast the departure to other bootstrap nodes accordingly. As for the ungraceful departure – node failure, since the node would most probably in unstructured network, it would be handled only when such failure is detected by any node that tries to make contact. When a node is not responding after a predefined T the node initiates the contact would send out the departure notification on-behalf of the failing node to the bootstrap node it boots from. Due to all bootstraps are in synchronization, the impact of such ungraceful departure would be minimized. The existing mesh nodes may remain intact while existing DHT nodes may patch route tables.

3.2 Local query engine

Local query engine supports only spatial queries in this system. The local query engine would first resolve the region of interest (if presents) into spatial coverage quadkey as main component of the query message. For example, if a user would like to search services in Calgary, the engine would translate the coordinate (51. 0833, -114.0833) into 0212131231. A node would check local cache for service match (*i.e.*, coverage quadkey of service is close to request quadkey). If empty result found, it would forward the request to physical neighbours for match. Each requested neighbours would repeat the action until a match is resolved. The response would be forward reversely to the node originates the query. Along the route, the service information would be cached at each node for future usage.

If no match found in the end, a node could contact bootstrap node to get a new list of nodes that are physically nearest to the requested coverage quadkey. The query process is the same as above described; service information would be cached in both clusters so cross clusters query is made possible.

4. DISCUSSION AND FUTURE WORK

We have presented a locality-aware hybrid P2P system for geospatial web service discovery. We expect the system provides two capabilities: service discovery and publisher. That is, if a user decides not to register to an OGC CSW server in order to publish his/her service, the service can still be accessible and searchable if using the proposed system.

Experiments are still ongoing by the time of this work is documented however the preliminary result indicates the potential and capability of the proposed system. We plan to extend the work in several directions. First, we will design and perform a system performance evaluation for validation. Second, we intend to study the query success rate under dynamic network environments. Moreover, system security is an important element to be explored.

References

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

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

[3] Open Geospatial Consortium Inc., "About OGC", Wayland, MA, USA. http://www.opengeospatial.org/ogc (accessed 30 Apr. 2010)

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

[5] Y. Chawathe, S. Ratnasamy, L. Breslau, N. Lanham, and S. Shenker, 2003. Making gnutella-like p2p systems scalable. In: *Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications*, pp. 407-418.

[6] S. Oaks and L. Gong, 2002. *JXTA in a Nutshell*. Sebastopol, CA, USA: O'Reilly & Associates, Inc.

[7] J. Liang, R. Kumar and K.W. Ross, 2004. "Understanding KaZaA". Polytechnic University, New York. http://cis.poly.edu/~ross/papers/UnderstandingKaZaA.pdf (accessed 30 Apr. 2010)

[8] J. Liang, R. Kumar, and K. W. Ross, 2005. The kazaa overlay: A measurement study. *Computer Networks Journal,* 49.

[9] J. Eberspächer, R. Schollmeier, S. Zöls, and G. Kunzmann, 2004. Structured P2P networks in mobile and fixed environments. In: *Proceedings of HET-NETs*, Ilkley, UK.

[10] B. T. Loo, R. Huebsch, I. Stoica, and J. M. Hellerstein, 2004. The case for a hybrid P2P search infrastructure. In: *Proceedings of the 3rd IPTPS.*

[11] Shashi Shekhar, Hui Xiong (Eds.), 2008. *Space-Filling Curves: Encyclopedia of GIS*. Springer, New York, pp. 1068-1070.