Distributed Planetary Object Name Service: Issues and Design Principles

Marcelo Dias de Amorim, Serge Fdida, Nathalie Mitton, Loïc Schmidt, David Simplot-Ryl

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Distributed Planetary Object Name Service: Issues and Design Principles

Marcelo Dias de Amorim — Serge Fdida — Nathalie Mitton — Loïc Schmidt — David Simplot-Ryl

N° 7042

Septembre 2009

ISSN 0249-6399 ISRN INRIA/RR--7042--FR+ENG

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE
INRIA
centre de recherche LILLE - NORD EUROPE

Distributed Planetary Object Name Service: Issues and Design Principles

Marcelo Dias de Amorim *, Serge Fdida * , Nathalie Mitton †, Loïc Schmidt † , David Simplot-Ryl †

Thème : Systèmes et services distribués
Équipe-Projet POPS

Rapport de recherche n° 7042 — Septembre 2009 — 13 pages

Abstract: The ONS is a central lookup service used in the EPCglobal network for retrieving the location of information about a specific EPC. This centralized solution lacks scalability and fault tolerance. We present the design principles of a distributed solution for ONS lookup service. In distributed systems, the problem of providing a scalable location service requires a dynamic mechanism to associate identification and location. We show that the use of Distributed Hash Tables (DHT) is a good candidate for distributing as it provides such a mechanism. We then outline how to adapt the DHT principles (operations on objects or nodes) to the ONS distribution problem.

Key-words: distributed ONS, EPC Global standards, RFID systems

* LIP6 - UPMC
† INRIA/CNRS/Univ. Lille 1

Centre de recherche INRIA Lille – Nord Europe
Parc Scientifique de la Haute Borne
40, avenue Halley, 59650 Villeneuve d'Ascq
Téléphone : +33 3 59 57 78 00 — Télécopie : +33 3 59 57 78 50

ONS distribué: Problématiques et descriptions

Résumé : L'ONS est un service de lookup centralisé utilisé par le réseau EPC Global pour récupérer l'adresse d'une information associée à un identifiant EPC donné. Cette solution supporte mal le passage à l'échelle et est peu tolérante aux pannes. Dans ce rapport, nous introduisons les principes d'une solution distribuée pour le service ONS basé sur l'utilisation de tables de hachage distribuées.

Mots-clés : Système RFID, ONS distribué, EPCGlobal

1 Introduction

As we move forward towards ambient intelligence environments where most devices are connected to seamless, ubiquitous networks, inter-enterprise interoperability becomes an essential prerequisite. Integrated complex networks, composed of a huge amount of different types of objects, will form the so-called Internet of Things [2]. Amongst these networks, Internet of Goods will manage information exchanges of a networked business-to-business world [1]. The architecture supporting this increase in scale should be designed accordingly to follow an open governance model.

The rationales for an open governance model have been strongly outlined at a political level but it should also be a key opportunity for businesses to think about technological and usage innovations as well as security, stability and performance requirements. Limitations and weaknesses of the current Internet architecture are indeed strong incentives to carefully study clean slate architectural approaches for the future Internet of things.

The large scale EPCglobal network will be part of the Internet of Things. One of its key standard-based components is a centralized objects directory service called the ONS (Object Naming Service) [5] and based on the DNS (Domain name System) [4]. Given the importance of ONS systems in the near future, it is wiser to take the time to develop alternative solutions and compare them with the incumbent architecture as it is still possible today to influence the evolution of the networks.

Since the World Summit on Information Society (Tunis, 2005) and the negotiations on Internet Governance that occurred, the European awareness on this topic arose dramatically. The Internet of Things architectural design is consequently considered, rightly or wrongly, as a political issue that has to be negotiated on a governance perspective before it can be implemented. In particular, logistic applications that could imply sensitive product localization (eg. drugs, weapons, nuclear waste) or sensitive applications (product recall, anticounterfeiting applications, food or drugs safety control) are considered as critical applications as well as the architectures that support them. In this perspective, the European Commission is currently working on a draft communication on the Future networks and the Internet, Early Challenges regarding the Internet of Things that emphasizes the need for an open governance model of its architecture.

