A Context-Relational Approach for the Internet of Things

Jamie Walters*†, Theo Kanter†
*Department of Information Technology and Media
Mid Sweden University, Sundsvall, Sweden
email: jamie.walters@miun.se
†Department of Computer and System Science
Stockholm University, Forum 100 Kista, Sweden
email: {kanter, jamiew}@dsv.su.se

***Abstract*—Context-centric applications and services are premised on the ability to readily respond to changes in context. Centralized approaches to enabling this are undermined by their dependencies on DNS naming services while decentralized approaches using DHT variants have been centred on the provisioning of the underlying context information, creating information-centric rather than context-centric solutions. A dynamic Internet of Things mandates a new paradigm; approaches storing, discovering and associating context entities relevant to their context state. In this paper, we explore such a paradigm, and with the implementation of a prototype, show the advantages of moving towards the notion of context-state centricity on the Internet of Things.**

Keywords-context awareness; context; context models; Internet of things; context proximity; sensor information; p2p context

I. Introduction

Current trends in computing bring the paradigm of pervasive and ubiquitous computing into focus. Users are now even more connected; demanding a range of everything everywhere services. These services, including as social networking and media, benefit from the availability of context information seamlessly gathered and shared; providing customized and user-centric experiences.

Dey[1] contributed significantly to the understanding of this context centric paradigm and its central role in advancing ubiquitous and pervasive computing research. The two definitions introduced in [1], remain concrete definitions; pillars of modern context aware computing research. Defining context as:

"any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves"

and context awareness as:

"a system is context aware if it uses context to provide relevant information and or services to a user, where relevancy depends on the user's task."

Research towards this realization of context awareness has largely been focused on the ability to construct an accurate and timely representations of a user's state from the information gleaned from an intricately woven infrastructure of sensor and actuators.

Such an interconnected things infrastructure is expected to have an installed device base in the range of several billion [2] and will be capable of supporting a diverse set of experiences ranging from personalized and seamless media access, to intelligent commuting or environmental monitoring services. Dubbed the *Internet of Things*, it will incorporate devices such as electronics, vehicles, mobile telephones and even municipal infrastructures and the people themselves, merging towards the paradigm of everywhere computing [3].

This underpinning Internet of Things (IoT) is a key enabling factor in the creation and deployment of applications and services in response to the situation of a user and his current relationship with his environment, applications and services. This, according to Dey [1] constitutes the working definition of context and cements its central role in the explosion of pervasive and ubiquitous applications and services.

Approaches towards the realisation of such an Internet of Things have largely been focused on the design and implementation of systems that are capable of provisioning context information with acceptable degrees of availability, accuracy and reliability. However, our ability to discover related entities are limited by the arbitrary storage mechanisms such as the DHT approaches used by Kanter et. al. in [4] and Baloch et. al. in [5]. This resulted in bottom up approaches to finding related context entities through the use of costly searches over their constituent context information. Additionally, indexing and caching approaches such as [6] cannot offer guarantees in freshness of information as is required to drive real-time applications.

This in turn mandates newer approaches to the storage and retrieval of context entities in order provide

 ISBN: 978-1-61208-239-4

support for the wide array of context centric applications and services that will occupy an Internet of Things. One such approach is the storage and discovery of entities as a factor of their overall context relationships. In this approach, we seek to persist and discover related context entities solely over their context relations and verify that we are capable of retrieving these entities with the same level of accuracy while negating the need for applications and services to compose multiple queries over composite context information.

Further, we explore the advantages of context-relational queries with respects to publish - subscribed based approaches. We demonstrate that relational discovery permits us to reduce the subscription related communication and computational overheads by changing the dynamics used to create and maintain subscriptions.

While we show that solutions can be created for simplifying the discovery of related entities within a region of interest or a degree of relationship, defining the rules for quantifying this relationship is a non-trivial problem. This requires further investigation into approaches for deriving and representing degrees of relationships or context proximity amongst entities in order to create the underlying relational context networks for driving future applications and services.

The remainder of the paper is structured as follows: Section II looks at the background work and motivation, Section IV outlines the approach. Section V presents our verification and results while Section VI summarises the conclusions and future work.

II. Background and Motivation

The early context provisioning architectures that offer support for an Internet of Things varied significantly with respects to both implementation and approach. They however converge on fulfilling a set of fundamental requirements capable of enabling access to the information required for driving the exploration of ubiquitous applications and services.

