

WikiTable: A New Tool for Collaborative Authoring and Data Management

Xianjun Sam Zheng, Ilian Sapundshiev, and Robert Rauschenberger

Siemens Corporate Research
755 College Road East
Princeton, NJ 08540, USA
{sam.zheng,sapundshiev.ilian.ext,
robert.rauschenberger}@siemens.com

Abstract. Tables are an efficient tool for organizing complex data. Even though they are pervasively used in all kinds of documentation, current implementations of tables often limit the power of data management because generally they do not support concurrent collaborative authoring; they only allow keyword search, which typically yields poor search performance; and transporting tables among different applications is cumbersome. We present a new table tool, WikiTable, which permits multiple users to work on the same table simultaneously. The content of each table is stored in a database, which enables accurate data inquiry. More importantly, WikiTable is highly portable, permitting easy integration with other applications, such as Wikis or Blogs. An effort to apply the WikiTable in a global collaboration project of software development is also discussed.

Keywords: Tables, spreadsheet, Wiki, collaborative authoring, data organization, sharing, and management.

1 Introduction

Tables are a powerful tool for organizing complex data. They exploit the structuring of data into two-dimensional arrays, which permit not only the alignment of the data along two separate dimensions, but also make joint membership at intersections of categories along orthogonal dimensions explicit [7,11]. Even though spreadsheet applications (e.g., Excel) are designed with powerful functions for manipulating quantitative data, many tables are created only containing plain texts or qualitative contents. These tables are pervasively used in a variety of documentation, such as contact lists, schedules, project status reports, and software requirement documents. Moreover, these tables are often used by multiple teams at various geographical locations.

However, several major constraints of the current table implementation inhibit the power of effective data management. First, conventional tables do not support collaborative authoring. A user working on a single cell of the table will typically result in the table being locked for concurrent modification by a second user. This inefficiency is exacerbated when a large table requires the input from many users. Furthermore, even

though items in the table are interrelated, they are not semantically coded. As a result, this only permits a keyword search through the entire document, typically yielding poor search performance. Lastly, transporting tables among different applications is cumbersome. This is partially because the table format is mixed with its contents, and there is no single format that universally supports different applications.

Here, we present a new table tool, WikiTable, which permits collaborative authoring, allowing multiple users to work on the same table simultaneously. The content of each table is stored in a database, which also contains the semantic information for each cell item in the table; therefore, this supports accurate data searching and inquiry. More importantly, the WikiTable is a highly portable tool. It can be easily integrated into other applications, such as Wikis or Blogs. This provides a means for the user to access and modify the table contents without concerns about format compatibility. In addition, the user interface of the WikiTable is implemented with AJAX technology, which enables direct manipulation of the table.

In this paper, we first discuss why table is a powerful tool for data organization. Then we review the current implementations of tables and describe the limitations. We elaborate our approach to implement this new tool, WikiTable. We discuss our design considerations in architecture, integration as well as the user interface. This follows by a use case, which shows the benefits of using the WikiTable in a global software development project for document sharing and management. We conclude with a summary and the future directions.

2 Motivation and Related Work

The tabular organization of discrete items – first physical items, then abstract data – has a long history that dates back at least to the middle ages, in which tables (as in: the piece of furniture) with checkered cloth coverings were used to count money [ref.: Wikipedia: "table", 10]. In its modern instantiation – the spreadsheet – the tabular organization of discrete items retained its function in the services of accounting but moved away from the literal "table" to a two-dimensional array on paper. The arrangement of data in a two-dimensional array has inherent benefits that likely contributed to its popularity and longevity: Apart from making category membership explicit, the table (or spreadsheet) permits an immediate apprehension of the logical relationship between categories on orthogonal dimensions by encoding an "AND" relationship that pivots around items that are contained in cells at the intersection of a particular table row and a particular table column. For example, in a table with "size" as one dimension and "vertebrates" as the other, "elephant" will find itself in the cell at the intersection of "large" AND "mammal" because it simultaneously belongs to both of these categories. Similarly, a table permits the immediate apprehension of "OR" relationships, as mammals can be either "large" OR "small;" and large vertebrates can include "mammals" OR "fish" (in the case of sharks, for example).

The table furthermore capitalizes on one of the most basic metaphors used in human cognition: space – the other being time. Spatial layouts are essential for survival (e.g., finding "home" after a hunt); they have served as mnemonics for ages (e.g., method of loci); and they even form the basis of the most common visualization of time: the calendar. It should not be further surprising, therefore, that tables enjoy a

particular and enduring popularity. In an important parallel to physical terrains, where we ascend elevations to gain an overview over the layout of the landscape, spreadsheets permit us to take a step back and appreciate the global pattern of entries, rather than focusing on small details. These global patterns can often be as meaningful as the individual cell entries. In the periodic table of elements, for example, the fact that certain regions of the table remain blank has important implications for the nature of the elements, their physical properties and the principles that govern them.