On the other hand, the current architecture of the Internet has received more and more criticisms while a growing number of users and new usages are accompanied by increasing security failures (spamming, Kaminski bug, worms). A clean slate approach for the next generation Internet is quoted more and more often.

For all these reasons, a single approach for the Internet of Things architecture that would be based on the current Internet standards should be carefully considered. A wiser approach would suggest comparing the incumbent architecture with alternative models. While the incumbent approach would probably enable a huge economy of scale by using already widespread technologies, alternative approaches could be the opportunity to address governance as well as security issues. It has to be noted that security issues and service continuity regarding Internet of goods operations will be much more critical than it is with

the current Internet. Strong technical and economical comparative studies of both models should therefore be conducted.

But a pertinent approach should also carefully address the emergence of usages and critical applications based on these architectures.

2 Towards multi-root ONS

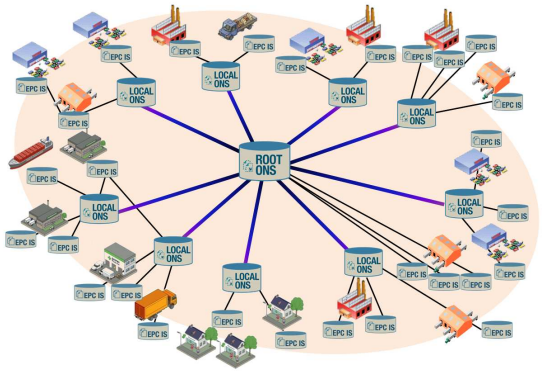
2.1 Problem statement

Since the launch of the EPCglobal standards in 2001, more and more companies have started to explore the possibilities of using the related technologies, services and interfaces such as the EPCIS, which represents the first step towards the usage of the EPCglobal network. As the foundation for the kind of connectivity that will increase visibility throughout global supply chains and help trace shipments, combat the introduction of counterfeit products and prevent retailer out-of-stocks, new class of applications are various. But to understand the need of the EPCglobal network evolution we have to remember that, at the beginning, it was based predominantly on the needs of food manufacturers and retailers. Therefore, the architecture of the current EPCglobal network is heavily focused on the needs of these business scenarios. However, in the course of time, organizations are beginning to adopt RFID further up and down the supply chain and also beyond small scale or sporadic deployments, involving a growing number of industries in various sectors such as healthcare, aerospace, automotive, defense etc...

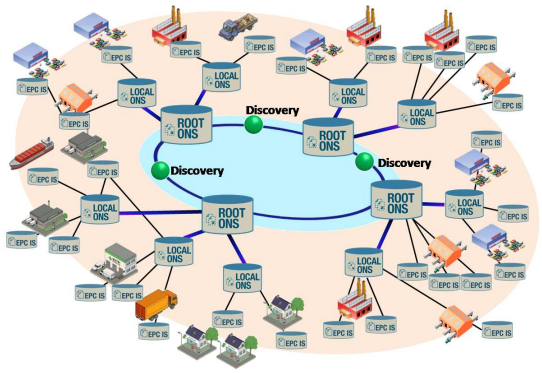
So the next phase of the EPCglobal network development will have to allow flexible integration of product information provided by a large number of organizations horizontally across the supply chain, and also vertically across various other industries. This moves from small localized activities to large cross-company and cross-country networks will require both more complete and more comprehensive data sets. This implies efficient data synchronization, guaranteed data availability and improved data security. There is, as a result, a need for data alignment and standards evolution, including one for a so-called Object Naming Services (ONS).

The Object Name Service (ONS) defines the interface for lookup services by providing quasi-permanent or relatively static links between the identity of a company responsible for an object (often the manufacturer) and the authoritative information services which that company provides. This company has a database made of object relative data. These information are managed via information systems EPCIS (Electronic Code Product Information Services) of the network partners. Based on the Internet DNS protocol, the ONS is a client/server model hierarchically organized, offering the possibility to orient requests coming from client applications towards the EPCIS of the right company.