DNS Independence was explored in MediaSense [4] and SCOPE [5] through the use of Distributed Hash Table (DHT) type overlays such as [7]. This moved context provisioning architectures away from centralized approaches such as SenseWeb [8] and IP Multimedia Subsytem (IMS) [9], which are dependent on Domain Name Service (DNS) as a means of locating service portals, users and applications.

Real Time Provisioning using Distributed Hash Table (DHT) based architectures such as [4] and [5] enabled the resolution of context information within times that were comparable with User Datagram Protocol (UDP) and deemed adequate enough to support real-time context dependent services.

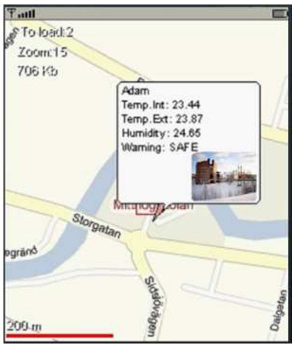

Figure 1. A Simple Mobile Context Awareness Application

Self-Organization of constituent context information was explored in by Walters . et. al in [10] verifying that we are capable creating solutions that retain the advantages of DHTs while introducing more dynamic self-organization qualities.

The problem, however, is that architectures that seek to represent the context interactions supporting the real world must be able to represent context entities over their higher level context states and the degrees or relational affinity among these states. This would serve to extend early solutions such as those employing DHTs to effectively create context-state centric networks.

Early solutions enabled the provisioning of context information concerning users and their environments. As such, a user finds it relatively simply to understand his current state. Such a user will readily comprehend: *It is five degrees Celsius and I am walking 2 km/h on the High Street.* Additionally, this information is made available to applications and services interested in the user's state of being.

Such approaches permitted the creation of simpler applications such as [11], shown in Figure 1, which permits users to locate other users based on their context.

However, context centric applications and services are becoming more collaborative requiring approaches to consider not only localized user context but rather a user's relationship within a dynamic context centric network.

Schilit [12] regarded context as being composed of three key aspects used to define an entity's situation: *"where you are, who you are with[near], and what resources are nearby"*. With further evolution into *Presentities* [13], this can be refined as: *a presentity, its related presentities and the degree of their relationship.*

Here, the complex context networks that underpin the pervasive computing paradigm exists as more general collections of *presentity-relationship-presentity* triples shown in Figure 2, where the relationship is the degree of affinity between the current context states of the presentities. Enabling us to create context networks

Copyright (c) IARIA, 2012. ISBN: 978-1-61208-239-4

where organization is achieved at a higher level. We used the terms presentity and context entity interchangeably in this paper.

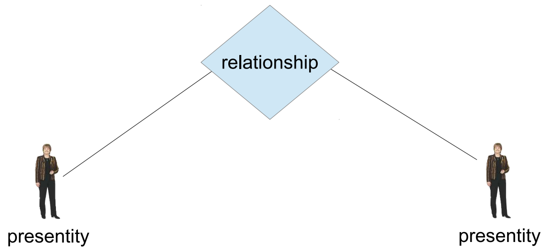

Figure 2. The Presentity Triple

While semantic approaches to triples provide a means of characterizing the relationships between entities, it obtains limited expressiveness with respect to a measurement of affinity. Here, models that support a metric over these relationships are, according to Schmohl and Baumgarten [14], complementary in characterizing the types of relationship illustrated in Figure 2.

III. Space Filling Curves

Peano first proposed the idea of a space filling curve as a finite curve which begins at the origin and traverse every point in an n-dimensional hypercube [15]. Several versions of space-filling curves have been derived with varying properties. Our approach regards each context state as corresponding to a point in n-dimensional space traversed by a space-filling curve. Thus, of particular interest are space filling curves having superior locality preservation. That is, the order in which the curve visits a point highly correlates with their observable proximity within the n-dimensional space, and subsequently their distance of separation on the curve.

Additionally, the heterogeneity of context information required a variant that was capable of handling dimensions measured on different scales without the need for padding. The Hilbert Space Filling Curve obtains superior locality preservation [16] and for our implementation we selected an extension, the Compact Hilbert Space Filling Curve [17], which is capable of generating Hilbert Indices for n-dimensional hypercubes where sides are of different scales.

