A Survey on the Resource Space Model

Jingqiang Chen

Key Lab of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Science
Beijing 100080, People's Republic of China
cjq3305@gmail.com

***Abstract*— This paper surveys research on the Resource Space Model RSM. RSM is a classification-based, multi-dimensional and content-based space model for efficiently and effectively managing various resources. As a non-relational data model, it has a rather complete theoretical basis and has significant applications in faceted search and the future cyber-physical society. Applications in picture resources and email resources management are introduced.**

I. Introduction

A Resource Space Model is a multidimensional classification space where dimensions are discrete and hierarchical that may have no linear ordering relationships. A resource space can be normalized to increase the correctness of resource management by setting constraints on dimensions. Different from traditional relational database which is attribute-based, the Resource Space Model is classification-based.

Unlike file systems of operating systems and current keyword-based search engines, the Resource Space Model is multi-dimensional. With a Resource Space Model, resources are classified by various classification methods. Previous classification methods only focus of the accuracy of classifications but how to effectively manage classifications are neglected.

Fig. 1 displays a 2-dimension resource space of books in a library. Note that the coordinate of *Maths* on the category dimension is hierarchical, which can further be classified as *High-Maths*, *Linear-Maths* and *Discrete-Maths*. As a matter of fact, any coordinate can be refined if needed. Such characteristic is fairly significant to the Resource Space Model, and leads to generalization and specialization characteristic of the Resource Space Model. Furthermore, there may be no linear ordering among coordinates on each dimension. Consequently, the Resource Space Model is quite distinguishing from other multi-dimensional technologies like Data Warehouse. The point in Fig. 1 represents a list of books that belong to the category of *Computer Science* and the publisher is *Springer*. That is to say, each dimension, coordinate and point in the Resource Space Model represent a resource set.

Its fundamental theory and methodology, including the definition, normal forms, possible operations on it and storage mechanism [1][2][3] have been founded. A Probabilistic Resource Space Model has been proposed for dealing with uncertain contents [4]. In the following, the foundation of RSM is briefly introduced.

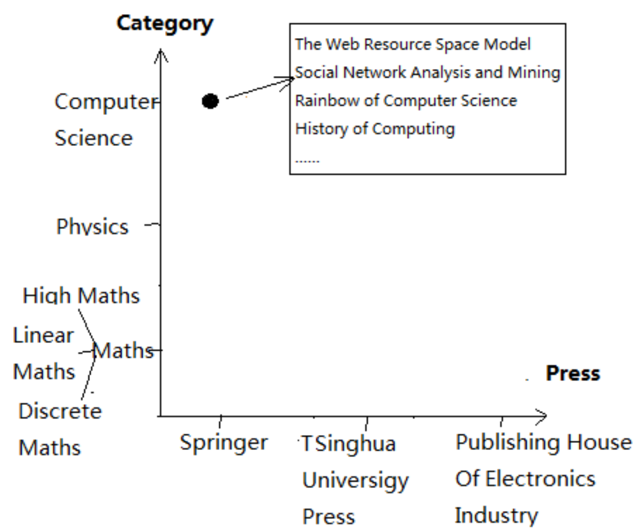

Fig. 1 A 2-dimension resource space representing books in a library

II. Foundation of Resource Space Model

A. *Definition of The Resource Space Model*

A resource space is an n-dimensional space where every point uniquely locates a set (possibly null) of related resources. A resource space has a name, a type, a logical location, and an access privilege. It is formally defined as follows:

1) A resource space is represented by RS $(X_1, X_2, \ldots, X_n)$ or RS for short. RS is the name of the space and X_i is the name of an axis, |RS| denotes the number of dimensions of RS.
2) $X_i = \{C_{i1}, C_{i2}, \ldots, C_{im}\}$ represents an axis with its coordinates. Each element denotes a coordinate name in the form of a noun or a noun phrase.
3) Any coordinate C_i can be denoted as $<V_i, E_i>$ which means that the structure of coordinate Ci is hierarchical.
4) Let $R(RS)$, $R(X_i)$, $R(C_{ij})$ denote resource set represented by the resource space RS, the dimension X_i, the coordinate C_{ij} respectively. Then $R(RS) \supseteq R(X_i) \supseteq R(C_{ij})$, $R(RS)= \bigcup_{i=1}^{n} R(Xi)$,herein n denotes the number of dimensions in RS. $R(X_i)= \bigcup_{j=1}^{m} R(Cij)$ where m denotes the number of coordinates of the dimension X_i.

