

---

## Building Semantic Webs for e-government with Wiki technology

---

Christian Wagner,\* Karen S.K. Cheung and Rachael K.F. Ip

Department of Information Systems,  
City University of Hong Kong,  
83 Tat Chee Avenue, Kowloon, Hong Kong  
E-mail: iscw@cityu.edu.hk    E-mail: iskaren@cityu.edu.hk  
E-mail: isip@cityu.edu.hk  
\*Corresponding author

Stefan Böttcher

Department of Computer Science,  
University of Paderborn,  
Fürstenallee 11, D-33102 Paderborn, Germany  
E-mail: stb@upb.de

**Abstract:** E-government webs are among the largest webs in existence, based on the size, number of users and number of information providers. Thus, creating a Semantic Web infrastructure to meaningfully organise e-government webs is highly desirable. At the same time, the complexity of the existing e-government implementations also challenges the feasibility of Semantic Web creation. We therefore propose the design of a two-layer semantic Wiki web, which consists of a content Wiki, largely identical to the traditional web and a semantic layer, also maintained within the Wiki, that describes semantic relationships. This architectural design promises several advantages that enable incremental growth, collaborative development by a large community of non-technical users and the ability to continually grow the content layer without the immediate overhead of parallel maintenance of the semantic layer. This paper explains current challenges to the development of a Semantic Web, identifies Wiki advantages, illustrates a potential solution and summarises major directions for further research.

**Keywords:** e-government; Semantic Web; Wiki technology; Knowledge Management (KM).

**Reference** to this paper should be made as follows: Wagner, C., Cheung, K.S.K., Ip, R.K.F. and Böttcher, S. (2006) 'Building Semantic Webs for e-government with Wiki technology', *Electronic Government*, Vol. 3, No. 1, pp.36–55.

**Biographical notes:** Christian Wagner is a Professor in Information Systems at the City University of Hong Kong. He has authored and coauthored more than 60 papers, on issues such as Knowledge Management (KM), e-government, decision support and database design. He presently leads a project on KM with Wikis for the SIM Advanced Practices Council. Previously, he advised Hong Kong civil servants on KM and the use of

information technology to support KM efforts. He serves on the Advisory Board of the Teradata University Network, and on the Editorial Boards of several journals. He obtained his PhD in Commerce and Business Administration from the University of British Columbia (Vancouver) and Graduate and Undergraduate degrees in Industrial Engineering from the Technical University Berlin.

Karen S.K. Cheung is an Information Systems Instructor at the City University of Hong Kong. She received her Bachelor, Master and PhD degrees in Information Systems from the City University of Hong Kong. Her research interests include information retrieval, virtual communities and e-government.

Rachel K.F. Ip is an Information Systems Instructor at the City University of Hong Kong. She received a Master's degree in Information Systems from the City University of Hong Kong and a Master's degree in Education from the University of Hong Kong. Her research interests include social computing, virtual communities and e-government.

Stefan Böttcher is a Professor of Computer Science at the University of Paderborn. He has authored or coauthored more than 50 papers, on issues such as *Transaction Management in Databases and Mobile Ad hoc Networks*, query optimisation in XML documents and knowledge bases, data integrity, security and privacy protection in information systems and middleware technologies for e-commerce. After receiving his PhD from the Johann Wolfgang Goethe-University in Frankfurt, he held positions as a Research Fellow at IBM Scientific Center in Stuttgart and as a Senior Researcher at the Daimler Benz Research Center Ulm. He became a Professor at the University of Applied Science in Ulm, before he joined the University of Paderborn.

---

## 1 Introduction

The effectiveness of any governments is reflected by its efficiency, better services to citizens and improved governmental processes (Grönlund, 2004). Many governments therefore have embarked on aggressive campaigns to vastly increase the number of online interactions with citizens, and to provide large amount of online information and knowledge to citizens (as well as to government employees) (Amaravadi, 2005; Liebowitz, 2004; Metaxiotis and Psarras, 2005). A review of government pages available through the Google search engine demonstrates this reality, as shown in Table 1. The largest contingent of government pages, from the USA and represented by the .gov top-level domain, accounts for 368 million pages. No other government comes close. But even smaller e-government sites, such as .gov.uk (9.28 million pages) or .gov.au (7.2 million pages) exceed the size of major company sites such as IBM (3.93 million for ibm.com), eBay (3.14 million for ebay.com) and dwarf sites of companies such as Ford (55,700 for ford.com) or Barclays Bank (24,200 for barclays.co.uk). Even Slovenia, a country with only 2 million citizens, maintains a vastly larger e-government website of over 380,000 pages.

To encourage usage of all this information and knowledge, an e-government must be able to provide a convenient way for its citizens to access and obtain the information and knowledge they desire, without having to manually filter out too much content that is not

needed. Some governments try to do this by building aggregator sites or portals to capture the content created by multiple government sources. For example, Hong Kong's government has individual portal solutions (ESDLife) for almost every government department, as well as a portal for electronic content and service delivery to citizens (<http://ESDlife.com.hk> and <http://esd.gov.hk>).

**Table 1** Page count of selected e-government sites available through Google (June 2005)

<i>Country</i>	<i>Government domain</i>	<i>Number of web pages</i>
USA	.gov	368,000,000
Canada	.gc.ca	12,100,000
UK	.gov.uk	9,280,000
Australia	.gov.au	7,200,000
China	.gov.cn	2,630,000
New Zealand	.gov.nz	1,290,000
South Africa	.gov.za	816,000
Hong Kong	.gov.hk	887,000
Thailand	.gov.th	728,000
Slovenia	.gov.si	388,000

Other than portals, as in the case of ESDLife, there are several technologies governments can use to provide useful information to its citizens through the internet, such as special KM portals, CRM software, content management systems, e-mail broadcasts, listservs or discussion forums. All these solutions can help to disseminate and exchange information, each of which with its own strengths and weaknesses. Overall however, they face the same difficulties, namely the management of too much information, created by too many heterogeneous, distributed sources. Resulting issues such as inconsistent terminologies, information overload and too little maintenance of outdated knowledge are only too frequent.