IV. The Approach

Our approach is not focused on the creation of a new context model, but rather the organization of entities represented by existing context models. To this end, we adopt the context models described in [18] and [19] as outlined below. In response to the shortfalls discussed in Section II, we are required to organize and represent

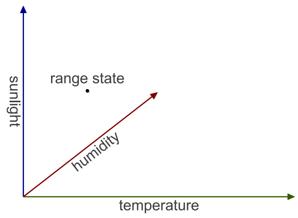

Figure 3. Weather Range

presentities and their context at a higher order, permitting us to reason over the state relations of presentities rather than their underlying context information.

We define Context Networks as collections of *presentity triples*, where a presentity through its context information, possesses a specific state with a degree of relationship to other states within the context network. To enable this organization, each context state occupied by a presentity is indexed using a space-filling curve, distributing these values on a modified DHT overlay to preserve order and degrees of relationship among presentities. While this approach is partially explored in [20], its focus on geographical information does not provide a sufficient response to the requirements of Schmidt et. al. in [21]. We further implement a range query algorithm for querying presentities over current context states as opposed to raw underlying context information.

Our key point of departure from existing context aware implementations is therefore that we store, retrieve and query presentities relative to their state as opposed to their underlying context information

A. Presentity

We regard a presentity [13] to be all entities which, at any point in time, possess presence and context information that describes its current state. A presentity may contain any combination of context information and be continuously updated and retrieved. The distance or similarity between the states of presentities determine their suitability for a context-aware application or service.

B. Context Range

We adopted the concept of*context ranges* as described by Schmohl and Baumgarten [18]. Here, a context range is a group of individually observable context information that contribute to describing a specific category of context. For example, the range *weather* could be defined over temperature, humidity and sunshine intensity. We model each range as an n-dimensional hypercube as shown in Figure 3, where each face of the cube corresponds to a dimension of context information.

 ISBN: 978-1-61208-239-4

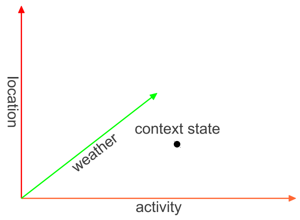

Figure 4. Application Space

C. Context State

The context state was defined by Padovitz [22] as the current situation of a context entity as defined by its current set of context information. A context state is therefore a point in an n-dimensional space whose position is defined by the values for each context dimension. A context state only exists for a specific time, t. We deviate from the Padovitz's [22] original definition and create two states; namely the *range state* as shown in Figure 3 and the *context state* as shown in Figure 4. A range state is the current or state of an entity within a particular range, e.g., the current state of the weather. Each state is a point in a hypercube where each face corresponds to an actual piece of context information.

The context state is the current state of an entity over a subset of its ranges, e.g. weather and location and time. This corresponds with the application space defined by Padovitz [22]; the n-dimensional space within which a state is valid with respects to the fulfilment of an application or service. As is currently attained, an application in this scenario would require a specific weather in a particular location at a given time. An application domain D could therefore be any combination of ranges R such that:

$$D \leftarrow \{R_1, R_2, R_3, ...R_i\}$$

And a context state, C, within this domain being expressed over all the range states, r of a presentity such that:

$$C \leftarrow \{r_1, r_2, r_3, ...r_k\}$$

Application spaces are, therefore, hypercubes with the difference being that each dimension being a range, see Figure 4

D. General Persistence

In order to support our approach, we created two DHT overlays namely, a General Persistence and a State Persistence. The general persistence overlay is an extension of our previous work towards a context provisioning architecture and uses a DHT overlay for persistence. When a presentity is introduced into our solution, its constituent context information is decomposed and persisted in the general persistence overlay, given an identifier, a URI of the format:

$$dcxp://user@domain/contextdimension$$

as described in by Kanter, et.al. [4]. For this, we used a Pastry DHT with SHA-1 as the underlying hashing algorithm, making no significant modifications. We selected a URI of this format in keeping with our previous work, however any chosen unique identifier would provide sufficient support.

