Cover Page

16th ICCRTS

"Collective C2 in Multinational Civil-Military Operations"

Title of Paper

Link Correlated Military Data for Better Decision Support

Topics

Topic 4: Information and Knowledge Exploitation
Topic 5: Collaboration, Shared Awareness, and Decision Making

Authors

Xin Jin
Xiaoman Wang
JingJing Yan
WeiTai Liang
(Science and Technology on Information Systems Engineering Laboratory,
Nanjing, Jiangsu, 210014, P.R. China)

Point of Contact

Xin Jin
JingJing Yan

Name of Organization

Science and Technology on Information Systems Engineering Laboratory

Address

Nanjing, Jiangsu, 210014, P.R. China

Email

jinxin28work@163.com
mooncailyjj@yahoo.com.cn

Link Correlated Military Data for Better Decision Support

Xin Jin, Xiaoman Wang, Jinjin Yan, Weitai Liang

(Science and Technology on Information System Engineering Laboratory,
Nanjing Research Institute of Electronics Engineering,
Nanjing 210007, China)

Abstract:

Near all military decisions need data-support. For nowadays warfare's complexity and vast use of sensors, decision data are usually of great amount, while what commanders require are usually just a small subset covering his interest. Sometimes it is distractive and time-consuming to lookup such a subset, because it contains data in different domains, under different topics, from different sources. In problem domain, theses data are correlated, while machine can not understand it and does not support correlation-based data lookup. In this paper, applying Linked Data technique, correlations among decision data are described in machine-understandable and process-effective way, and visualized on user interface as navigation links that can guide people to find related data. Methods are proposed on link-construction and data-lookup. It is proved through experiments that correlation-based data lookup method is a good complement to traditional tree-based one, which brings decision data support with higher problem relativity, user thinking aligning ability, and operation efficiency.

1 Introduction

Nowadays military decision has significant differences with past experiences. In the old ways, commanders must place themselves in the battle fields, judge situation by their own observation, and make decision upon their own knowledge. In modern ways, most commanders just sit in the commander's room, judge situation by collected sensor data, and make decision upon various kinds of knowledge data. With the prosperity of sensor technology, the ability of collecting data has improved greatly compared with the past. Currently, near all Command and Control system requires data as input, while in the future, sensors may substitute human eyes and ears, and data may probably become the main input of human brain systems while decision making.

Therefore, qualities of data - such as accuracy, relativity, redundancy, real-time performance, and so on, are quite important to a decision maker. Quality of data is decided by the performance of data support. From user view point, performance of data support may be roughly divided into two categories: good and bad. The bad data support is usually careless of users who need the data, careless of the problem which the data should be about, and simply provide all data available, probably organize them by categories, arrange them in "trees" to enable data browse, as shown in Fig. 1. By this support, user has to unfold the tree branch by branch till he finds the required data. He may be unfamiliar with the tree's architecture, and has to unfold every branch to learn about it firstly. So much effort may be wasted on data lookup, that when all required data are found, he may even forget what problem he was dealing with.

On the other side, good data support should be able to:

1) Focus on user concerned problem, and provide only data closely related with it;

2) Align with user's thinking process, learn his thinking habit, anticipate his requirement, and always provide useful data;
3) Provide data at near-zero delay, enabled by the availability of near-infinite processing and storage capacity at near-zero unit cost.

By this support, user can focus on problem solving with little concern distracted on data lookup.

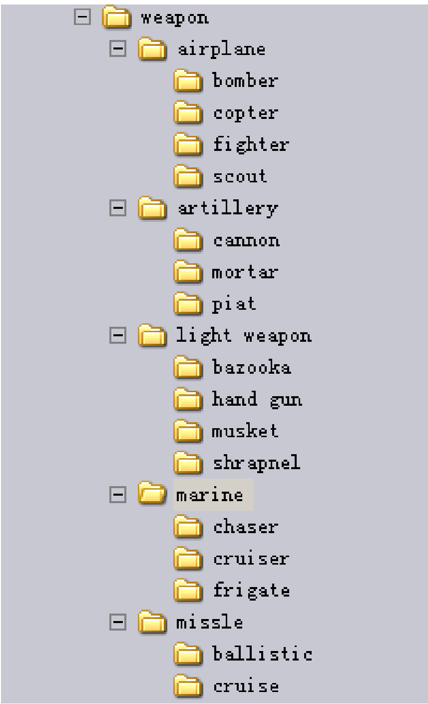


Fig. 1 Tree-based data browse pattern

In simple decision making cases, where the data amount is relatively small, bad decision support may be enough, because the effort paid on data lookup is relatively small, not quite noticeable by trained decision makers. However, when the complexity increases, it no longer fits:

1) Data amount increase: With cost brought down, sensors are vastly used, creating enormous data every day, costing more time on data analysis than in the past;
2) Data diversity increase: When dealing with a complex decision problem, required data may cover many areas, such as military, humanity, society, law, biology, physics, cyber space and so on.

As a result, sensor collected data have to be classified into more categories and subcategories, making the data "tree" wider and deeper. Consequently, time cost on data lookup within the tree increases. Comparatively, however, user required data does not increase obviously. They only care about those data that could help them to make right decision. Therefore, it becomes harder and harder to lookup required data from such an increasingly huge "tree".

Our method is by utilizing data correlations. Everyone knows that data are correlated with each other, especially within a problem domain. The correlations act as a kind of clew that may lead commanders to right decisions. If machine understands data correlations, it may act as a guider rather than simply a data provider. For instance, when user finds a data "Organization A", he may immediately think about another one "Company B", because B provides weapons to A. At the same time, machine will list all data related with "Organization A" that can help people get all information about such organization, for example, who's the leader of the organization, how many members does it have, which places are they recently active in, which targets did they strike in recent activities, as well as from which companies their weapons are provided, as shown in Fig. 2. Then he may choose one of the companies, for example “Company B”, and the machine will list all data related with “Company B”, like its addresses, telephone numbers, and so on. In this way, machine can align with user’s thinking process, and provide data that may be helpful before he think about. No matter how much data have been collected by sensors, machine can always filter out most of them, left only those data closely related with the user currently focused one. This is a small but useful subset, which can help the user to think further.