Hence, to handle knowledge organisation and access in a more feasible way, an e-government needs to add some 'logic', a semantic structure, to organise its knowledge offerings. Hyperlinks can be used to provide information to citizens by allowing them to access information with a relatively little effort. However, to organise the links in a manageable way is a great challenge to the websites developers, and hyperlinks in conventional HTML do not have a definable meaning.

One approach to solve this problem is the development of a Semantic Web for e-government. Such a web can be used to develop a more effective and transparent e-government (Klischewski, 2003). As with any Semantic Web, site developers would annotate the web pages with semantic markup, semantic links and metadata so as to enable machines to follow the links and ideally to facilitate the integration of knowledge and information from many different sources (Berners-Lee and Miller, 2002). Semantic markup refers to a markup language whose name spaces, vocabulary and relationships are meaningfully definable. Semantic links are pointers between web objects, which

can be meaningfully interpreted because of their labelling elsewhere in the system, and because they may have properties or methods associated with them. The metadata adds further well-defined, meaningful information so as to facilitate machine readability even more and to enable better web analysis to furnish best results to citizen users.

Semantic Web technology has been used for Knowledge Management (KM) in e-government, with ontologies to provide a well-defined means for users to structure and access the knowledge effectively. For instance, Klischewski (2003) launched a project at Hamburg University to apply Semantic Web technology to enable the contextualisation of DiBIS ([www.dibis.de](http://www.dibis.de)), a web information service for Hamburg's citizens. Another project, conducted by Fraser et al. (2003), developed the SmartGov e-government ontology to provide the public authority with a knowledge-based core repository for government transaction services. Daddieco (2004) developed an ontology for the subject domain of export controls in the US government for effective knowledge retrieval and sharing.

An obvious starting point for these projects has been the development of *ex ante* ontologies, with the objective to provide an authoritative framework, which could then structure all contents subsequently added to the web. Unfortunately, the practical application is not as easy as the theoretical concept suggests. As has been recognised in the research on schema integration in database design (e.g. Storey et al., 1997), or knowledge integration among just a few experts (Gaines and Shaw, 1993), creating a common vocabulary or a common terminology is a difficult task even at a relatively small scale. In a large environment with multiple, heterogeneous knowledge sources, and fast changing content, maintaining such a 'clean' ontology appears highly challenging (Klischewski, 2003), if not impossible. Furthermore, since the knowledge resources on e-government websites consist of different electronic objects such as files for downloads (e.g. forms and customer applications), transactions processed by back end applications, links and services that require further interaction (e.g. authentication), it is difficult to identify which objects are the candidate objects that need semantic markup. The more markup, the more effort required for creation and maintenance of the semantic links. An analysis of a large online encyclopaedia, the Wikipedia ([wikipedia.org](http://wikipedia.org)), for instance shows that inter-web hyperlinks grow linear with the number of web pages ( $R^2 = 0.997$ , based on 38 observations). In the Wikipedia's case, each page contained on average more than 22 links, resulting in over 10 million links for over 500,000 web pages. By comparison, an e-government web with millions of web pages and possibly tens of millions of links between them would require formidable maintenance capability.

Clearly then, both the challenge of creating and maintaining a semantic e-government web with different name spaces, formats and structures for the same kind of knowledge, and the challenge of integrating knowledge represented at different levels of granularity, refinement, or precision by multiple heterogeneous sources demand novel solutions. We will discuss the main challenges and their treatment through existing approaches more systematically in Section 2. In Section 3, we explain our proposed solution, the creation of a two-layer semantic Wiki web. Section 4 illustrates our solution with an example, while Section 5 reviews strengths and weaknesses, and identifies possible extensions to this research. Section 6 draws conclusions.

## **2 Challenges of existing technology solutions**

### *2.1 Overview*

The application of Semantic Web technology has promising results in e-government and other applications, such as e-commerce and search engines (Klischewski, 2003). However, designing and developing a Semantic Web for an application such as an e-government web is also a daunting task. The following points highlight major challenges.

#### *2.1.1 Difficulty in extracting knowledge*

It is difficult to extract the knowledge and information from documents and people. Extracting knowledge from documents requires some form of data or text mining, which is difficult when documents are unstructured. Knowledge extraction from people is often hindered by the fact that people tell less than they know (Polanyi, 1966), a phenomenon often explained by the term “tacit knowledge” (see also Nonaka and Takeuchi, 1995).

Yet, it is even more difficult to identify the semantic relationships between these knowledge objects. Semantic relationships are evolving, and thus it is impossible to know all of them upfront. This creates a problem in updating, interpreting and navigating the information.

#### *2.1.2 Heterogeneous knowledge sources*

Knowledge Management in e-government services has to address at least three areas: in relationship to the public, within government departments and in the coordination between government departments (Wagner et al., 2003). A large amount of diverse knowledge is required, with knowledge sources being located in different departments, often spread out over the entire country or state. Yet we need to build linkages across, to meaningfully connect the separate knowledge offerings from various sources. For example, a citizen who wishes to run a liquor store may have to deal with one government department that issues business licenses, one that issues liquor licenses, with the tax department, labour department and so on. All will be involved, but neither one will provide the ‘big picture’, unless their knowledge is semantically linked. Furthermore, different sources will represent knowledge at different levels of granularity, with different points of view, and thus possibly inconsistencies or contradictions.

#### *2.1.3 Limit to expertise*

With large size and heterogeneity of knowledge sources comes the difficulty in maintaining a common, consistent truth. Yet, with even a smaller government having hundreds of thousands of web pages, no single individual or group may have the necessary knowledge to verify the entire Semantic Web, both content and the semantic links. Hence, having an e-government Chief Knowledge Officer (CKO) dictate what is correct or incorrect knowledge may not at all work, due to the limits of the CKO’s own knowledge.

#### *2.1.4 Non-technical end user contributions*

Given the size of the e-government Semantic Web, it cannot be developed by a group of programmers. Non-technical 'end users' will be the ones who need to specify the semantic knowledge. Yet they commonly lack the technical knowledge to understand the implicit semantic links between web documents in the format and notation that is typically part of Semantic Web development. In fact many of these non-technical users may not even be able to write web documents in HTML.

#### *2.1.5 Mistakes made in Semantic Web development*

Given the size of e-government Semantic Webs, and the involvement of non-technical users, mistakes will be made. Semantic Web developers require training in the Semantic Web language. End users will not have such training, and thus, even though they possess the domain knowledge, will make mistakes in representing it, usually by choosing inefficient or wrong representations to describe a concept in the ontology (compare Wagner, 2000).