E. State Persistence

In an effort to improve the performance for finding related presentities, we created a State Persistence component. This is a modified DHT used for persisting the current context state of a presentity derived from a calculation of its Hilbert Index [17]. The Hilbert Index we derived for each context state is of datatype *long*. We therefore modified the DHT overlay to created a distributed *long* overlay, effectively replacing the SHA-1 [23] hashing function in the DHT and modifying the overlay construction functions to build an ordered overlay which stores long values in a 96-bit distributed space. The resulting being that the space is organized in a clockwise order-preserving manner.

F. Indexing Presentities

The main aim of the solution is to index presentities over their current context state and permit applications and services to locate these presentities with respects to the affinity between current states of context. In achieving this, we first index all presentities within the general persistence component including all composite context information and the presentity identifier. Secondly, each context state is indexed.

In order to achieve this, we model each range described in Section IV-B as an n-dimensional context space containing a Compact Hilbert Space filling curve. Figure 5 shows a such a Hilbert Curve traversing a 2-dimensional range. The resulting is an order on all the possible states within a range as points on a line, where the distance between the points are indicative of the similarity between the states.

Indexing the context state of an entity is achieved using the same method, with the exception that each presentity is indexed over its constituent range indices, i.e., an index of indices.

The locality preserving property of the Hilbert Curve ensures that presentities with similar context values are assigned similar Hilbert Indices and their states subsequently are indexed within close proximity.

G. Range Search

This results in two types of index values for each presentity, namely range indices and state (application)

 ISBN: 978-1-61208-239-4

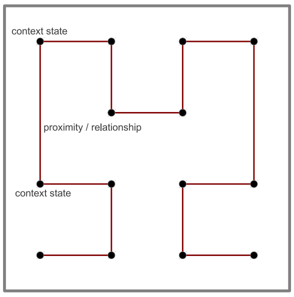

Figure 5. Context State Indexing

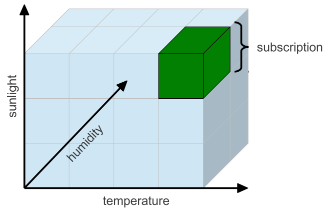

Figure 6. Range Subscription

indices. We persist these indices in the state persistence layer by extending them to add additional information about the type of index being persisted. The first 16-bits describe the type of index, i.e., context state of range state. The consecutive 16-bit represented the range being index while the final 64-bits contains the index value for the range or context state.

The consequence of this being that index information is successively partitioned by type, range and by value. Replication is employed on the state persistence overlay in order to negate any issues that may arise with this partitioning. Additionally, the distance a presentity is stored from a another presentity is a factor of the context similarity or distance they between them; presentities with a similar expression of context are stored close together on a DHT.

Searching for presentities within the solution is achieved from a context state perspective. Applications typically search for presentities that, to a certain degree, possess similar context states. By utilizing a space filling curve, we created an order on the presentities as a factor of their states. Consequently, the distance between any two points on this curve is a factor of the distances between their collective underlying context information.

We exploit this property in creating range queries for locating groups of presentities that are within a specified area on the curve and therefore within a specified area of the n-dimensional context spaces and subsequently with similar context information. An application wishing to locate an entity, first decides on the maximum and minimum values for each dimension and with these values, calculate the maximum and the minimum Hilbert Indices, indicating the maximum distances to be traversed on the space filling curve. We then construct a single query which is sent to the nodes responsible for upper bound, the lower bound and median value in the range. Each node, on receiving the query forwards it in either a clockwise or an anti-clockwise direction around the DHT depending on its position in the range. The query is not forwarded beyond the nodes located at either ends of the query range, with the results returned to the querying node. With this approach, we can query entities based on their context states and the degrees of affinity amongst these states, with limited knowledge of the vast and heterogeneous array of underlying context information.

H. Range Subscription

The search functionality described in the previous section is extended to support subscriptions. The subscription functionality moves away from previous approaches in context scenarios where a subscription was made to an entity. Here, a subscription is made to a region of interest as show in Figure 6, which is defined by the upper and lower bounds of the range query. The algorithm functions almost identically to the searching, with the exception that along with retuning the presentities matching the search query, a subscription is constructed where where any entity whose changes in context state effects a change in its position within this region of interest is sent to the subscribing/querying node.

V. Results

We simulated our approach using the Free Pastry variant of the Pastry DHT [24]. The Java-based API contains a simulation environment permitting us to create the numbers of nodes required to test our implementation. The lack of sufficient quantities of context information was addressed by using the constituent values of the RGB and CMYK colour scales as context information with each scale being considered a *context range*.