ONS Standard version 1.0.1 [5] designs the EPC network as a global information system, centralized, in which several local ONS are interconnected. To respond to application queries by routing them towards the right local ONS, a root domain is implemented (onsepc.com). This root domain, or root-ONS, is the system core, structuring the network and localizing associated services. This is a unique and authoritarian root, it refers in facto every local ONS of



(a) Centralized (existing) ONS.



(b) Multi-root ONS.

Figure 1: Distributed Vs Centralized ONS.

the EPCglobal network. The emergence of multiple roots highlights the need for GS1 to make the standard evolve toward a symmetric architectural model, like illustrated by Figure 1.

2.2 Challenges

GS1 France has launched its own ONS root platform in order to respond to geopolitical concerns and to elaborate theoretical concepts of the EPCglobal network components into a real environment. Other regions in the world are also evaluating to have their own ONS running. That's why, to support various organizations for achieving world-wide adoption and standardization of the EPC technologies in an ethical and responsible way, this increase in scale for the network also demands the development of an open governance model. Subsequently, this open governance model can be extended to incorporate various ONS systems from other parts of the world, both on technical and business aspects that would be administrated under a common set of rules. Drawing on the ONS root operated by GS1 France, a set of rules for the governance, including for instance standards for naming issues and Discovery Services with the use of security tools such as certificate authority, privacy management, etc. has

to be explored. Furthermore, the aim of this platform is to give the European Community a leadership role in developing ambient intelligence in the supply chain and thereby enhancing competitiveness through leadership in implementing broadly-based, open business enterprise networks

2.3 Initiatives

To address this emerging issues, the ANR WINGS[1] [3] project has been launched. It focuses on the ONS as seen as the link between the physical and the virtual world in the Internet of Things. This latter one needs to be open, visible and available to all. WINGS gathers research institutes (INRIA[2], UPMC[3] and GREYC[4]) who are well versed in RFID developments and business research, FT (Orange Lab) and the French Network Information Center (DNS Registry) (AFNIC) and GS1. GS1 France brings the 'real life' requirements to the project by providing the wide user base with many tens of thousands of SMEs, and on the ground, indigenous capabilities to disseminate the results of the research worldwide.

3 A DHT-based solution

In this section, we present our proposal to solve the problem of multi-root ONS. The basic idea behind the solution is to rely on the concept of distributed hash tables (DHT). The reason for such a choice, as it will become clearer in the remainder of this document, is that DHTs gather enough properties to cover all the requirements initially imposed by GS1 and stated in Section 2.1. We started from the assumption (as stated by GS1) that we should not consider any constraints imposed by the system. In this way, the solution proposed here is not final; instead, it must be discussed and enriched in order to integrate all the necessary specificities.

3.1 Distributed hash tables

Providing a scalable location service in distributed systems is a difficult problem. This requires a dynamic association between identification and location of an object, and the specification of a mechanism to manage this association. In response to these requirements, distributed hash tables have been adopted as a scalable substrate to provide a number of functionalities such as information distribution, location service, and location-independent identity upon which a variety of self-organizing systems have been built. The functionality of decoupling identification from location, and of providing a general mapping between them, have made the DHT abstraction an interesting principle to be integrated in large-scale distributed lookup services. The idea is to use a hash function to distribute location information among rendezvous points throughout the network in a uniform fashion. This hash function is also used by a source

[1] Widening Interoperability for Networking Global Supply Chains
[2] www.inria.fr
[3] www.upmc.fr
[4] www.greyc.unicaen.fr/

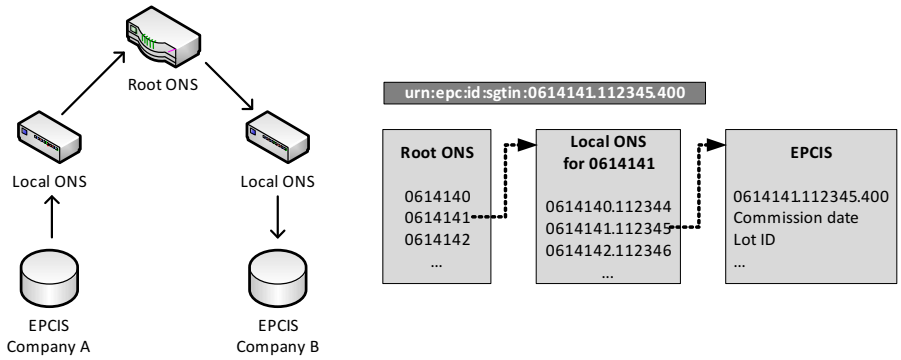


Figure 2: Original lookup service as presented by GS1.