#### *2.1.6 Difficulty in maintaining knowledge*

The government is one of the largest online knowledge providers for citizens. In fact, as Table 1 demonstrates, many government websites are an order of magnitude larger than those of big corporations. As the knowledge volume grows, KM, including links management and maintenance becomes increasingly difficult, error prone and costly.

#### *2.1.7 Ambiguous web*

As web documents change rapidly, so does their semantic relationship. If one page, say exp.htm, previously served as an explanation to another page, qp.htm, but now contradicts it, the semantic relationship is no longer one of explanation. Moreover, web documents will invariably combine or 'couple' multiple knowledge components, which makes a unique classification in a Semantic Web ambiguous. For example, a single page may raise a question, answer that question, describe conditions under which the answer is true and give counter examples. Hence, such a page cannot be classified as a question, or an answer, or by any other single classifier. This requires extra attention and cost to update and maintain the knowledge base.

### *2.2 Research question*

Can we build a semantic knowledge repository that deals with all these challenges? Attempting to find a solution alternative to overcome the challenges is the purpose of this paper. In the previous section, we have discussed the significant challenges faced by a semantic e-government web. These challenges are present in the entire KM cycle, from knowledge elicitation to formalisation, to Semantic Web design, implementation and maintenance of the knowledge (compare Holsapple and Joshi, 1999). A new paradigm is needed which recognises these inherent limitations, and defines a sufficiently robust development approach to address the challenges. Table 2 summarises the challenges and their corresponding possible remedies.

**Table 2** Challenges and potential remedies

<i>Challenge</i>	<i>Suggested remedies</i>
Difficulty in extracting knowledge	Develop a mechanism to capture and maintain evolving knowledge using Wiki technology
Heterogeneous knowledge sources	Involve multiple users across departments and outside government
Limit to expertise	Collaboratively aggregate the knowledge of multiple experts and non-experts
Non-technical end user contributions	Make the semantic relationships accessible to the end users, by making them independent of the content web in a separate layer
Mistakes made in Semantic Web development	Create a semantic knowledge repository that represents Semantic Web definitions explicitly, scrutinisable by many ‘eyeballs’ (users)
Difficulty in maintaining knowledge	Use Wiki technology to enable a community to openly create and maintain the knowledge in the Semantic Web. Allow changes in content web and semantic relationships to be made separately from each other
Ambiguous web	Remove ambiguities through iterative refinements and distributed knowledge refactoring

Given all the challenges faced by a semantic e-government web, we propose to develop a semantic Wiki web, details of which are discussed in Section 3.

### 3 Solution proposal: semantic Wiki web

#### 3.1 Proposed solution overview

The conceptualisation of a knowledge resource, which can address the challenges of e-government semantics webs, will be the purpose of this section. Fundamentally, the proposed solution relies on the following key elements:

- use a many-to-many technology for free and open knowledge sharing, with a few technical demands on the user; namely Wiki technology and the Wiki way
- create a two-layer design, one layer being the traditional web, the second layer being semantic relationships, also *explicitly written* and maintained as Wiki web pages using simplified syntax; namely Notation 3
- rely on a community of users to maintain the semantic relationships in the form of a Wiki web.

Each of these elements will be discussed in the following subsections. Taken together, we surmise, they will enable the design of a Semantic Web architecture for e-government that is not only technologically possible, but also socially feasible.

### 3.2 Wiki technology and the Wiki way

#### 3.2.1 Wiki definition

A Wiki (from *WikiWiki*, meaning ‘fast’ in Hawaiian) is a set of linked web pages, created through the incremental development by a group of collaborating users (Leuf and Cunningham, 2001), as well as the software used to manage the set of web pages. Ward Cunningham developed the first Wiki in 1995, as the *PortlandPatternRepository*, to communicate specifications for software design. The term Wiki (from the Hawaiian *WikiWiki* meaning ‘fast’) gives reference to the speed with which content can be created with a Wiki. According to the Wikipedia ([www.wikipedia.org](http://www.wikipedia.org)), an online encyclopaedia written as a Wiki, its key characteristics are as follows:

- it enables web documents to be authored collectively
- it uses a simple markup scheme (usually a simplified version of HTML)
- Wiki content is not published instantly, once the author submits the Wiki page to the Wiki engine
- new web pages are created when users create a hyperlink that points nowhere (usually simply by writing a term in ‘CamelCase’, concatenating two or more words and capitalising them).

#### 3.2.2 Wiki design principles

Wiki design is based on 11 principles originally formulated by Cunningham (e.g. <http://c2.com/cgi/wiki?WikiDesignPrinciples>), shown in Table 3. These principles describe, in part, what has been termed the ‘Wiki way’, namely a form of collaborative web writing. As such, the principles describe how web creation, maintenance and access should take place, as well as the features the technology has to provide, so as to enable this form of collaboration.

**Table 3** Wiki design principles

<i>Principle</i>	<i>Explanation</i>
Open	If a page is found to be incomplete or poorly organised, any reader can edit it as he/she sees fit
Incremental	Pages can cite other pages, including pages that have not been written yet
Organic	The structure and text content of the site is open to editing and evolution
Mundane	A small number of (irregular) text conventions will provide access to the most useful (but limited) page markup
Universal	The mechanisms of editing and organising are the same as those of writing so that any writer is automatically an editor and organiser
Overt	The formatted (and printed) output will suggest the input required to reproduce it (e.g. location of the page)

**Table 3** Wiki design principles (continued)

<i>Principle</i>	<i>Explanation</i>
Unified	Page names will be drawn from a flat space so that no additional context is required to interpret them
Precise	Pages will be titled with sufficient precision to avoid most name clashes, typically by forming noun phrases
Tolerant	Interpretable (even if undesirable) behaviour is preferred to error messages
Observable	Activity within the site can be watched and reviewed by any other visitor to the site
Convergent	Duplication can be discouraged or removed by finding and citing similar or related content

Why are these principles important? Previously we described that the design and maintenance of a knowledge-based system is a difficult task for those who have domain knowledge expertise, but who are usually not technical experts. This calls for a web technology whose effort and cost of web page creation is very low, whose technology is forgiving towards mistakes, whose errors are easily detected and where detected errors can be instantly repaired.