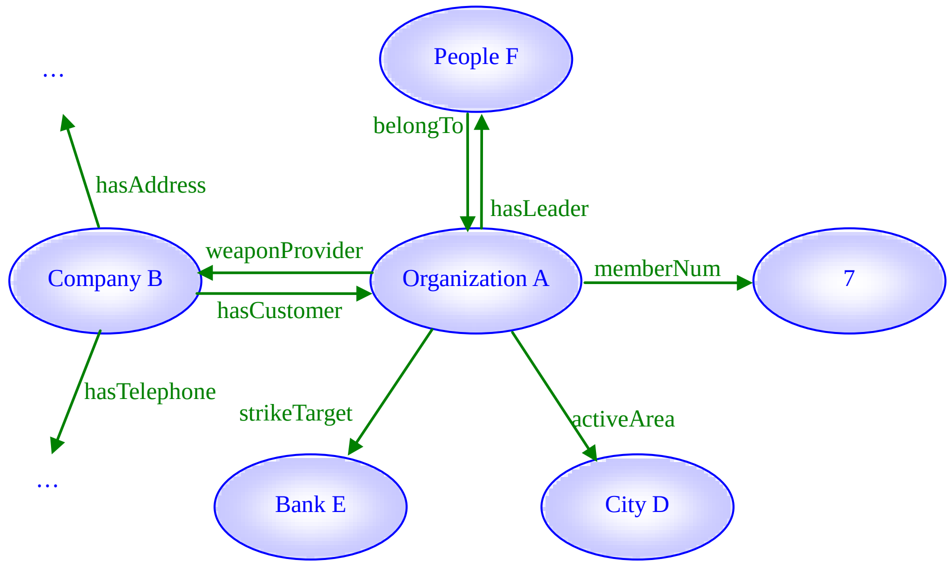


Fig. 2 Data correlations

Linked data [1] is a set of best practices for publishing and deploying instance and class data using the RDF data model, naming the data objects using uniform resource identifiers (URIs), thereby exposing the data for access via the HTTP protocol, while emphasizing data interconnections, interrelationships and context useful to both humans and machine agents. These best practices have been adopted by an increasing number of data providers over the last three years, leading to the creation of a global data space containing billions of assertions - the Web of Data. In our opinion, Linked Data technique may be applied to present data correlations into formal links, which is machine -understandable and process-effective, and may lead to realization of correlation-based data browse.

The remainder of this paper is structured as follows. In Section 2 we provide an overview of the related works. Linked Data technique is introduced in brief in Section 3. Section 4 describes our methods on link construction and data lookup. They are implemented and tested in Section 5, with performance analyzed in section 6. Section 7 makes conclusions.

2 Related Works

The most visible example of adoption and application of the Linked Data principles has been the Linking Open Data project [2], a grassroots community effort founded in January 2007 and supported by the W3C Semantic Web Education and Outreach Group [3]. The original and ongoing aim of the project is to bootstrap the Web of Data by identifying existing data sets that are available under open licenses, converting these to RDF according to the Linked Data principles, and publishing them on the Web.

Participants in the early stages of the project were primarily researchers and developers in university research labs and small companies. Since that time the project has grown considerably, to include significant involvement from large organizations such as the BBC, Thomson Reuters and the Library of Congress. This growth is enabled by the open nature of the project, where anyone can participate simply by publishing a data set according to the Linked Data principles and interlinking it with existing data sets.

An indication of the range and scale of the Web of Data originating from the Linking Open Data project is provided in Fig. 3. Each node in this cloud diagram represents a distinct data set published as Linked Data, while each arc indicate that links exist between items in the two connected data sets. The content of the cloud is diverse in nature, comprising data about geographic locations, people, companies, books, scientific publications, films, music, television and radio programs, genes, proteins, drugs and clinical trials, online communities, statistical data, census results, and reviews. According to statistics in May 2009, the Web of Data consists of 4.7 billion RDF triples, which are interlinked by around 142 million RDF links [4].

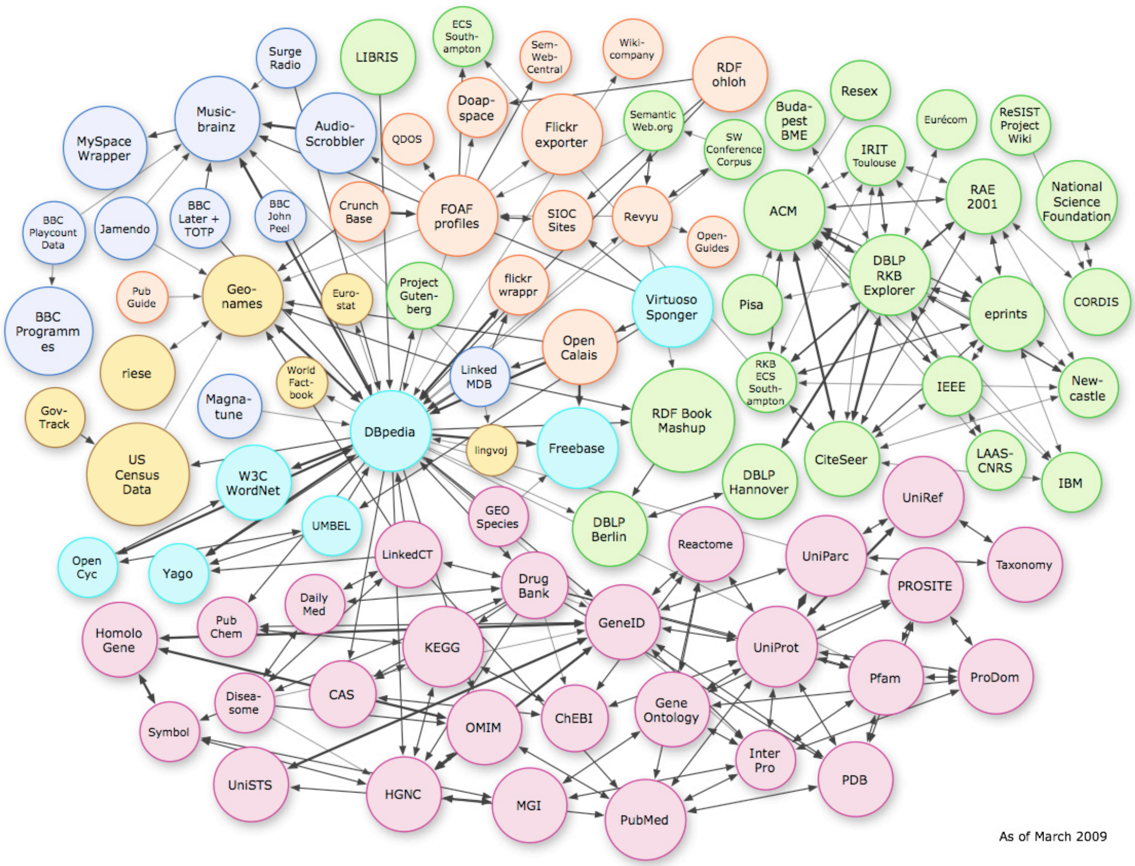


Fig. 3 Linked Open Data

Linked Data is a new technique, and little researches has been found till now on its application in military decision support domain. On the other hand, Semantic Web [5], proposed in 1999, long before Linked Data - while Linked Data is usually considered as part of Semantic Web, or “the Semantic Web done right” as described by Tim himself - has been applied into military domains such as Cooperative

Command and Control [6, 7], Situation Awareness Enhancement [8, 9], and Military Knowledge Base [10, 11].