B. Normal Forms of The Resource Space Model

Just as the Relational Database has 1NF, 2NF, 3NF, BCNF, 4NF and 5NF, the RSM has the following four types of normal forms:

1) A 1NF resource space is a resource space and there does not exist name duplication (semantic overlap) between coordinates at any axis.
2) A 2NF resource space first satisfies 1NF, and for any axis, any two coordinates are independent from one another.
3) A 3NF resource space first satisfies 2NF, and $R(X_i) = R(X_j)$, where $i \neq j$.
4) A 4NF resource space is first a 3NF resource space and there are no null points in the Resource Space.

C. Operations on RSM

There four basic types of operations defined for the Resource Space Model:

1) Join

If two spaces RS1 and RS2 hold the same type of resources and they have n ($\geqslant 1$) axes in common, then they can be joined as one RS such that RS1 and RS2 share these n common axes and |RS| = |RS1| + |RS2| - n. RS is called the join of RS1 and RS2 and is denoted by RS1 • RS2 => RS.

If RS1 and RS2 are union-compatible then R(RS) = R(RS1) ∪ R(RS2). If RS1 and RS2 are not union-compatible then R(RS) = R(RS1) ∩ R(RS2). For any point $P(X_1, X_2, \ldots, X_n, Y_1, Y_2, \ldots, Y_t, Z_1, Z_2, \ldots, Z_m)$, P is non-null if and only if both $P1(X_1, X_2, \ldots, X_n, Y_1, Y_2, \ldots, Y_t)$ in RS1 and $P2(Y_1, Y_2, \ldots, Y_t, Z_1, Z_2, \ldots, Z_m)$ are non-null. If RS1 and RS2 are union-compatible then R(P)=R(P1)∪R(P2), else R(P)=R(P1) ∩ R(P2).

2) Disjoin

A space RS can be separated into two spaces RS1 and RS2 (denoted by RS => RS1 • RS2) such that they have n ($1 \leqq n \leqq$ minimum(|RS1|, |RS2|)) axes in common, |RS| - n different axes, and |RS1| + |RS2| = |RS| + n.

For any point $P1(X_1, X_2, \ldots, X_n, Y_1, Y_2, \ldots, Y_t)$ in RS1, P1 is non-null if there exist any non-null point $P(X_1, X_2, \ldots, X_n, Y_1, Y_2, \ldots, Y_t, Z_1, Z_2, \ldots, Z_m)$ in RS. The special operation Projection can be derived from Disjoin operation.

3) Merge

If two spaces RS1 and RS2 hold the same type of resources and satisfy: 1) |RS1| = |RS2| = n, 2) they have n-1 common axes, and 3) the distinct axes X_1 and X_2 can be merge into one axis X such that $X=X_1 \cup X_2$ and $R(X) = R(X_1) \cup R(X_2)$. Then RS1 and RS2 can be merged into one RS by retaining the n-1 common axes and adding a new axis X. RS is called the merge of RS1 and RS2, denoted by RS1∪RS2=>RS, and |RS|=n.

For Merge operation, R(RS)=R(RS1)∪R(RS2), and for any common Y in RS, suppose the corresponding axis in RS1 and RS2 be Y_1 and Y_2 respectively, then $R(Y)=R(Y_1)\cup R(Y_2)$. For uncommon axis X in RS, $R(X)=R(X_1)\cup R(X_2)$.

4) Split

A resource space RS can be split into two spaces RS1 and RS2 that hold the same type of resources as that of RS and have |RS|-1 common axes, by splitting axis into two axes X' and X'' such that $X=X' \cup X''$. This split operation is denoted by RS=>RS1∪RS2. For Split operation, the resource set of RS is also split into two independent parts according to X' and X''.

The above four operations are RSM-specific operations, all of which keep 1NF, 2NF and 3NF of the original resource space. Some other operations, for example, set-related operations like union, intersection and difference can also be applied to RSM. The completeness of operations is also discussed [3].