Table 4 overviews the desired web benefits, and the Wiki principles that support them. For instance, low effort in creation and maintenance is realised through an interface that requires very little technical knowledge to write and publish content. Using plain text, or a simplified ('mundane') markup language, even non-technical users can create text (albeit not necessarily RDF syntax). The principle of incrementalism allows contributors to add a little or a lot, even links to non-existing pages, which the Wiki identifies as such. Web pages typically come with an edit button as part of Wikis' organic nature, allowing anyone to edit.

**Table 4** Wiki principles and benefits

<i>Desired web benefit</i>	<i>Supporting Wiki principles</i>
Low effort of development and maintenance	Open, mundane, incremental, organic
Forgiving towards mistakes	Tolerant, open
Ease of error detection and repair	Open, observable, universal

Inevitable mistakes can be easily corrected, as Wikis are open for edit by anyone, and even incorrectly written pages will be rendered. This process is further simplified by the fact that results of any page creation or maintenance are instantly visible ('observable'), and accessible by a standard user interface for content creation and maintenance throughout ('universal').

Other principles, such as precision or unified naming describe less the technology and more the way in which Wikis should be used. In addition to these, there are further conventions for content creation and maintenance, as can be found for instance in the

Wikipedia (<http://wikipedia.org>). One such convention describes ‘refactoring’ of content, that is, the decomposition and recomposition of web pages to create a single semantic unit as a single page, with pointers (links) being used rather than content being duplicated (‘convergence’).

As previously identified, in a Semantic Web environment, where both the content and the semantic knowledge will be evolving, being able to easily refactor the content appears to be a high priority, one that can be realised by Wiki technology.

Given the benefits of Wiki technology in the representation of Semantic Web content, one would expect that there are already Wiki implementations. In fact, the Platypus Wiki (Tazzoli et al., 2004) is one such solution. Its main difference to the solution proposed here, however, is the representation of semantic content ‘behind’ the rendered web pages, as will be explained in the next subsection.

### *3.3 Two-layer design*

One of the proposed design elements is an explicit representation of Semantic Web links, as a separate layer of web pages themselves. Instead of hiding the semantic information ‘behind’ the rendered pages, one layer of the Semantic Web would consist of explicitly visible and maintainable pages, expressing the semantic relationships.

Using a multi-layer design is a concept borrowed from database systems (cf. Garcia-Molina et al., 2002). Within database systems, the bottom layer involves the raw data containing the data or information itself, whereas on top of the raw data layer additional indexes are used for fast access to specific data according to an index-specific search criterion. Whenever multiple different search criteria shall be supported, multiple indexes must be defined. A database index, however, is always stable in the sense that the criterion used to search data with a given index cannot change. This non-flexible behaviour is an advantage in database systems. In knowledge bases for e-government it is not, because the criteria used to search for knowledge or to follow a link are likely to vary over time. Thus, an adaptive indexing approach is needed as part of the solution. It is incorporated in our two-layer design.

Using a two-layer approach in the web means that the traditional web pages that contain the content can be developed independently from a semantic structure overlaid on it, thus enabling the incremental growth of either, if necessary, independently. In other words, a group of content creators would be able to develop new web pages, without having to resolve all the semantic relationships between the new web pages and existing ones. Hence, content creation would not be stifled. In addition, semantic relationships could be built incrementally, growing the reasoning capability of the Semantic Web incrementally.

Making semantic links explicit is another approach borrowed from relational database design. The relational data model models relationships by so-called foreign-key constraints that require the use of identical values in related attributes of the key and the foreign key. In comparison, other data models (the hierarchical, network, object-oriented or XML data model) hide meaning within their implicit structures such as parent–child hierarchies, pointer sets or network sets. Within these data models, database users and maintainers use the implicit (semantic) structures in queries. We consider explicit semantic relationships a significant advantage whenever flexible semantic-based search has to be supported, as for instance in KM for e-government. A further advantage is that once relationships are made explicit, they can be more easily managed (e.g. updated),

interpreted and navigated. For the Semantic Web, an explicit representation of the semantic structure enables the separation of the document web itself from the semantic relationships and a separate maintenance of each.

Note however that our concept of making links explicit is less restrictive than the foreign-key concept of relational databases in the following sense. In the relational data model, a reference to a data record can only be defined when the referred-to data record already exists (otherwise a foreign key constraint violation error occurs). In comparison, our concept of modelling semantic relationships by Wikis has the same advantage of making the relationship explicit, but, additionally allows a content provider to define a link to a piece of knowledge that has not yet been created. We consider this way of representing semantic relationships advantageous compared to foreign key constraints, because it allows Wiki web knowledge bases to grow dynamically, rapidly and without global coordination.

Our semantic overlay also draws on lessons from P2P networks (Bergamaschi and Guerra, 2002), where overlay network structures are defined on top of a given network structure. As in P2P systems, our two-layer architecture provides a semantic view of a network of nodes (or pieces of information in our case), which is independent of the nodes' physical location. Another property that our two-layer architecture has in common with P2P overlay networks is that the links followed in the network depend on the task and may change over time. However, in P2P systems the optimisation criterion that selects the appropriate links is fixed and implemented as part of an algorithm. In comparison, in our approach the criteria to follow a link depend on the Wiki specifications in the overlay layer, which are modifiable on demand by the users that manage the overlay layer. This enables our overlay structure to adapt its search structure with the help of the community users and according to the requirements of the community users, which we consider to be a significant contribution to community-supported KM for e-government.

### 3.4 *Multi-user network*

Maintaining the Semantic Web for an e-government site invariably becomes a large-scale project, with many heterogeneous data sources, separated both physically and semantically from each other. No individual or small group of people will be able to maintain such a web. It has to rely on the resources of a large, possibly volunteer, network to create and maintain meaning relationships.

As the case of the Wikipedia demonstrates, a volunteer network was able to create the world's largest encyclopaedia (now with over 500,000 concepts in the English language), within only a few years. This result has been achieved by a volunteer network, that as of May 2005 counts more than 25,000 members, with more than 1000 of them making more than 100 edits per month each, equalling the full-time efforts of a staff of over 200 researchers (estimate based on Wikipedia Statistics). See <http://en.wikipedia.org/wikistats/EN/TablesWikipediaEN.htm> for details.