to identify the rendezvous point that stores a destination's location information. Distributed systems that make use of such a strategy inherit robustness, ease-of-operation, and scaling properties.

DHTs provide a general mapping between any information and a location, establishing then a location-independent routing substrate. According to the design of the location structure, a content is associated with an identifier, which is in turn hashed into a key of a virtual addressing space. Partitions of this same virtual space are assigned to nodes in the network. A node whose partition contains a key is responsible for storing the location of the content associated with this key. This mapping between keys and nodes determines the information that each node is responsible for.

3.2 Adapting the DHT principle to the distributed ONS problem

We consider as the basic operation of the traditional ONS lookup service as the one presented in Fig. 2. It is clear that the principle of DHT responds to the requirements of a distributed multi-root ONS. More specifically:

- Root ONS are the nodes on the DHT that manage the addressing space.
- EPCIs are the "keys" that serve the location service. They are mapped into a position on the addressing space. The root ONS that manages this position has a pointer to the local ONS that really manages the location of the object.

Fig. 3 illustrates the modification introduced in the lookup service when a DHT is used. In this sense, the root nodes serve as a virtual collection of pointers. This principle has two main advantages. First, it allows the systems to be completely distributed and independent of the underlying system. Second, the real location information remains within the boundaries of the local management tree. We will basically use two hash distributed functions.

In the following, we go into more details of the system. But before, we would like to underline the necessary conditions for the system to work:

- The two hash functions must be the same for all nodes participating in the system. This condition is necessary for the consistency.

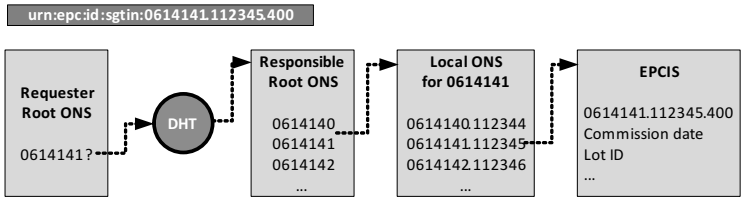


Figure 3: Lookup service when a DHT is introduced. The responsible Root ONS is not necessarily the referent ONS of the local ONS but can be any of the ONS roots.

- The addressing space must be well defined and should be the same for both hash functions.
- The same hash function has to be used for both registering a content and requiring for it.

3.3 Structure of the addressing space

We propose to rely on a Chord-like substrate to implement the DHT [6]. Chord defines a simple wrapped one-dimensional addressing space (i.e., a circle) of size 2^K, where K is the order of the system. In other words, a position on the circle can be any value within the range $[0; 2^K - 1]$. The value of K depends on the expected scale of the system. A value traditionally used in existing systems is $K = 128$.

We define two hash functions: $f(\cdot)$ and $g(\cdot)$. They will be used, respectively, to position root nodes and pointers to objects on the curve. The reason of defining two functions is to give enough freedom to the system to separate nodes from objects.

First, let I_{r_i} be the identifier of root node r_i. This identifier can be an IP address, a URL, or any other value.[5] It is important to underline that all root nodes have to join the DHT to be considered at the top level of the ONS. Root node r_i is associated with a position on the curve given by:

$$H_{r_i} = f(I_{r_i}). \tag{1}$$

Root r_i must know its predecessor and successor nodes on the curve, i.e., the nodes r_p and r_s resp. on curves such that there is no node x such that $I_{r_p} < I_x < I_{r_i}$ (resp. $I_{r_i} < I_x < I_{r_s}$); we refer to these nodes as p_{r_i} and s_{r_i}.[6] The reasons are twofold. First, such information will be used to route requests on the system. Second, it allows us defining the notion of "responsibility region". Without loss of generality, root node r_i is assigned the region C_{r_i} comprised between its own position and the one of its predecessor (plus one):

$$C_{r_i} = [H_{p_{r_i}} + 1; H_{r_i}]. \tag{2}$$

[5]This is a point to be discussed. For the time being, we make abstraction of technical dependencies.