For the verification of the range query, the RGB scale was used as a range with each colour representing a range state. We created, indexed and persisted a number presentities with these states. A random colour was chosen and range query performed between an upper and lower colour. A range query was executed and by querying for range of colours from a given shades of colours from a given point regions of the RGB scale we compared expected result sets. The results are shown in Table I.

We verified the advantages of organizing context entities over their states as well as the ordered persistence of these states advantages of context state by observing the number of messages required to resolve a query. As

# Presentity States	# Queries	Accuracy
2000	250	98%
2000	2000	99%
100	250	100%

Table I
Range Query Verification

queries are not randomly flooded but rather propagated around the DHTs ring like structure, the number of messages required should increase as as a function of the range increase. For this, we created ranges as described earlier and executed searches gradually increasing the range with each iteration.

We observed an increase in the number of messages required to resolve a query as the range of the query increased. As queries are propagated exponentially with respects to range along the search path, the resulting graph shown in Figure 7 verifies the ordering of the context entities as a factor of the affinities of their underlying context states. A random distribution would have produce random numbers of query messages.

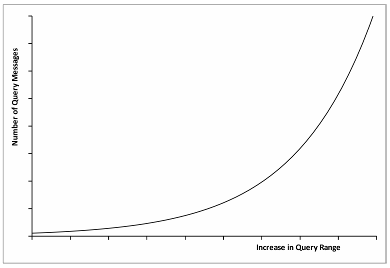

Figure 7. Range / Message Increases

The range subscription removed the need to subscribe to entities directly or issues relating to new entities arising after a search has been completed. Context provisioning approaches such as the one described by Kanter et. al. [4] maintain information on all subscriber - publisher relationships. Maintaining subscriptions for an area of interest N, with presentities P, each with S number of subscribers would maintain

$$\sum_{P \in N} (P \cdot S_P)$$

subscriptions. However, in our implementation, we are only interested in the number of subscribers to the area resulting in the number of subscriptions being equal to the number of subscribers, S.

VI. Conclusion & Discussion

Ubiquitous applications require access to relevant context entities in order to accurately provision the services demanded by end users. Approaches to provisioning context information have not satisfactorily addressed the need to model context entities with respect to their relations, enabling a more natural way to resource discovery.

In this paper, we presented an approach for indexing context entities relative to their current context states and exploiting this indexing information for searching and subscribing. Context information is categorised into context ranges and each presentity is indexed both within the context range and within an application space; a collection of context ranges. Indexing is achieved by modelling each range as a *n-dimensional* hypercube and calculating the Compact Hilbert Index for each state. This is persisted on an order preserving overlay, in order to maintain the locality achieved by the space filling curve. With this, we are able to query a context network relative to the affinity observed between any two entities using a range query algorithm.

We have shown that this approach is capable of finding relevant entities based on their context higher level context relationships and with respects to the requirements of an application or service. We implemented an approach that permits subscription to an interest area relative to a context state and reduce the need to discover and subscribe to all entities within a context network or to perform continuous searches.

Our solution however, is dependent on the Hilbert Curve as the means of realising our context relational model. While this permits us to motivate research into this area, applications and services will require more intuitive relational search and discovery. Future work therefore mandates more dynamic and user driven definitions of context relations, measures of proximity or affinity and efficient algorithms for constructing dynamic context-relational networks.

References

[1] A. K. Dey, "Understanding and Using Context," *Personal and Ubiquitous Computing*, vol. 5, no. 1, pp. 4–7, Feb. 2001.

[2] H. Sundmaeker, P. Guillemin, P. Friess, and S. Woelfflé, "Vision and Challenges for Realising the Internet of Things," in *Cluster of European Research Projects on the Internet of Things (CERP-IoT)*, no. March, 2010, pp. 43–67.

[3] J. Lee, J. Song, H. Kim, J. Choi, and M. Yun, "A User-Centered Approach for Ubiquitous Service Evaluation: An Evaluation Metrics Focused on Human-System Interaction Capability," in *Computer-Human Interaction*, ser. Lecture Notes in Computer Science, S. Lee, H. Choo, S. Ha, and I. C. Shin, Eds., vol. 5068. Berlin, Heidelberg: Springer, 2008, pp. 21–29.