Not only can such a network provide the needed manpower to maintain a large Semantic Web, it can also provide the multiple points of view needed to create a multitude of semantic relationships, and can provide, following Linus' Law, the needed 'many eyeballs' that would identify mistakes in the existing Semantic Web structure and enable their correction.

## 4 Implementation and illustration

In this section, we will explore how a semantic Wiki web might work in an actual problem scenario. The example will illustrate a ‘help facility’, a simplified expert system (Hayes-Roth et al., 1983) implemented as a semantic Wiki web. The illustration makes use of the MediaWiki software, which has no embedded RDF or Notation 3 interpreter. Hence we will not demonstrate the actual knowledge processing, but the conceptual structure, web content and navigation structure.

### 4.1 ‘Help Facility’ Wiki

One important class of software applications today is KM systems that provide interactive help. These include, for instance, embedded systems, such as the help function of applications software, as well as helpdesk applications that exist in many companies to facilitate customer support. Frequently, these systems analyse a problem by traversing through a hierarchical or network search tree, ruling out irrelevant nodes through question and answer dialogue and then relay the most appropriate response for the given condition. In the past, such systems were implemented through a range of technologies. During 1980s, expert systems emerged as a technology particularly suited for this task type. Expert systems were successful in a number of well-publicised applications, but also suffered from several weaknesses, such as their brittleness at the limits of expert system knowledge, narrow domain focus, maintenance difficulty and the role conflict between domain expert and knowledge engineer (e.g. Hayes-Roth et al., 1983; Waterman, 1986).

Semantic Wiki webs promise an opportunity to acquire the expertise needed for help system development in a less rigid, incremental manner. The resulting system might be able to answer questions based on expert knowledge, but with a less formal knowledge base than traditional expert systems. In fact, the knowledge base could emerge from previously disconnected or only loosely connected web pages. For example, a forward-looking government requires its citizens to save part of their income in a self-managed ‘mandatory provident fund’, maintained by a non-government financial institution. Citizens who are not skilled in making investment decisions may then look towards the government to provide guidance on how best to invest. Similarly, government and civil service will have an interest in citizens making good investment decisions, so as to avoid large social welfare payments or social unrest in case of delinquencies.

Hence, a citizen might approach a government information site with the following question: “I have money in my mandatory provident fund, presently in a savings account, earning almost no interest. I want to invest this money. What is my best choice?” Traditionally, a response would have been furnished in form of stand-alone hard copy documents, or in an e-government site as flat web pages, pages such as ‘Investing in Real Estate’ or ‘How to Invest in Stocks?’, logically not connected and with overlapping content. Figure 1 shows one such example.

Working towards the development of a semantic Wiki web, a first step would be the refactoring of individual web pages into a more loosely coupled, internally a more cohesive set of information components, which together would provide the needed information. The conceptual structure of such a system is shown in Figure 2. The system depicts a content web, with pages shown as rectangular boxes, and arrows depicting the (semantic) relationships between the pages.

**Figure 1** ‘Flat’ page ‘how to invest in stocks’

## HowToInvestInStocks

A **stock**, also referred to as a **share**, is commonly a share of ownership in a [corporation](#).

**Table of contents** [\[hide\]](#)

- [1 Investment Choices](#)
- [2 Why Invest in Stocks?](#)
- [3 About Stocks](#)
- [4 Investment](#)

### Investment Choices

Investors today have many choices. One popular one, especially for long-term investors, is stock investment.

### Why Invest in Stocks?

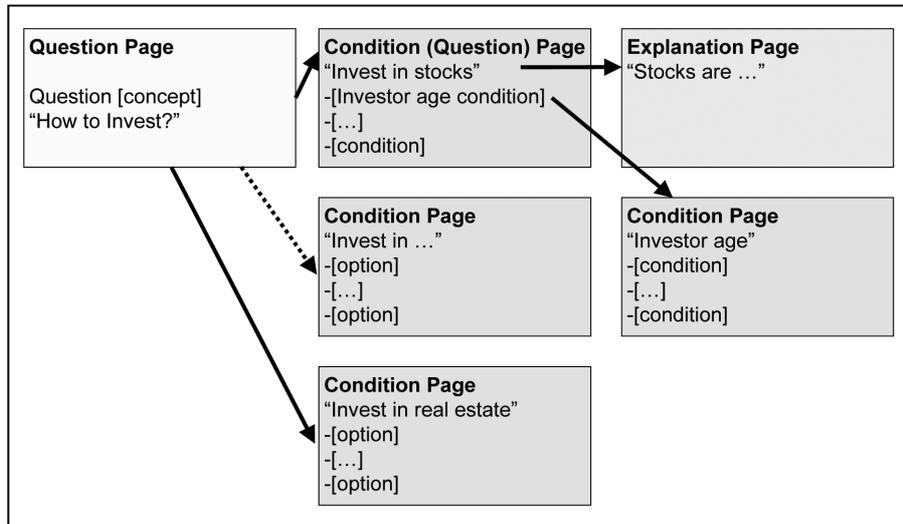
As a stock investor, you become part owner of a corporation and can take part in that company’s success (or failure). Stocks frequently offer above average returns, thus fighting off the depreciation of your assets brought about by inflation.

### About Stocks

The owners and [financial](#) backers of a company may want additional capital to invest in new projects within the company. If they were to sell the company it would represent a loss of control over the company.

Alternatively, by selling shares, they can sell part or all of the company to many part-owners. The purchase of one share entitles the owner of that share to literally a share in the ownership of the company, including the right to a fraction of the [assets](#) of the company, a fraction of the decision-making power, and potentially a fraction of the profits, which the company may issue as [dividends](#).