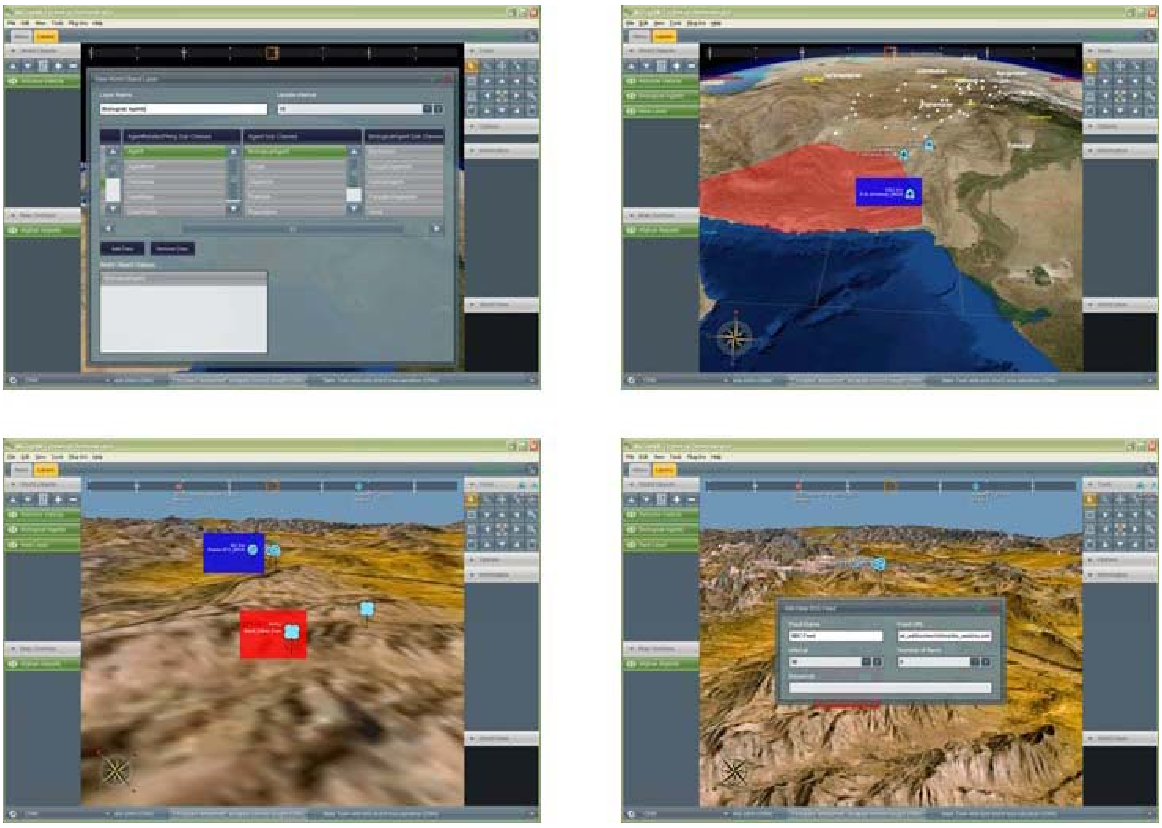


Fig. 4 AKTiveSA's user interface

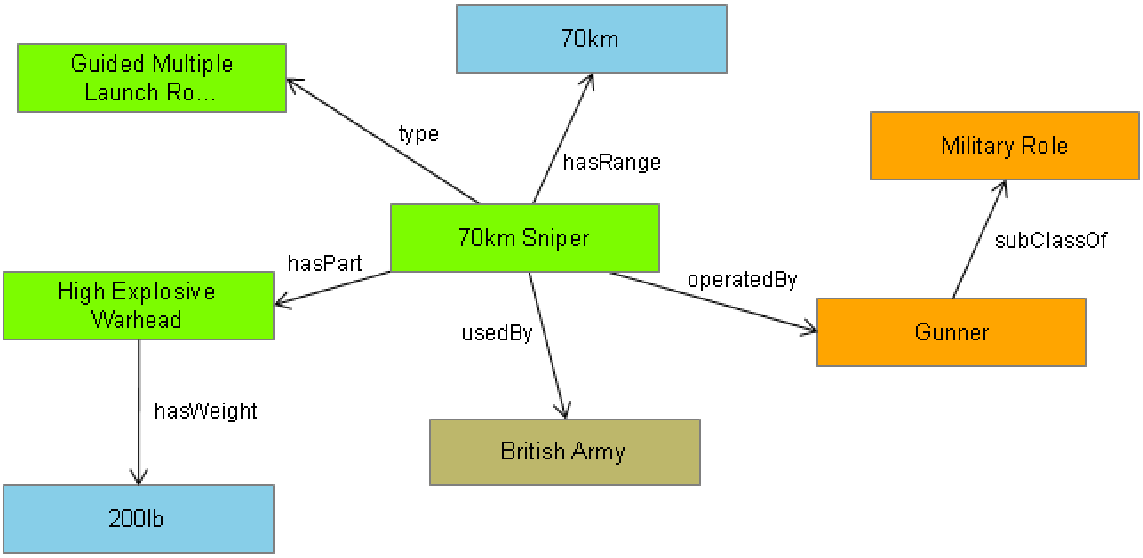


Fig. 5 AKTiveSA's linked data

Among them, AKTiveSA [12] is a successful attempt on situation awareness enhancement in military operational contexts other than war (MOOTW), specifically humanitarian assistance and disaster relief. Its principle is quite similar with Linked Data. On its user interface, as shown in Fig. 4, situation is shown as various elements distributed on a world map. On selection of one element, it can list all its attributes, while on selection of one attribute, it can list all its values or object elements. Some of its linked data are shown in Fig. 5. One of its deficiency lies in that all the attributes and links are designed and coded manually. No automatic mechanism on link construction was proposed. In our approach, some of the data correlations - especially those defined in relational databases - can be automatically translated into URI based links, thus can greatly reduce man power cost on software development.

3 Linked Data Technique

Tim Berners-Lee (2006) outlined a set of 'rules' for publishing data on the Web in a way that all published data becomes part of a single global data space [13]:

1) Use URIs as names for things;
2) Use HTTP URIs so that people can look up those names;
3) When someone looks up a URI, provide useful information, using the standards (RDF, SPARQL);
4) Include links to other URIs, so that they can discover more things.