[6]In order to improve the efficiency of the system, we also assume that each nodes also knows some other nodes on the curve. The specific strategy will be defined ultimately.

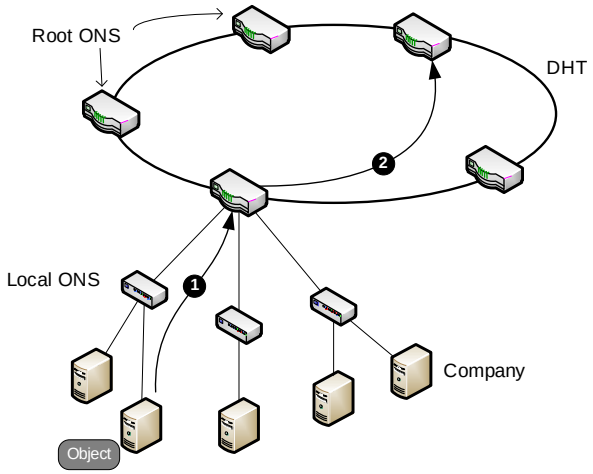


Figure 4: The `put` operation.

We refer to this interval as root r_i's responsibility region. This means that all objects that will be associated with any point in the interval C_{r_i} will be under the responsibility of node r_i. To this end, we use the second hash function $g(\cdot)$ to link objects to points on the curve. Let us call J_{e_j} the identifier of object e_j. Its point on the curve, L_{e_j}, is simply given by:

$$L_{e_j} = g(J_{e_j}). \tag{3}$$

In this way, we say that root r_i is responsible for object e_j iif $L_{r_i} \in C_{r_i}$. By responsibility, we mean that node r_i knows which Local ONS can handle a request.

3.4 Operations on objects

Before explaining how nodes join and leave the circle, we describe the operations on objects. The operations are in number of three: `put`, `get`, and `remove`.

3.4.1 Put

It consists in registering a new object in the system. For the sake of explanation, Fig. 4 shows the different steps of the `put` procedure. First of all, the company sends a registration message of a new object to the root node it is directly connected to (arrow ❶). Note that this step is inherently compliant with the current system. This root will be responsible for inserting the pointer into the DHT. The pointer is basically the address of the local ONS. It computes $g(J_{e_j}) = L_{e_j}$ and sends a `put` message to the root node r whose responsibility region contains L_{e_j}, i.e., $L_{e_j} \in C_r$ (arrow ❷).

3.4.2 Get

The `get` operation consists in obtaining the pointer to some local ONS that has the location of the object (cf. Fig: 5). This operation is very similar in essence

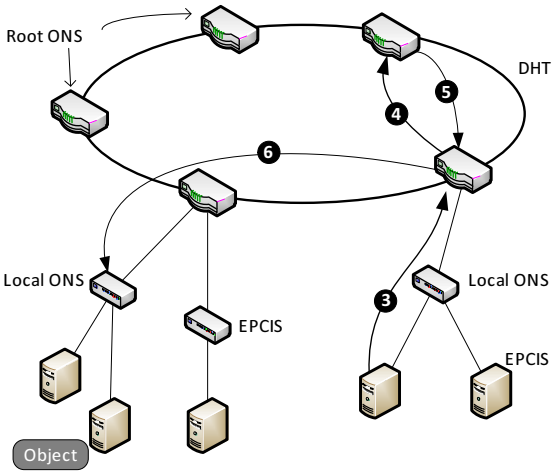


Figure 5: The `get` operation.