D. Storage Mechanism

To store resource space is to find an appropriate way to index multidimensional and hierarchical structure so as to support various queries. There are multiple ways to implement the storage of a resource space.

1) Making use of relational tables

Relational tables can be used to store resource space. Let a relational table corresponds to a resource space, and each attributes of the relational table represents an axis of the resource space [5]. Then, each resource in the resource space can be represented as a tuple of the table. In this case, Resources belonging to the same point in the resource space will have the same attribute values in the relational table. The management efficiency depends on that of the relational database.

The defect is that the hierarchical structure of the coordinates of the resource space will be lost in this way. Furthermore, the normal forms of Relational Database and Resource Space Model is quite different from each other.

2) Making use of XML files

Another feasible approach to storing resource space is making using of XML. Let the name of the resource space corresponds to the root node of the XML file, and each axis corresponds to one child node of the root node. In this way, the whole structure of the resource space can be exactly preserved by XML.

Each resource has the same number of copies in XML tree as the dimensionality of resource space. Such a redundancy requires an additional cost of integrity maintenance.

3) Making use of spatial indexing structures

The spatial indexing structures like RTree [6] are also considerable approaches to storing resource space. However, most spatial indexing structures require that each dimension has a linear ordering of its coordinates, but RSM are discrete and usually have hierarchical semantic relationships rather than linear order. Such that to appropriately make use of RTree, a linear order among sibling coordinates of axes must be predefined. An encoding technique named bitstring is introduced to properly encode dimensions and coordinates of the resource space [3][7].

The difficulty is how to predefine the linear order despite there are indeed no such relationships between coordinates. One proper way is to compute the semantic distance.

4) Making use of P2P mechanism

This is useful to deploy a resource space onto a peer-to-peer network for decentralized applications [11][12][15][16][19]. Several proposals have been proposed, e.g., in RSM-Based Gossip on P2P network. The structured and unstructured P2P resource space systems have been introduced [3].

As a part of the model, the design method of the resource space and the fuzzy resource space were proposed [17][18].

III. DISCUSSION

As mentioned above, RSM is a classification-based space model. The first issue is how to appropriately establish dimensions. Every dimension of the Resource Space Model represents some type of classification method. Dimensions of RSM are somewhat like attributes of relational tables. The difference between them is that dimensions represent sets of resources while attributes only reflect some surface properties of an individual resource. In more detail, dimensions are set-based and attributes are individual-based. In some ways, attributes could be used as dimensions changing the meaning of attributes such that they never merely represent some property of an individual resource but a set of resources.

Dimensions can be internal or external. By internal, the dimensions are based on contents of resources while by external the dimensions are built from external properties of resources. Take the above book resources for example. The press dimension is an external dimension while the category dimension is an internal dimension. Internal dimensions can be derived from the contents of resources. However, external dimensions can only be specified by users, resource owners or domain experts.

For content-based dimensions, the existing clustering methods and classification methods could be used to automatically establish dimensions. Since they are hierarchical, some hierarchical clustering algorithms like Agglomerative Hierarchical Clustering Algorithm can be employed. However, dimensions built in this way should be proved that they will make sense.

IV. APPLICATIONS

A. *A Picture Resource Space*

Fig. 2 shows an example of using a 3-dimensional resource space (*Category*, *Place*, *Format*) to organize picture resources. Here *Format* is an external property of pictures while other two are content-based dimensions.

Usually, we store pictures in the file system which can be regarded as a 1-dimensional resource. With multidimensional picture resource space model, the efficiency of managing and locating pictures will be greatly improved. For example, the point(Living Things, China, jpg) in the resource space represents all pictures of jpg format about living things in China. Specially, the Living Things coordinate on Category dimension can further be refined as Human beings, Animals and Plants. If needed, one more dimension like Time denoting when the picture is taken can be added to the RSM Schema. Users can specialize or generalize the search result through hierarchical coordinates.