These have become known as the “Linked Data principles”, and provide a basic recipe for publishing and connecting data using the infrastructure of the Web while adhering to its architecture and standards.

Linked Data relies on two technologies that are fundamental to the Web – URIs (Uniform Resource Identifiers) and HTTP (Hypertext Transfer Protocol). URIs provide a generic means to identify any entity that exists in the world. These entities can be looked up simply by dereferencing the URI over the HTTP protocol.

URIs and HTTP are supplemented by a technology that is critical to the Web of Data – RDF (Resource Description Framework). RDF provides a generic, graph-based data model with which to structure and link data that describes things in the world. The RDF model encodes data in the form of subject, predicate, object triples. The subject and object of a triple are both URIs that each identify a resource, or a URI and a string literal respectively. The predicate specifies how the subject and object are related, and is also represented by a URI.

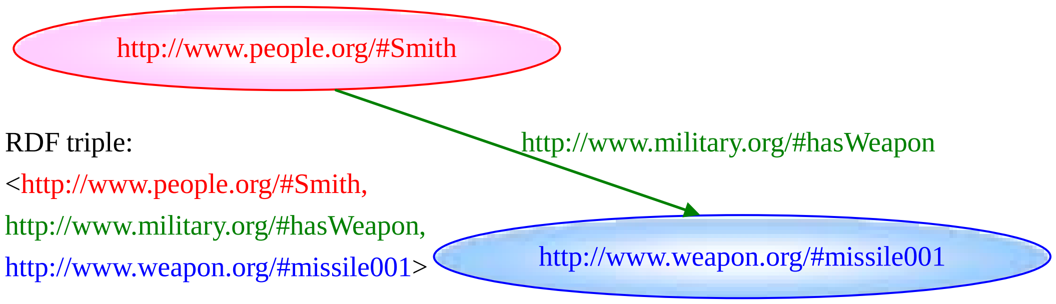


Fig. 6 Example RDF description

For example as shown in Fig. 6, an RDF triple can state that a people named “Smith”, and a weapon named “missile001”, each identified by a URI (with namespace added as a prefix such as “http://www.people.org”), are related by “hasWeapon”, means that Smith has a weapon called "missile001". Similarly, an RDF triple may relate a weapon "missile 001" to a weapon producer "Iraq" by "hasProducer", means that missile001 is produced in Iraq. Two resources linked in this fashion can be drawn from different data sets on the Web, allowing data in one data source to be linked to that in another, thereby creating a Web of Data.

Consequently it is possible to think of RDF triples that link items in different data sets as analogous to the hypertext links that tie together the Web of documents. RDF links take the form of RDF triples, where the subject of the triple is a URI reference in the namespace of one data set, while the object of the triple is a URI reference in the other.

4 Methods

In this section, we will introduce methods on automatic link construction based on relational database and how to look up data in the data net constructed by such links.

a) Link Construction

There is no standardized form for data correlations. In general, correlation between a couple of Data can be described as "Data1, Relation1, Data2", means "Data1" is related to "Data2" through "Relation1". If there is "Data2, Relation2, Data1", then "Relation1" and "Relation2" are a couple of bidirectional correlation. Not all data have bidirectional correlation, such as "Data3, hasValue, 123.45", etc. Correlations may have meaning - human defined semantics, so that data can be connected at conceptual level.

Within various kinds of correlations, relational database is a typical one. In a database, different tables are correlated by primary & foreign keys. Within a table, objects and values are correlated by attribute fields, as shown in Fig. 7. There are still more kinds of correlations hidden under data's lateral expression. For example, "010101 F14" can be related to "0101 Battleplane" through "SubClassOf", as deduced from human expression habits. However, this is far more complex than the former two kinds defined purely by database structure, thus is not considered in this paper.

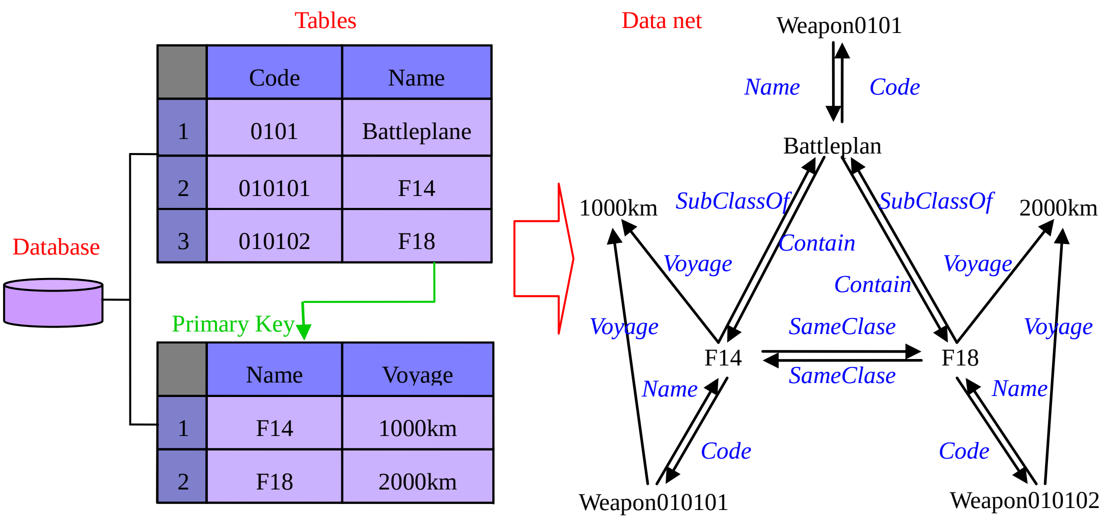


Fig. 7 Correlations in database

D2R Server [14] is a tool for publishing the content of relational databases on the Semantic Web, a global information space consisting of linked data. Data on the Semantic Web is modeled and represented in RDF. As shown in Fig. 8, D2R Server uses a customizable D2RQ mapping to map database content into this format, and allows the RDF data to be browsed and searched – the two main access paradigms to the Semantic Web. D2R Server's Linked Data interface makes RDF descriptions of individual resources available over the HTTP protocol. An RDF description can be retrieved simply by accessing the resource's URI over the Web. Using a Semantic Web browser like Tabulator (slides) or Disco, you can follow links from one resource to the next, surfing the Web of Data. The SPARQL interface enables applications to search and query the database using the SPARQL query language over the SPARQL protocol. A traditional HTML interface offers access to the familiar Web browsers. Requests from the Web are rewritten into SQL queries via the mapping. This on-the-fly translation allows publishing of RDF from large live databases and eliminates the need for replicating the data into a dedicated RDF triple store.