[4] T. Kanter, S. Pettersson, S. Forsstrom, V. Kardeby, R. Norling, J. Walters, and P. Osterberg, "Distributed context support for ubiquitous mobile awareness services," in *2009 Fourth International Conference on Communications and Networking in China*. IEEE, Aug. 2009, pp. 1–5.

[5] R. Baloch and N. Crespi, "Addressing context dependency using profile context in overlay networks," in *Consumer Communications and Networking Conference (CCNC), 2010 7th IEEE*. IEEE, Jan. 2010, pp. 1–5.

[6] Y. Zhu, S. Ye, and X. Li, "Distributed PageRank computation based on iterative aggregation-disaggregation methods," in *Proceedings of the 14th ACM international conference on Information and knowledge management*. New York, New York, USA: ACM, 2005, pp. 578–585.

[7] B. Zhao, L. Huang, J. Stribling, S. Rhea, A. Joseph, and J. Kubiatowicz, "Tapestry: A resilient global-scale overlay for service deployment," *Selected Areas in Communications, IEEE Journal on*, vol. 22, no. 1, pp. 41–53, Jan. 2004.

[8] A. Kansal, S. Nath, J. Liu, and F. Zhao, "Senseweb: An infrastructure for shared sensing," *IEEE MultiMedia*, vol. 14, no. 4, pp. 8–13, Oct. 2007.

[9] C. Gonzalo, "3G IP Multimedia Subsystem (IMS): Merging the Internet and the Cellular World," p. 381, 2005.

[10] J. Walters, T. Kanter, and E. Savioli, "A Distributed Framework for Organizing an Internet of Things," in *The 3rd International ICST Conference on Mobile Lightweight Wireless Systems*, Bilbao, 2011, pp. 1–17.

[11] T. Kanter, S. Pettersson, S. Forsström, V. Kardeby, and P. \\"Osterberg, "Ubiquitous mobile awareness from sensor networks," in *Mobile Wireless Middleware, Operating Systems, and Applications-Workshops*. Springer, 2009, pp. 147–150.

[12] B. Schilit and N. Adams, "Context-aware computing applications," *Systems and Applications, 1994.*, pp. 85–90, 1994.

[13] H. Christein, *Distributed Communities on the Web*, ser. Lecture Notes in Computer Science, J. Plaice, P. G. Kropf, P. Schulthess, and J. Slonim, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, Dec. 2002, vol. 2468.

[14] R. Schmohl, "The Contextual Map," *deposit.ddb.de*, 2010.

[15] G. Peano, "Sur une courbe, qui remplit toute une aire plane," *Mathematische Annalen*, vol. 36, no. 1, pp. 157–160, Mar. 1890.

[16] J. Alber and R. Niedermeier, "On Multidimensional Curves with Hilbert Property," *Theory of Computing Systems*, vol. 33, no. 4, pp. 295–312, Jun. 2000.

[17] C. H. Hamilton and A. Rau-Chaplin, "Compact Hilbert indices: Space-filling curves for domains with unequal side lengths," *Information Processing Letters*, vol. 105, no. 5, pp. 155–163, Feb. 2008.

[18] R. Schmohl and U. Baumgarten, "The Contextual Map-A Context Model for Detecting Affinity between Contexts," *MobileWireless Middleware, Operating Systems, and Applications*, pp. 171–184, 2009.

[19] A. Padovitz, S. Loke, and A. Zaslavsky, "Towards a theory of context spaces," in *Pervasive Computing and Communications Workshops, 2004. Proceedings of the Second IEEE Annual Conference on*, no. March. IEEE, 2004, pp. 38–42.

[20] M. Knoll, "A P2P-Framework for Context-based Information," *1st International Workshop on Requirements and*, 2006.

[21] A. Schmidt, M. Beigl, and H. Gellersen, "There is more to context than location," *Computers & Graphics*, vol. 23, no. 6, pp. 893–901, 1999.

[22] A. Padovitz, "Context Management and Reasoning about Situations in Pervasive Computing," Doctoral Thesis, Monash University, 2006.

[23] D. Eastlake 3rd and P. Jones, "US Secure Hash Algorithm 1 (SHA1)," 2001.

[24] A. Rowstron and P. Druschel, "Pastry: Scalable, distributed object location and routing for large-scale peer-to-peer systems," *Design*, 2001.