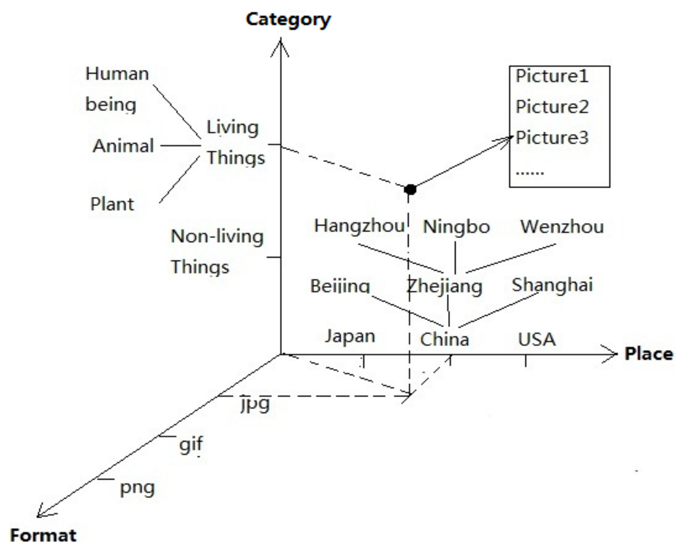

Fig. 2 A resource space model of pictures

The specific point shown in the resource space represents a set of picture indices pointing where the corresponding pictures store. Also, other information like brief description about the picture can be combined with each index entry, which makes it more efficient to retrieve resources.

The picture resource space model is in 3NF. When adding a new picture, we need to specify its category, format and place. We need some convenient approach to adding new picture resource efficiently. Thus we should refer to the autonomy of RSM.

However, as mentioned, it is a hard to mine useful information from an image, which makes it difficult to add a new picture into the resource space automatically. One feasible way to represent a picture with keywords, which are specified by users or obtained from the web tags.

In addition, to discover semantic relationships between pictures is an interesting issue remained to be addressed.

B. *An Email Resource Space*

Every day we send and receive a great number of emails and the amount increases sharply with time. How to efficiently manage the amount of emails becomes an issue to be addressed.

Fig. 3 shows an email resource space model, the dimensions of which are Topic, Contactor, State and Time. Topic and Time are hierarchical dimensions. All of these four dimensions are orthogonal with each other, i.e. R(Topic)=R(Contactor)=R(State)=R(Time). Each resource could be got from any of the dimensions. The email resource space model is in 3NF.

When a new email is added to the resource space, its corresponding coordinates on each axis should be specified. There are two ways to get the right coordinates if no proper coordinate can be found in the dimensions. The one is to put some coordinate representing all other resources that are not represented by the rest coordinates of the dimension, like the coordinate *Other*. In this way, when resources belonging to the *Other* coordinate accumulate to some degree, we should redistribute them to the rest coordinates or create new coordinates if needed. The other approach is to create new coordinate when the newly added resource doesn't belong to any existing coordinates.

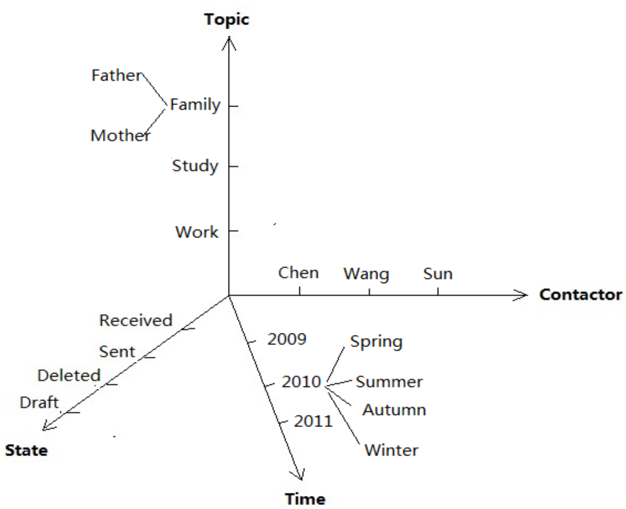

Fig. 3 An email resource space.

Since most emails are text-based, existing text mining approaches can be applied to classify emails. Usually, the title of an email contains more useful information than the subject, so sometimes we can merely use information extract from the titles to represent emails. One efficient approach is to use keywords which can represent either coordinates or emails. The similarity between them is measured through cosine similarity or Euclid distance. When a new email issues, its keywords are mined first and then compute the similarity between coordinates and the new email.

Emails can also be automatically added into the email resource space through semantic relationships between emails. For example, it is reasonable that emails are categorized into two types: initiator which initiates a session and responder which responds to the initiating emails. Responding emails have strong relation with initiating emails, so they should be in the same category.