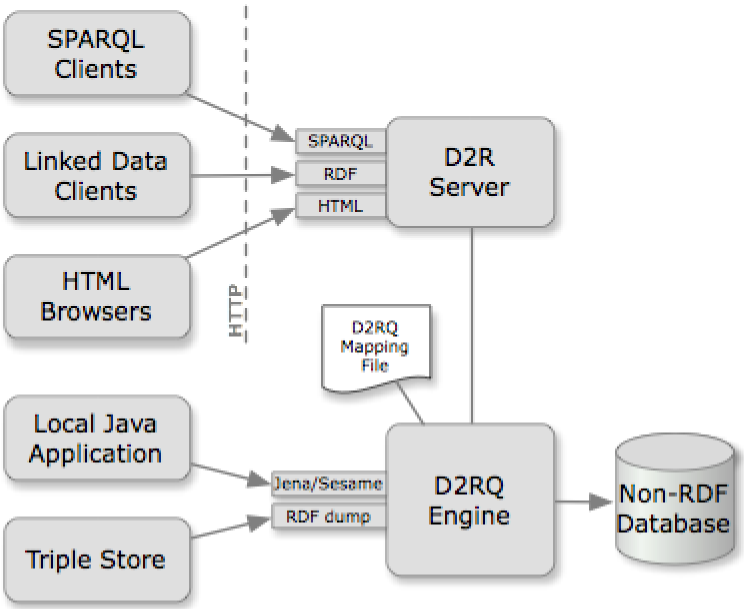


Fig. 8 D2R Server

Note that D2R method does not change the data storage pattern - non-RDF database - means that it does not save data correlations statically in RDF format to construct an RDF database, because N data may has as many as N! correlations, which need much more storage to save than data themselves. On a query, D2R translate it into SQL language according to correlations defined by database structure. On response from database, D2R generates links among returned data according to their correlations. In this way, related data can be extracted and organized to form a virtual data net. Such data net can only be generated dynamically on queries.

Finally, through a data browser that can read the RDF files, and represent the data and their links as objects and arcs, a visualized data net can be shown on user interfaces.

b) Data Lookup

By above method, correlated data can be extracted from database, and visualized as a virtual data net to users. But it needs a query as activation, which is just like picking one node from the data net as a start point. Selection of start point is totally up to users, for examples, from a search result, a leaf node of the data tree, dynamically received intelligence, or else.

On selection of the start point, an RDF-based query is sent to D2R Server, which is then translated into SQL language to interact with database to get result data. From the net view, by this query, all nodes connected with the start node are picked out, with semantics shown on each connection. Under their guidance, user can then pick one connection, and all its connected nodes will be shown. By repeating this, user can finally find the exact data he required, and by the way, some related data that may be useful.

The operation of picking one nodes and the next under guidance from correlation semantics, is quite similar with web page navigation. A correlation is just like a Hypertext Link, and navigation in data nets is as simple as surfing on Internet. User can follow such navigation to the required data – called forward navigation, or follow his footprints back to the start point – called backward navigation. User can be navigated to any direction, but always under his intend, heading his wondered destination.

5 Implementation

To implement and verify the methods proposed in above, a scenario of anti-terrorism operation decision support has been designed. When analyzing a bank robbery event, for example, from the already arrested criminals from one organization, one may think about some other organizations that consume weapons from a same company, and wonder whether they are also suspects. To support this decision, data are required about people, weapons and weapon producers.

Main data table structures, with core data elements and their correlations are shown in Fig. 9. As one can see, each core class – such as People, Identity, Weapon, Type, and Producer – are correlated through bidirectional relations. These elements are extracted and extend from table attribute names, and are core elements that construct the data correlation ontology.

	People	Identity	Weapon
1	John	citizen	-
2	Smith	terrorist	missile
3	Tom	police	gun

Primary Key

	Weapon	Type	Producer
1	missile	heavyWeapon	Iraq
2	gun	lightWeapon	USA

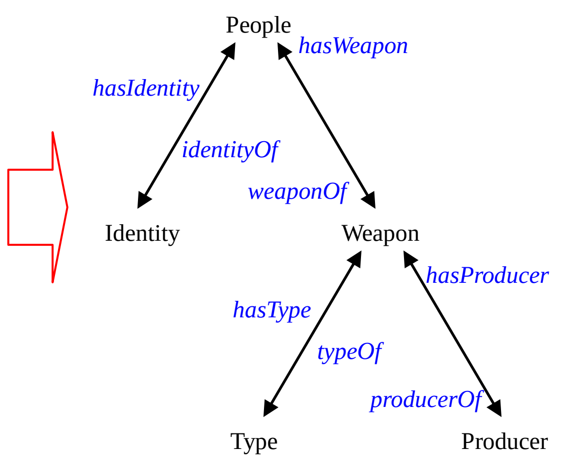


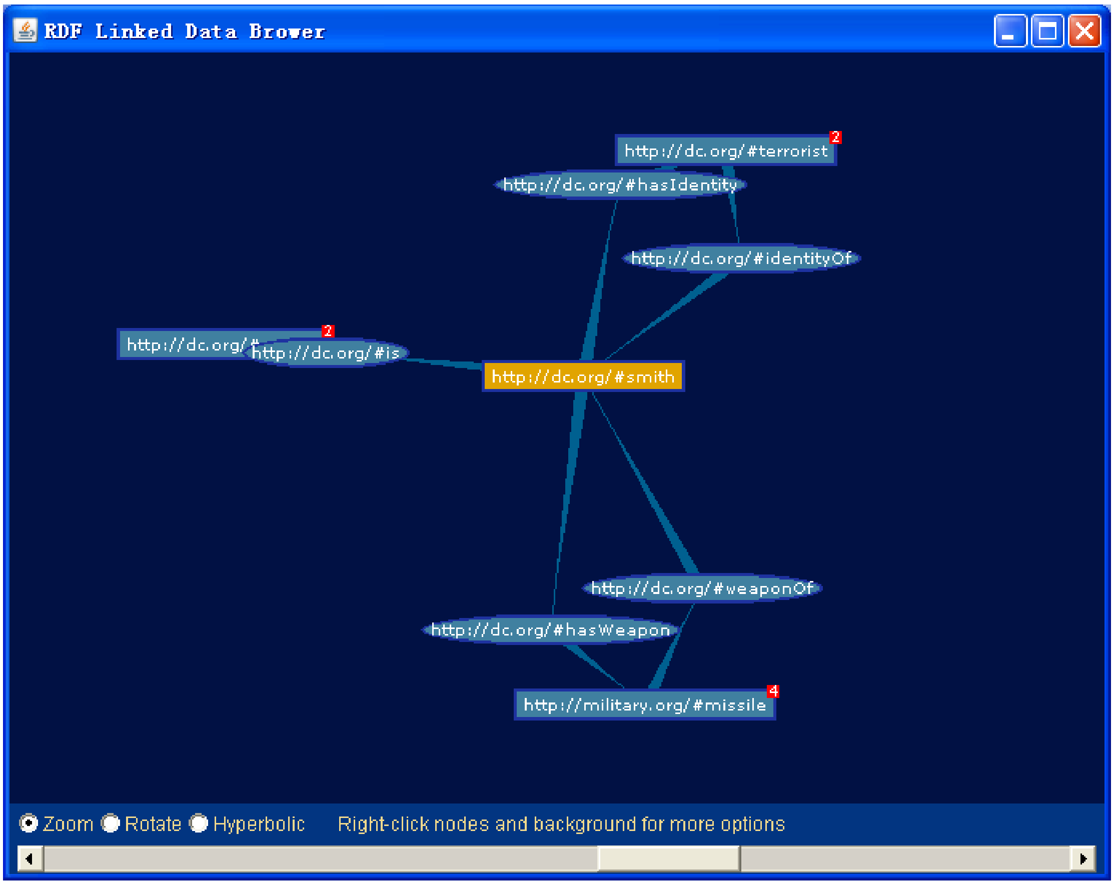
Fig. 9 Scenario data structure and core correlations