**Figure 2** Wiki help facility argument flow (old – replace CW)



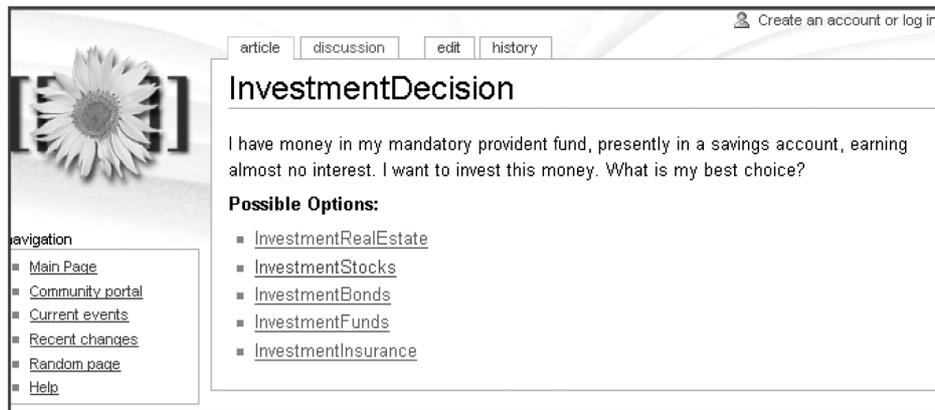
The system consists of pages of different types, question pages, condition pages and explanation pages.

*Question pages* state questions together with a list of potential answer links (multiple options). Options might be sequenced by likelihood or other meaningful criteria. Answer links point to condition pages. See Figure 3 for an example.

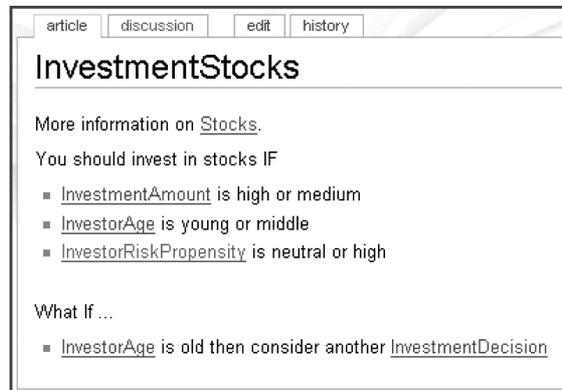
*Condition pages* list the conditions under which the answer is appropriate. Condition pages have an explanation page link (if properly refactored) or contain the explanation of the discussed answer condition on the page. Condition pages also list one or more

conditions that need to be fulfilled. For instance, the page shown in Figure 4 depicts three conditions associated with ‘investing in stocks’ (e.g. ‘InvestorAge is young or middle’). Condition pages link to further condition pages, so as to explore complex conditions in a breadth-first manor. Finally, condition pages have an off-link that points elsewhere, possible to the previous condition or to the next option, in case the condition is not fulfilled. This enables the citizen to explore another condition or the next answer option.

**Figure 3** Question page



**Figure 4** Condition page

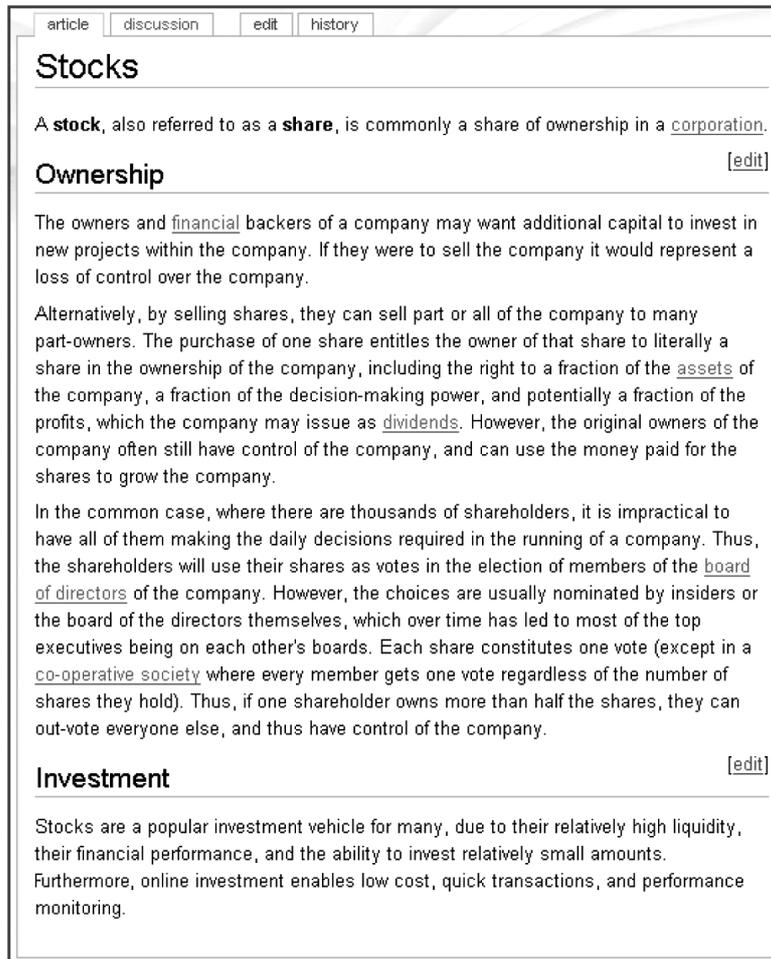


*Explanation pages* provide explanatory content for each complex condition, explaining the concept (answer option) that is reflected in the complex condition. They remain as ‘flat’ documents, possibly with a ‘back’ button to redirect the user to the page that linked to the explanation. Figure 5 shows an example. The document is somewhat similar to that shown in Figure 1, but purposely lacking the conditional logic and question.

A system with this structure, even without an overlaid Semantic Web framework can already be useful as a question answering system or helpdesk system. However, in building a Semantic Web, we envision that such a helpdesk structure is not envisioned by the original page designers, but that a logic of question – condition – explanation can be

overlaid over existing pages, thus allowing a previously not associated set of pages to be meaningfully queried. Such a system can be developed incrementally, adding more content options, conditions and explanations as they are created, or as they are found within the available web content. Semantically, the structure is relatively simple, which also makes it suitable for citizen users to structure Semantic Webs accordingly which could then be machine interpreted and furnished to other users in a more expert system-like fashion.

