Design of a Scalable and Expressive Content Naming System  
Using CAN Routing Algorithm  
Nguyen HOAISON\textsuperscript{1}, Hiroyuki MORIKAWA\textsuperscript{2}, Tomonori AOYAMA\textsuperscript{1}  
\textsuperscript{1} Graduate School of Information Science and Technology  
\textsuperscript{2} Graduate School of Frontier Sciences, The University of Tokyo  

1 Introduction  
With the rising up of ubiquitous computing, the design of a scalable and expressive content naming system will become essential\cite{1}. Several researches \cite{2,3} have proposed the use of attribute-value based content naming and DHT-based routing for such content naming system. However, because original DHT-based routing can achieve only exact query, realization of partial query and range query is still a challenge.

This paper introduces our design of a scalable and expressive content naming system as an overlay network of content name servers (CNS). Our system names a content by a set of attribute-value pairs and performs message routing by CAN routing algorithm \cite{3}. Especially, by the use of a novel matching scheme that matches the content name to CAN’s virtual coordinate space such that the content’s coordinate reflects the relationship between attribute and value, our system can effectively achieve the realization of partial query and range query.  

2 System Design  
2.1 CAN routing algorithm  
We utilize CAN routing algorithm for message routing in the overlay network. In CAN, a virtual d-dimensional Cartesian coordinate space is partitioned into hyper-rectangles, called zones. Each CNS is responsible for a zone, and is identified by the boundaries of its zone. They also maintain a coordinate routing table that holds the address and virtual coordinate zone of its neighbors. Using its neighbor coordinate set, a node routes a CAN message towards its destination by simple greedy forwarding to the neighbor with coordinates closest to the destination coordinates.

A content name is hashed to a content ID and then deterministically mapped onto a point P in the coordinate space. The data associated with the content name is then stored at the CNS that owns the zone within which the point P lies. When a client performs a query, it matches the content name to a name ID. By the use of the content ID as the destination coordinates, the query message is routed to the appropriate CNS. The content server looks up its database for the content name and returns data to the query client.

2.2 Content name space  
Our system utilizes a content name space for content naming and a content ID space for message routing in the overlay network. A content name is identified by a set of attribute-value pairs such as \((\text{attr}_1 : \text{val}_1), (\text{attr}_2 : \text{val}_2), ..., (\text{attr}_N : \text{val}_N))\). A content ID as d-dimensional coordinates is identified by d dimensions, each has m bit value.

We propose a novel matching scheme between the content name space and the content ID space as follows. The value and the attribute in each attribute-value pair \((\text{attr}_i : \text{val}_i)\) are respectively hashed to a m-bit value ID \(H(\text{val}_i)\) and an attribute ID \(H(\text{attr}_i)\) which ranges from 1 to d. The content ID is produced from value IDs such that for each attribute-value pair, the attribute ID decides the dimension’s order number of the value ID. For instance, in the case of attribute-value pair \((\text{attr}_i : \text{val}_i)\), \(H(\text{val}_i)\) will be the value of the \(H(\text{attr}_i)\)th dimension. A default value \((\text{e.g. } 0)\) is assigned to the value of the dimensions which are not matched with any value. Figure 1 shows our matching scheme.

Furthermore, in order to ease the realization of range query, our system hashes a numerical value to a value ID by a locality preserving hashing function, which is defined as if \(\text{val}_{i} > \text{val}_{j}\) then \(H(\text{val}_{i}) > H(\text{val}_{j})\).

If there are multiple attributes that are hashed to the same attribute ID, the content name is matched to multiple name IDs, each of them includes the value ID corresponding to each attribute. In this case, the data associated with the content name is distributed to a set of CNSs.

2.3 Partial query and Range query  
Our system achieves partial query and range query by broadcasting query message along dimensions that have indeterministic value. The partial query queries the data whose content name contains all attribute-value pairs as in query message. In this case the content ID in the query message is different to the content ID of queried data in the dimensions that have default value ID. To achieve partial query, after routed to the target CNS by its included content ID, the query message will be broadcasted along the dimensions that have default value ID in its included content ID. A wildcard can be used in query message to explicitly describe dimensions that have indeterminate value. In this case, the query message is broadcasted along only dimensions that have wildcard value.

In the case of the range query defined as the one that queries a range of an attribute’s value, the range query is matched to a range of value ID. The query message is forwarded to the target CNS’s neighbors which are responsible to value ID in the queried range. The property of the locality preserving hashing function guarantees the query to be resolved properly. If the client performs the range query over several attribute-value pairs, the message will be forwarded along dimensions corresponding to the range-queried attributes.

3 Conclusions and future work  
Our content naming system realizes the expressive and scalable content name service by the use of attribute-value based content name and CAN-based message routing. Furthermore, our novel matching scheme between content name and content ID can keep the relationship between attribute and value and therefore allows our system to realize partial query and range query effectively. In the future, we will demonstrate the effectiveness of our system by simulation and implementation.

References  
\cite{2} J. Gao and P. Steenkiste ”Design and Evaluation of a Distributed Scalable Content Discovery System”, Journal on Selected Areas in Communications, January, 2004  
\cite{3} M. Balazinska et. al “INS/Twine: A Scalable Peer-to-Peer Architecture for Intentional Resource Discovery”, International Conference on Pervasive Computing, August 2002  
\cite{4} S. Ratnasamy, et. al ”A Scalable Content-Addressable Network”, ACM SIGCOMM’01, August 2001