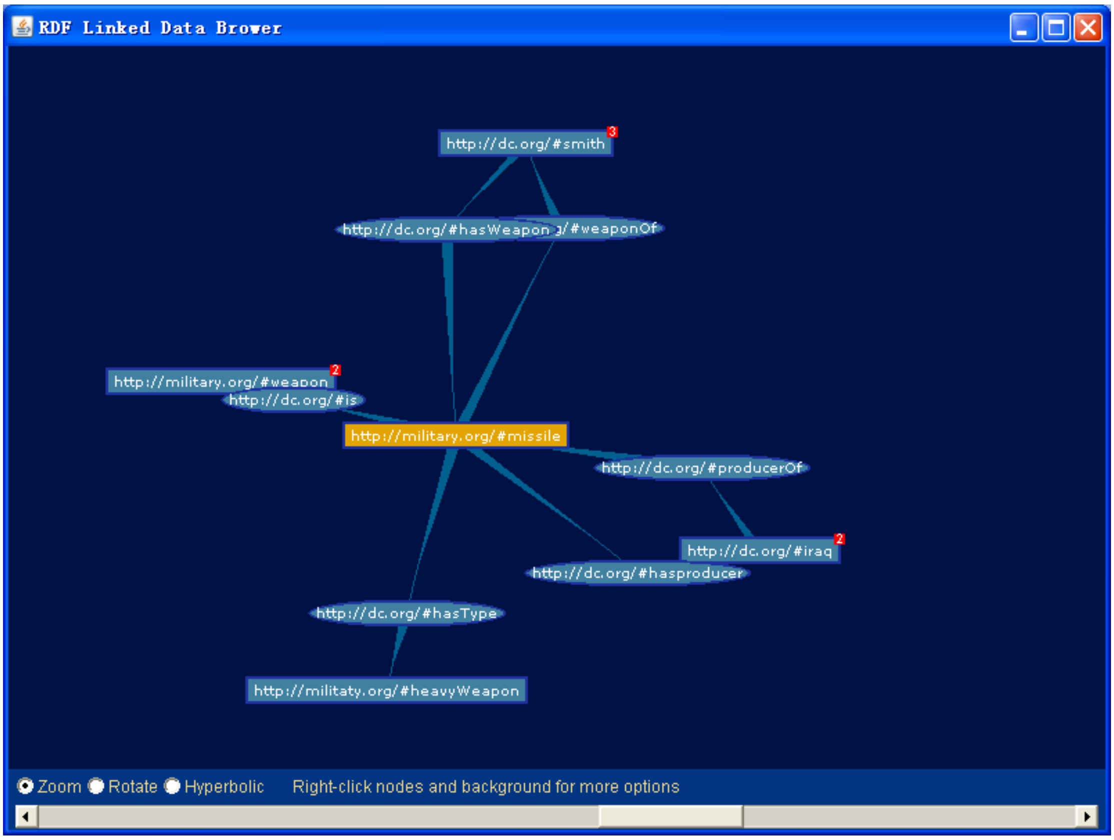
Table values are subclasses of these core elements, and have the same correlation types as defined by the super classes. So the correlations among table values are generated automatically by cloning those of their super classes. For example, core element "People" is related to "Weapon" by "hasWeapon", while table values "Smith" and "missile" are subclasses of "People" and "Weapon" respectively. So "Smith" is related to "missile" through "hasWeapon". These generated correlations can also be saved in coded format – RDF format. A piece of such code is shown in Fig. 10.

```
<rdf:RDF

     xmlns:dc="http://dc.org/#"

     xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"

     xmlns:military="http://military.org/#" >

  <rdf:Description rdf:about="http://dc.org/#tom">

     <dc:hasWeapon rdf:resource="http://military.org/#gun"/>

     <dc:hasIdentity rdf:resource="http://dc.org/#police"/>

     <dc:is rdf:resource="http://dc.org/#people"/>

  </rdf:Description>

  <rdf:Description rdf:about="http://dc.org/#terrorist">

     <dc:identityOf rdf:resource="http://dc.org/#smith"/>

  </rdf:Description>

  <rdf:Description rdf:about="http://dc.org/#USA">

     <dc:producerOf rdf:resource="http://military.org/#gun"/>

  </rdf:Description>

  <rdf:Description rdf:about="http://military.org/#missile">

     <dc:hasproducer rdf:resource="http://dc.org/#iraq"/>

     <dc:weaponOf rdf:resource="http://dc.org/#smith"/>
```

Fig. 10 RDF description of linked data

To visualize data correlations, a user interface has been developed, as shown in Fig. 11. On the user interface, a data is represented in a rectangular, while a correlation is represented in an ellipse. Red number on the right above of data or correlation means how many correlations the data has or how many data objects the correlation is related to.