**Figure 5** Explanation page

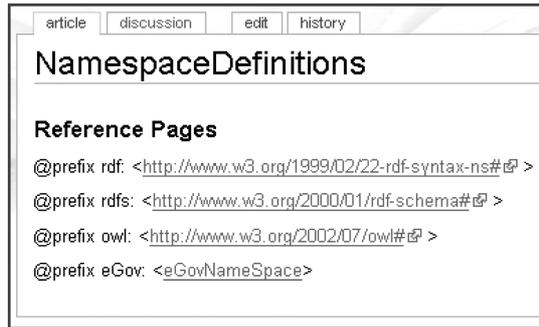


#### 4.2 Semantic Wiki web help desk

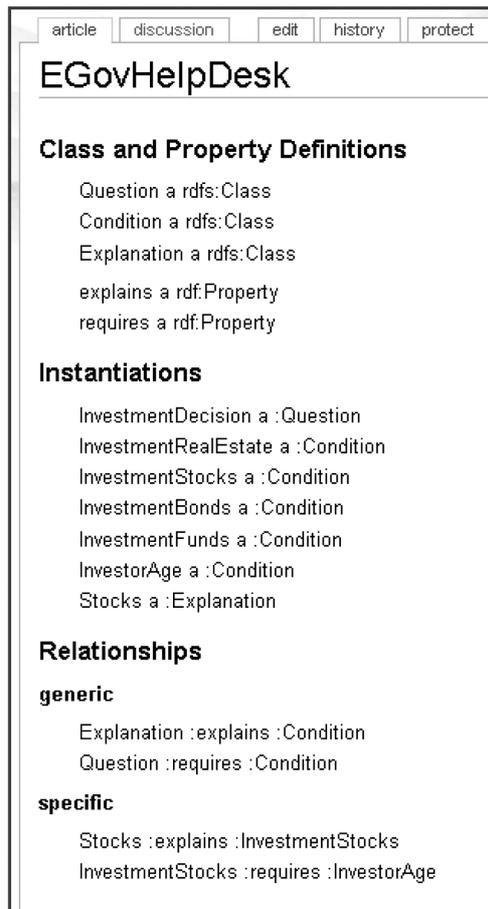
The overlaid semantic Wiki web would identify the semantic relationships, here question – condition – explanation, and explicitly express them in a separate structure, also within easily modifiable Wiki pages that are people- and machine-readable. A suitable organisation would separate name space definitions and 'logic' pages, as

illustrated in Figures 6 and 7. Name space definitions would define the basic terms of the e-government Semantic Web; logic pages would describe the vocabulary and relational concepts.

**Figure 6** Name space definitions



**Figure 7** Defining semantic relationships (excerpt only)



We envision numerous ‘logic’ pages, such as the special purpose page EgovHelpDesk that contains an (incomplete) logic for an e-government help desk (Figure 7). The illustrated example defines the pages as questions, conditions and so on, and furthermore defines the relationships existing between them. Generic rules denote general relationships, while specific rules identify relationships between specific pages. Hence, the rule ‘InvestmentStocks: requires: InvestorAge’, processed by a query Semantic Web processor should furnish the InvestorAge page in conjunction (as condition for) the InvestmentStocks page.

Generic rules, in this example, would result in any explanation page being provided as explanation to any condition page, without selectivity. They would hence only be useful in conjunction with links embedded in the content web. In other words, if the content web depicted a normal hyperlink from page InvestmentStocks to page Stocks, while the Semantic Web identified page Stocks as an explanation and InvestmentStocks as a condition, plus ‘Explanation: explains: Condition’ (see Figure 7), then the Semantic Web query processor could still select Stocks as the most appropriate explanation in the context.

This incremental approach to building Wikis, with a content web layer and a semantic layer, would enable the creation of an incrementally growing system containing the shared knowledge of multiple sources. Thus, a group of helpdesk experts can jointly create a helpdesk knowledge base that covers a wide range of conditions and answers.

## **5 Assessment and discussion**

Previously, we identified seven challenges to the feasibility of Semantic Web development. In this section, we will revisit these challenges and assess how our proposed solution would tackle them, as well as identify new challenges that our solution would create.

### *5.1 Feasibility challenges addressed*

#### *5.1.1 Significantly easier extraction of knowledge*

As the help desk overlay application illustrates, knowledge extraction is made easier as the user can be guided by questions to fulfil his or her specific information requirements. Note that different overlay networks may coexist, allowing each user to select the most suitable overlay network for his or her knowledge requirements. Some may want to query the web as a help desk, while others may want to structure workflows, to name just two alternatives.

#### *5.1.2 Integration of heterogeneous knowledge sources*

As the architecture does not require the knowledge to be stored in a specific format and because the storage format of one piece of knowledge is independent from the storage format of another piece of knowledge, it is possible to integrate heterogeneous knowledge sources under the common interface of an overlay layer.

### *5.1.3 Limit to expertise and non-technical end user contribution*

A key design aspect is the use of Wiki technology that can easily adopted by end users. End user can use the same technology to contribute to both the content web and the – usually hidden and complicated – overlay network.

### *5.1.4 Repair mistakes made in Semantic Web development and to knowledge maintenance*

The participation of users of the community in all levels including the structuring and the organisation of knowledge has several significant benefits. One advantage is that not only the knowledge itself but also its organisation can be adapted continuously according to the users' needs. Unlike in most other systems, errors can be repaired by any users who detect them. Similarly, pages that are difficult to read or use can be improved by exactly those who detect them.

To summarise, one of the main advantages is that Wiki technology used in both layers turns many normal users into experts for some details, and if they volunteer to contribute their expertise, they can significantly improve the entire knowledge base.

### *5.1.5 Ambiguous web*

Ambiguity remains in the knowledge base; however, the choice between multiple independent overlay layers allows the user to select that access path that best fits the user's requirements.

## *5.2 Challenges for the two-layer Wiki web solution and further research directions*

Although we envision the two-layer semantic Wiki web as a beneficial architecture for e-government websites, the following new challenges may arise when this technology is used by a large community of users.

As individual preferences and application behaviours will differ, one challenge is to provide individual users with the most appropriate overlay layer for their task and knowledge needs. There are multiple directions how this challenge can be addressed, ranging from user profiling to ranking of Wiki pages to providing a special search engine for overlay layers over large Wiki structures.