to `put`. The root ONS that receives a request for information about a given object (arrow ❸) first verifies if it knows the exact local ONS that manages this object. If not, it performs a lookup on the DHT. As in the `put` operation, it computes $g(J_{e_j})$ and asks about information on the object to the respective responsible root ONS (arrow ❹). If the object is properly registered (through the `get` operation), the right local ONS is given as an answer (arrow ❺). If not, an error message must be returned. In case of a positive answer, the request can then be forwarded to the right local ONS directly (arrow ❻). Note that the way of routing the request towards the right local ONS depends on the kind of address we use. In case of the use of IP addresses, we shall use the traditional DNS and IP routing like in the current implementation of the ONS. For other kinds of addresses, several implementations are feasible.[7]

3.4.3 Remove

The `remove` function is basically the same as `put`, with the difference that the rendezvous node removes from the DHT the indicated entry (if it is in the system).

3.5 Operations on root nodes

We now address the problem of how root nodes join and leave the system. These are fundamental operations toward a robust system.

3.5.1 Join

When a new root node r_i wants to join the system, it must first contact one of the nodes already in the system. This latter will help the new node to obtain the coordinates of the node that manages the region containing $f(I_{r_i}) = H_{r_i}$.

[7]There exist several possible implementations of this routing step. This needs to be discussed in case our proposition is chosen.

Let us call this node m_{r_i} ($H_{r_i} \in C_{m_{r_i}}$). To be inserted in the DHT, the new root node will inherit from m_{r_i} the part of the region comprised between its position and the position of m_{r_i}. More formally:

$$C_{r_i} = [p_{m_{r_i}} + 1; H_{r_i}]; \tag{4}$$
$$C_{m_{r_i}} = [H_{r_i} + 1; H_{m_{r_i}}]; \tag{5}$$
$$p_{r_i} = p_{m_{r_i}}; \tag{6}$$
$$s_{r_i} = m_{r_i}; \tag{7}$$
$$p_{m_{r_i}} = r_i; \tag{8}$$
$$s_{p_{m_{r_i}}} = r_i; \tag{9}$$

By inheriting a region, the new node also inherits the corresponding pointers of objects already registered in the system. Note that the join procedure requires collaborative behavior. This means that, in its basic form, nodes should be hierarchically equivalent and follow the same rules.

3.5.2 Leave

Nodes leaving the DHT can happen either smoothly or abruptly (due for instance to a fault). Here, for simplicity, we focus on the smooth case.[8] When a node leaves, it must give its responsibility region to its successor and trigger the update of the DHT:

$$C_{s_{r_i}} = [H_{p_{r_i}} + 1; H_{s_{r_i}}]; \tag{10}$$
$$p_{s_{r_i}} = p_{r_i}; \tag{11}$$
$$s_{p_{r_i}} = s_{r_i}; \tag{12}$$
$$\tag{13}$$

Here, by returning its responsibility region, the leaving node also transfer the pointer it maintains to its successor.

3.6 Bootstrap

As in any distributed systems, there is a critical step which is the bootstrapping phase. This is even more important when a node wants to join the system. In the case of this project, this issue can be easily tackled as EPCglobal can serve as an "introducing point". The purpose here is that, when a node needs to join the network and thus to perform a `join` operation, its needs either to know a contact node already in the system or to request for it to an entry point. For equality purpose among ONS root nodes, we suggest that EPCGlobal plays the role of entry point. It means that when a new root node needs to join the system, it requests a contact node address to EPCGlobal. EPCGlobal replies then with any active node in the system (at random). The new arriving ONS root can then contact that node and be effectively introduced in the ring.

[8] There are many different approaches to address the situation of abrupt leaves; an example is replication that allows recovery. We postpone this discussion in case our approach is kept.

If the new arriving node is the very first one to enter the system, i.e., there is no node yet, it ONS will be given the responsibility of the whole address space. All other roots perform a `join` operation like specified in Section 3.5.1. It is important to underlying the fact that the very first root introduced in the system does not get any importance compared to any other one introduced later. For instance, we shall just in a first step add the current ONS root in the system.

4 Discussion

4.1 Replication and multi-hash system

For the sake of data security and consistency, data need to be duplicate over the network. The `put` operation has to return several pointers, which can be obtained in two different ways. First, the hash function $g(\cdot)$ may return up to k pointers, where k is the level of redundancy chosen. Second, we may decide to define multiple hash functions that would be all applied in each operation of the system.