RDF Linked Data Brower
http://dc.org/#terrorist
http://dc.org/#hasIdentity
http://dc.org/#identityOf
http://dc.org/#is
http://dc.org/#smith
http://dc.org/#weaponOf
http://dc.org/#hasWeapon
http://military.org/#missile
Zoom Rotate Hyperbolic
Right-click nodes and background for more options


RDF Linked Data Brower
http://dc.org/#smith
http://dc.org/#hasWeapon
http://military.org/#weapon
http://dc.org/#is
http://military.org/#missile
http://dc.org/#producerOf
http://dc.org/#iraq
http://dc.org/#hasproducer
http://dc.org/#hasType
http://militaty.org/#heavyWeapon
Zoom Rotate Hyperbolic
Right-click nodes and background for more options

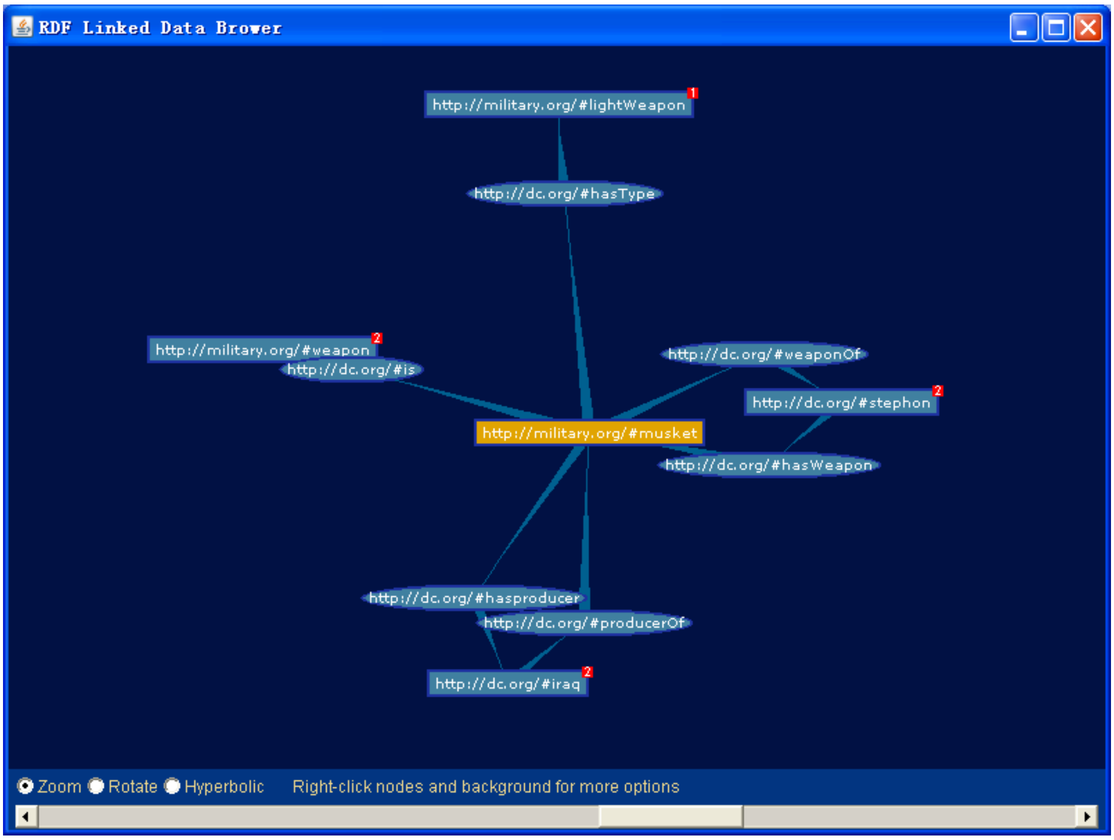


Fig. 11 Visualized data net

In this scenario, for example, the user has selected “smith” as a start point. 3 relations are shown as “hasIdentity” and “hasWeapon” and “is”. On selection of “hasWeapon”, 1 object data is shown as “missile”. On selection of “missile”, 4 relations are shown as “hasType”, “hasProducer”, “weaponOf” and “is”. On selection of “hasProducer”, 1 object data is shown as “Iraq”. On selection of “Iraq”, 1 relation is shown as “producerOf”. On selection of it, 1 object data is shown as “musket”. On selection of “musket”, 4 relations are shown as “hasProducer”, “hasType”, “weaponOf” and “is”. On selection of “weaponOf”, 1 object data is shown as “stephon”.

Thus, under the guidance from data correlation semantics, the user successfully found another suspect “stephon” who consumes weapons from the same company as “smith”.

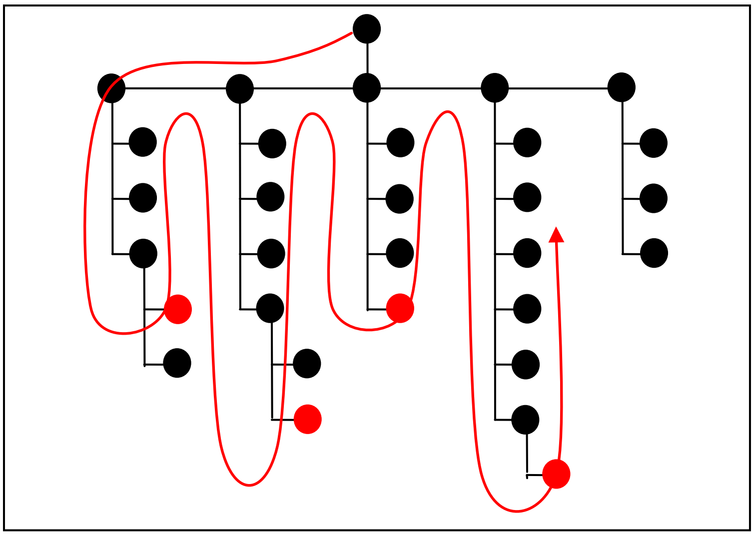
6 Comparison

From the experiment results, data lookup performance of the tree-based and correlation-based methods are analyzed and compared on navigation mode, problem relativity, thinking alignment, operation efficiency, and navigation convergence.

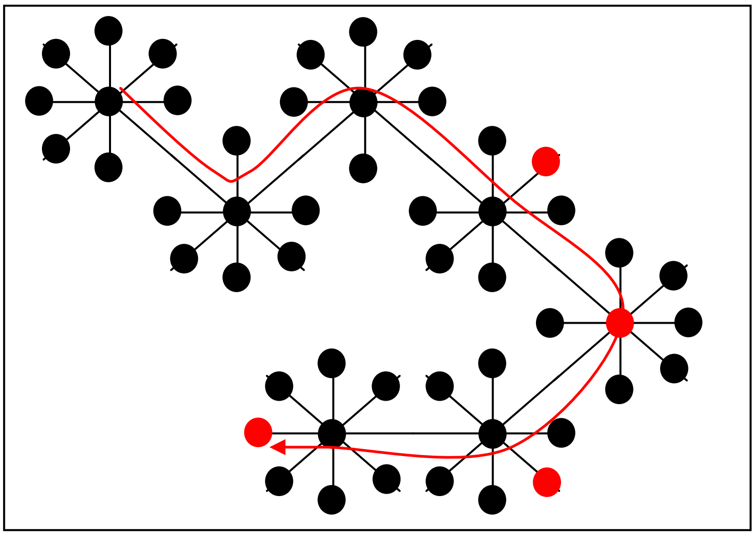
a) Navigation Mode

From user viewpoint, navigation modes in the two methods are quite different. As shown in Fig. 12(a), in tree-based method, user operations are like “unfold -> select -> unfold -> select -> … -> return to root -> unfold -> …”. Decision required data are often in different categories, usually distributed in different branches, and user needs to return to the root node frequently. However, in Fig. 12(b), in correlation-based method, user operations are like “pick -> select -> pick -> select -> …”.