In some time interval, emails to or from the same contactor are certainly on the same topic. This characteristic could also be used when adding a new email.

An email managing system based on RSM can be designed; it may consist of the following four parts:

1) A RSM management system including the storage mechanism and management mechanism.
2) An email system responsible for sending and receiving emails.
3) An information mining mechanism that extracts key information from emails.
4) A friendly user interface that helps users conveniently manage emails.

V. SUMMARY

The Resource Space Model is to address the issue of how to efficiently build an appropriate classification space for the resources of an application domain to normalize the space, and to enable users to easily operate the space to manage the contents of various resources.

Towards the future interconnection environment ideal [20], a complex semantic space is proposed by integrating the RSM with the semantic link network to make use of both advantages [7, 10]. It is also used as the model to simulate the mental space and semantic image [8, 9].

REFERENCES

[1] H.Zhuge, The Knowledge Grid, *World Scientific Publishing Co.*, Singapore, 2004.
[2] H.Zhuge, Resource Space Model, Its Design Method and Applications, *Journal of Systems and Software*, 72 (1) (2004) 71-81 .
[3] H.Zhuge, The Web Resource Space Model, *Springer*, 2008.
[4] H.Zhuge and Y.Xing, Probabilistic Resource Space Model for Managing Resources in Cyber-Physical Society, *IEEE Transactions on Service Computing*, http://doi.ieeecomputersociety.org/10.1109/TSC.2011.12.
[5] H.Zhuge, Y.Xing and P.Shi, Resource Space Model, OWL and Database: Mapping and Integration, *ACM Transactions on Internet Technology*, 8/4, 2008.
[6] V.Gaede, and O.Gnther, Multidimensional Access Methods. *ACM Computing Surveys*, 30(2)(1998)170-231.
[7] H.Zhuge, The Complex Semantic Space Model, Keynote at 20th IEEE International Conference on Collaboration Technologies and Infrastructures, June 27th -29th , 2011, Paris, France.
[8] H.Zhuge, Interactive Semantics, *Artificial Intelligence*, 174(2010)190-204.
[9] H.Zhuge, Semantic linking through spaces for cyber-physical-socio intelligence: A methodology, *Artificial Intelligence*, 175(2011)988-1019.
[10] H.Zhuge, Communities and Emerging Semantics in Semantic Link Network: Discovery and Learning, *IEEE Transactions on Knowledge and Data Engineering*, vol.21, no.6, 2009, pp. 785-799.
[11] H.Zhuge and X.Sun, A Virtual Ring Method for Building Small-World Structured P2P Overlays, *IEEE Transactions on Knowledge and Data Engineering*, 20 (12) (2008)1712-1725.
[12] H.Zhuge and J.Zhang, Topological Centrality and Its Applications, *Journal of the American Society for Information Science and Technology*, 61(9)(2010)1824-1841.
[13] H.Zhuge and L.Feng, Distributed Suffix Tree Overlay for Peer-to-Peer Search, *IEEE Transactions on Knowledge and Data Engineering*, 20 (2) (2008) 276-285.
[14] H.Zhuge and X.Li, Peer-to-Peer in Metric Space and Semantic Space, *IEEE Transactions on Knowledge and Data Engineering*, 19 (6) (2007) 759-771.
[15] H.Zhuge, X.Chen, X.Sun and E.Yao, HRing: A Structured P2P Overlay Based on Harmonic Series, *IEEE Transactions on Parallel and Distributed Systems*, 19 (2) (2008) 145-158.
[16] H.Zhuge, et al, A Scalable P2P Platform for the Knowledge Grid, IEEE Transactions on Knowledge and Data Engineering, 17 (12) (2005) 1721-1736.
[17] H. Zhuge, Resource space model, its design method and applications. *Journal of Systems and Software, 72*(1) (2004)71-81.
[18] H. Zhuge, Fuzzy resource space model and platform. *Journal of Systems and Software, 73*(3) (2004)389-396.
[19] H. Zhuge and X. Li, RSM-Based Gossip on P2P Network. ICA3PP 2007: 1-12.
[20] H.Zhuge, The Future Interconnection Environment, *IEEE Computer*, 38 (4) (2005) 27-33.