The use of more than one hash function offers the robustness if an ONS is down (local or root). If the first contact node fails (or the local ONS which seems to contain the required information), the system may request for a second contact node with a second hash function. The pointers to the different storage places are thus duplicated.

4.2 Search mechanism

So far, we described the basic way for searching and routing in the overlay structure. Each root node keeps a routing table in which it stores the addresses of its predecessor and its successor. In addition, it stores the address of its $2^{l^{th}}$ neighbors, for $l = 0, \ldots, K$. These nodes are thus chosen in an exponential way to optimize the searching time in an homogeneous distribution. For instance, if we suppose that root nodes are numbered in the circle in a contiguous way, node 0 stores nodes 1 (2^0), 2 (2^1), 4 (2^2), 8 (2^3), ..., 2^{K-1}, as weel as 2^K. When node 0 has to handle a request for a content with identifier X, it forwards it to the node in its table which has the closest identifier to X.

According to the application time and memory requirements, this routing may be enhanced by letting nodes have different routing tables. Nodes in table can be chosen in a different way or may be more numerous. This latter case would allow a quicker search but would imply a bigger storage space on ONS-roots and more data to update in case of apparition/disappearance of a root. Indeed, a tradeoff is to be found.

4.3 Multiple entry points

For retrieving EPC information, the system will ask its ONS-node (its entry point in the ONS structure) for the local ONS that contains the required information. When this entry point fails, the system depending on it will not be able to provide an answer. Multiplicity of entry points in local system provides the ability to send the request to a backup entry point in this case of failure.

5 Conclusion

There are several advantages in the use of DHT for distribution. The scalability and the dynamics of the network are the first of them. Because of the ability for joining or leaving the network, a new ONS-node can be added very easily. Moreover, as there is no predominant station and that none of them can achieve more power than another one (they depend on each other), the whole system automatically recovers from a failure of one of them. Such DHT systems are already used and have shown their efficiency in very large scale systems so the scalability is here a well-known advantage.

In this proposal, there are some specifics advantages for the distribution of the EPC Global ONS, such as the conservation of the existing current system of ONS root. This root becomes a node like any other. The result of information about EPC request is an address of a local ONS containing such information. So even if the ONS-node of this local ONS is down, all information about EPC managed by local ONS remain available.

References

[1] EPCglobal Inc. `http://www.epcglobalinc.org`.

[2] ITU. *Internet Reports 2005: The Internet of Things*, ITU, 2005.

[3] N. Pauvre. Wings project: Widening interoperability for networking global supply chains. In *Wireless Vitae*, Aalborg, Denmark, 2009.

[4] DNS. Domain Name Service. `http://www.howstuffworks.com/dns.htm`.

[5] EPCglobal ONS Standard. `http://www.epcglobalinc.org/standards/ons`, 2008.

[6] I. Stoica, R. Morris, D. Liben-Nowell, D. R. Karger, M. F. Kaashoek, F. Dabek, and H. Balakrishnan. Chord: a scalable peer-to-peer lookup protocol for internet applications. *Networking, IEEE/ACM Transactions on*, 11(1):17–32, 2003.

Centre de recherche INRIA Lille – Nord Europe
Parc Scientifique de la Haute Borne - 40, avenue Halley - 59650 Villeneuve d'Ascq (France)

Centre de recherche INRIA Bordeaux – Sud Ouest : Domaine Universitaire - 351, cours de la Libération - 33405 Talence Cedex
Centre de recherche INRIA Grenoble – Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier
Centre de recherche INRIA Nancy – Grand Est : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex
Centre de recherche INRIA Paris – Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex
Centre de recherche INRIA Rennes – Bretagne Atlantique : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex
Centre de recherche INRIA Saclay – Île-de-France : Parc Orsay Université - ZAC des Vignes : 4, rue Jacques Monod - 91893 Orsay Cedex
Centre de recherche INRIA Sophia Antipolis – Méditerranée : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399