There is no unique root. Because decision required data are always correlated, there are links between each other, and the required data will be distributed not far away from the user tripped path.



(a) Tree-based



(b) Correlation-based

Fig. 12 Navigation mode comparison

b) Problem relativity

As shown in Fig. 12(a), along the trip (in red line), before reaching the required data nodes, there are many branch nodes that has nothing related with the problem. However, in Fig. 12(b), most data nodes picked out are correlated with required data nodes, and thus closely related with the problem.

c) Thinking Alignment

As shown in Fig. 12(a), branch nodes along the trip are not cared by the user, and they are useful only because it can lead the user to the required leaf nodes. However, in Fig. 12(b), most data nodes picked out are correlated with required data, and can guide the user to any direction he is interested in.

d) Operation Efficiency

According to our statistics, in our experiments, in most cases, to find same number of required data, operations needed for tree-based method are relatively more than that for correlation-based one. By analysis, by tree-based method, this depends on the tree's depth and the user's familiarity with the tree's architecture; while by correlation-based method, this depends on richness and redundancy of defined data correlations.

e) Navigation Convergence

In correlation-base method, as user can choose the next data to pick out optionally, he may choose the right direction, or the wrong, as shown in Fig. 13. He may get to the required data quickly following the red line, or take a long trip following the blue line, or even get lost following the green line. This depends on the similarity between correlation semantics. However, in tree-based method, if only all data are categorized correctly, users will finally get to his required data.

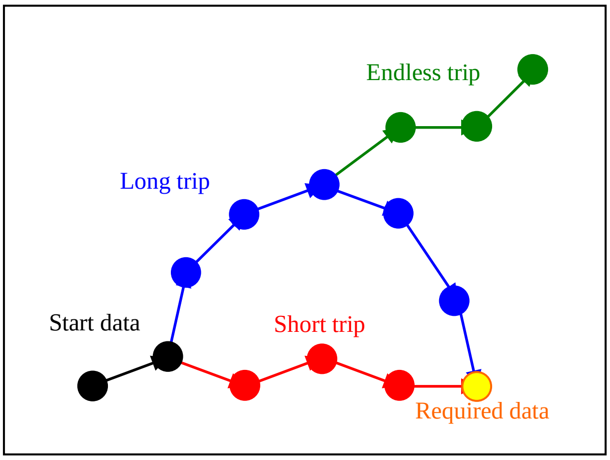


Fig. 13 Navigation convergence problem in correlation-based method

Data lookup performance comparison of the two methods is summarized in Tab. 1. From it, we get a conclusion that correlation-based method is a good complement to tree-based one, but not the substitute. Correlation-based method has higher problem relativity, thinking alignment ability, and operation efficiency, while tree-based method has higher navigation convergence. If the two methods can be combined together efficiently, the data support quality will be improved a lot than using either of them.

Tab. 1 Performance comparison of tree-based and correlation-based methods

	Problem Relativity	*Thinking Alignment*	*Operation Efficiency*	*Navigation Convergence*
Tree-based	Low	Low	Low (Dependent)	High
Correlation-based	High	High	High (Dependent)	Low

7 Conclusion

In this paper, correlation-based decision data support method has been proposed, with methods proposed on data link construction and data lookup, which is proved to be a good complement to traditional tree-based one, which brings decision data support with higher problem relativity, user thinking aligning ability, and operation efficiency. Future researches will be focused on development of automatic link construction mechanism on more kinds of correlations.

References

[1] B. L. Tim, “The next Web of open, linked data,” 2007. http://www.ted.com/talks/tim_berners_lee_on_the_next_web.html.

[2] Linking Open Data Community Project. 2011. http://esw.w3.org/SweoIG/TaskForces/CommunityProjects/LinkingOpenData.

[3] W3C Semantic Web Education and Outreach Group. 2010. http://www.w3.org/2001/sw/sweo/.

[4] Linked Open Data Statistics. 2010, http://esw.w3.org/topic/TaskForces/CommunityProjects/LinkingOpenData/DataSets/LinkStatistics

[5] B. L. Tim, H. James, and L. Ora, “The Semantic Web,” Scientific American, vol. 284, May 2001, pp. 34-43.

[6] C. Turnitsa, and T. Andreas, “Battle Management Language: A Triangle with Five Sides,” Spring Simulation Interoperability Workshop, Simulation Interoperability Standards Organization, IEEE CS Press, April 2006.

[7] D. Saikou, T. Andreas, and T. Chuck, “Merging Protocols, Grammar, Representation, and Ontological Approaches in Support of C-Bml,” Fall Simulation Interoperation Workshop, Orlando, FL, USA, 2006.

[8] P. R. Smart, A. Russell, and N. R. Shadbolt, “Integration of Knowledge Intensive Multi-Agent Systems,” KIMAS 2007, Int. Conf. Digital Object Identifier, 2007, pp. 434 – 439.

[9] C. Matheus, M. Kokar, K. Baclawski, J. Letkowski, C. Call, M. Hinman, J. Salerno and D. Boulware. SAWA: An Assistant for Higher-Level Fusion and Situation Awareness. In Proc. SPIE Conference on Multisensor, Multisource Information Fusion, Orlando, Florida, March 2005.

[10] P. Cohen, R. Schrag, E. Jones, A. Pease, A. Lin, B. Starr, et al, “The DARPA high-performance knowledge bases project,” AI Magazine, vol. 19, 1998, pp. 25-49.

[11] P. Bruce, and T. Dan, “Rapid Knowledge Formation,” http://userweb. cs.utexas.edu/users/mfkb/RKF/projects/rkf.html, April 19, 2006.

[12] P. R. Smart, A. Russell, N. Shadbolt, m. c. schraefel, and L. A. Carr, "AKTiveSA: A Technical Demonstrator System for Enhanced Situation Awareness in Military Operations Other Than War," The Computer Journal, vol. 50, pp. 703-716, 2007.

[13] B. L. Tim, “Design Issues: Linked Data,” 2006. http://www.w3.org/DesignIssues/LinkedData.

[14] C. Bizer, and R. Cyganiak, “D2R Server - Publishing Relational Databases on the Semantic Web,” 2009. http://www4.wiwiss.fu-berlin.de/bizer/pub/LinkedDataTutorial/#intro.