Another challenge is security of and public trust into such a publicly developed and maintained Wiki web knowledge base. This includes quality aspects of specific content, such as for information about investment opportunities. Additionally, it includes transparency of the authorship of a piece of knowledge as well as for the overlay layer that guides users to specific knowledge. Finally, it includes guarantees that such information cannot be falsified. Future research in security and trust guarantees may include search for information and security models which allow for multiple or mixed authorship models that nevertheless provide knowledge users with a transparent minimum of the above-mentioned guarantees. One such model could have the government provide trusted versions of Wikis, with all elements of an overlay structure and the information contained in content pages being government-checked and verified. Obviously, such a solution would lead to several other interesting questions concerning

verification and ongoing version control, as well as to questions on how to enable careful authorship checking and trust management, while still maintaining the ease and efficiency of user contributions that Wikis are made to provide.

Another challenge and therefore a topic for further research may be, how to integrate the Wiki way of organising knowledge with a search engine in such a way that the appropriate mix or integration of overlay layer and search engine can be used.

## 6 Conclusions

E-government encompasses the largest amount of web documents world wide, and is still rapidly growing. The organisation of this knowledge in a user-friendly, efficiently usable and widely accepted way will be one of the major hurdles of the communication between government and the e-government users who are citizens and voters.

Our research suggests that the development of feasible e-government Semantic Webs is as much a technical as a social challenge. The enormous complexity of such webs requires a paradigm shift in the design that enables large numbers of citizens to participate in the development. These are citizens whose technical expertise is not guaranteed. The scenario therefore calls for the creation of a knowledge web, which is easy to develop and maintain. In reviewing the challenges, a two-layer Semantic Web based on Wiki architecture and explicit representation of semantic information offers numerous benefits.

While such an architecture can address several existing challenges, other still remain and new ones emerge. Trading off accuracy of the web and trust in it against broad and easy participation remains an unanswered question. Managing a multitude of semantic overlays may become a task of unmanageable size and complexity; while the design of hybrid search engines that combine content search and Semantic Web analysis creates a new research challenge.

Overall our analysis suggests that the development of e-government Semantic Webs will create a multitude of difficult research questions and application issues for a long time to come. These questions and issues will increase in difficulty, as citizen demands for larger and better e-government sites grow.

## Acknowledgement

This research has been supported in part by CityU SRG Project 7001735.

## References

- Amaravadi, C.S. (2005) 'Digital repositories for e-government', *Electronic Government*, Vol. 2, No. 2, pp.205–218.
- Bergamaschi, S. and Guerra, F. (2002) 'Peer-to-peer paradigm for a Semantic Search Engine', in *Agents and Peer-to-Peer-Computing*, Bologna, Italy.
- Berners-Lee, T. and Miller, E. (2002) 'The Semantic Web lifts off', *Special Issue of ERCIM News*, No. 51, pp.9–11.
- Daddieco, R.J. (2004) 'Retrieving knowledge in e-government: the prospects of ontology for regulatory domain record keeping systems', in Wimmer 2004.

- Fraser, J., Adams, N., Macintosh, A., McKay-Hubbard, A., Lobo, T.P., Pardo, P.F., Martínez, R.C. and Vallecillo, C.S. (2003) 'Knowledge Management applied to e-government service: the use of an ontology', in Wimmer 2003.
- Gaines, B.R. and Shaw, M.L.G. (1993) 'Eliciting knowledge and transferring it effectively to a knowledge-based system', *IEEE Transactions on Knowledge and Data Engineering*, Vol. 5, No. 1, pp.4–14.
- Garcia-Molina, H., Ullman, J.D. and Widom, J. (2002) *Database Systems. The Complete Book*, Prentice Hall.
- Grönlund, Å. (2004) 'Introducing e-gov: history, definitions, and issues', *Communications of the Association for Information Systems*, Vol. 15, pp.713–729.
- Hayes-Roth, F., Waterman, D.A. and Lenat, D. (1983) *Building Expert Systems*, Reading, MA: Addison-Wesley.
- Holsapple, C.W. and Joshi, K.D. (1999) 'Description and analysis of existing knowledge management frameworks', *Proceedings of the 32nd Hawaii International Conference on System Sciences*, pp.1–15.
- Klischewski, R. (2003) 'Semantic Web for e-government', in R. Traunmuller (Ed). *Proceedings EGOV 2003, Springer Lecture Notes No. 2739*.
- Leuf, B. and Cunningham, W. (2001) *The Wiki Way: Collaboration and Sharing on the Internet*, Reading, MA: Addison-Wesley.
- Liebowitz, J. (2004) 'Will Knowledge Management work in the government?', *Electronic Government*, Vol. 1, No. 1, pp.1–7.
- Metaxiotis, K. and Psarras, J. (2005) 'A conceptual analysis of Knowledge Management in e-government', *Electronic Government*, Vol. 2, No. 1, pp.77–86.
- Nonaka, I. and Takeuchi, H. (1995) *The Knowledge-Creating Company: How Japanese Companies Create the Dynamics of Innovation*, Oxford University Press.
- Polanyi, M. (1966) *The Tacit Dimension*, New York: Doubleday and Company.
- Storey, V.C., Chiang, R.L., Dey, D., Goldstein, R.L. and Sundaresan, S. (1997) 'Database design with common sense business reasoning and learning', *ACM Transactions on Database Systems*, Vol. 22, No. 4, pp.471–512.
- Tazzoli, R., Castagna, P. and Campanini, S.E. (2004) 'Towards a Semantic Wiki Wiki Web', *Proceedings of the Third International Semantic Web Conference*, 7–11 November, Hiroshima, Japan.
- Wagner, C. (2000) 'End-users as expert system developers', *Journal of End User Computing*, Vol. 12, No. 3, pp.3–13.
- Wagner, C., Cheung, K., Lee, F. and Ip, R. (2003) 'Enhancing e-government in developing countries: managing knowledge through virtual communities', *Electronic Journal of Information Systems in Developing Countries*, Vol. 14, No. 4, pp.1–20.
- Waterman, D.A. (1986) *A Guide to Expert Systems*, Reading, MA: Addison-Wesley.
- Wimmer, M. (Ed) (2003) 'Knowledge Management in e-government', *Proceedings of the KMGov 2003*, Springer Lecture Notes No. 2645.
- Wimmer, M. (Ed) (2004) 'Knowledge Management in e-government', *Proceedings of the KMGov 2004*, Springer Lecture Notes No. 3035.