There are many software products that support the creation of tables, such as VisiCalc, Lotus 1-2-3, Excel, etc. These software products allow users to harness all of the beneficial qualities of tables. In addition, they integrate a multitude of features and functions that are intended to make spreadsheets an even more powerful tool. Modern spreadsheets, for example, can perform text manipulation (concatenation, letter string search, etc.), statistical computations (means, standard deviations, t-tests, etc.), algorithmic operations (addition, subtraction, division, etc.), and many other functions. They plot data in graphs, integrate visual basic and macro functions, and support a host of formatting parameters. Novice users and experts alike use spreadsheets in a large variety of fields ranging from accounting to zoology.

In this wealth of options, what has become increasingly obscured, however, are the basic virtues of tables expounded above. Spreadsheets have become so loaded with features that the applications themselves have bloated into very large software packages with tremendous overhead. Although all of these features are certainly useful (or they would presumably have been pruned in subsequent releases), tabular representations of information in and of themselves already possess considerable value. Under many circumstances, therefore, it may be desirable to have a 'stripped down' version of the spreadsheet used largely for the inherent benefits of two-dimensional data organization. The concept presented in the following is intended as exactly that.

2.1 Wiki and Collaborative Authoring

Although spreadsheets are useful for maintaining, organizing, and even sharing data, current instantiations are less amenable to collaborative authoring. In part, the limitations on cooperative authoring are a consequence of the complexity of the operations that can be performed on the cells of a spreadsheet, which often involve links and references to other cells. References are possible to other cells in the same spreadsheet, across spreadsheets or even to other 'workbooks' that, in turn, contain several spreadsheets, which comprise numerous cells that may have mutual links. If one author alters the contents of a given cell, this modification could result in a whole cascade of changes to the contents of linked cells, which depend on the original cell(s) their contents. Another author, unaware of these altered mutual dependencies, may similarly modify cells that are linked to other cells, resulting in abject chaos. For good reasons, therefore, present spreadsheet applications lock spreadsheets that have already been opened by another user for modification.

The emergence of what has been referred to as Web 2.0 [9] has demonstrated that it is valuable to rely on a 'collective intelligence' to produce content that is the result of multiple users concurrently working on the same document(s). To facilitate and support such collaborative authoring, new online tools have been developed, the most successful of which is the so-called Wiki [4, 10]. All authors work on the same

document, which, importantly, is displayed via the web, so that every co-author has dynamic visual access to the same, shared text. Unlike traditional web authoring using HTML, the Wiki requires no special knowledge of a dedicated markup language, nor does the Wiki require authors to edit the web document offline and upload the result. Modifications can be made directly on the Wiki itself. To a large extent, this mitigates the issue of mutually interfering modifications, as encountered with the current spreadsheet applications, and as described above.

2.2 Combining the Two: WikiTables

Based on these considerations, it occurred to us that the strengths of both Wikis and tables should be harnessed by stripping tables of all the excesses that rendered them into spreadsheets (however useful these may be in some contexts) and combining the two concepts into a single tool – the WikiTable. In the WikiTable, individual cells can be modified independently of one another (see below). In part, this independence can be achieved because cells do not make reference to one another except in the linguistic sense (in the same way this paragraph makes reference to the preceding paragraphs) [11]. This absence of mutual links exists, of course, by design and should be regarded as one of the strengths of the WikiTable. Multiple authors can therefore work on the same table concurrently, each on a single cell at a time. Authoring of the cells occurs in exactly the same manner as with the standard Wiki. The entire application becomes very light and flexible. In essence, a WikiTable can be mutated into any repository of information that fulfills a commonly negotiated, gradually emerging need or goal. Co-authors no longer need to wait until some other user has completed his or her changes to the document; copies of the document no longer need to be disseminated by e-mail to all the stakeholders. Instead, everyone works on the same document, and changes are immediately posted to the same location on the web, visible to all. In accordance with its flexibility, the WikiTable can also become integrated into more formal structures and serve the function of a multi-authored sketchpad within this host application. In this case, in essence, it becomes something akin to a portlet: a dedicated application viewed within the context of a larger assembly, the portal.

3 Implementation

3.1 Architecture

The WikiTable is a web-based application. On the client-side WikiTable relies on web-browsers such as Internet Explorer or Firefox. The JavaScript support in these browsers should be enabled, because it is essential for the AJAX (Asynchronous JavaScript and XML) technology we used. AJAX is a novel web technique that makes a web page feel more responsive by exchanging small amounts of data with the web-server without the need to reload the whole page. In order to utilize the AJAX technique in a more common way, we used a JavaScript library called Prototype[6]. On the server-side the WikiTable requires a database server and a web server with PHP-support. During the development we used a standard Apache web server. We

used MySQL as a database, but because the database design is not sophisticated, other databases could be easily integrated.

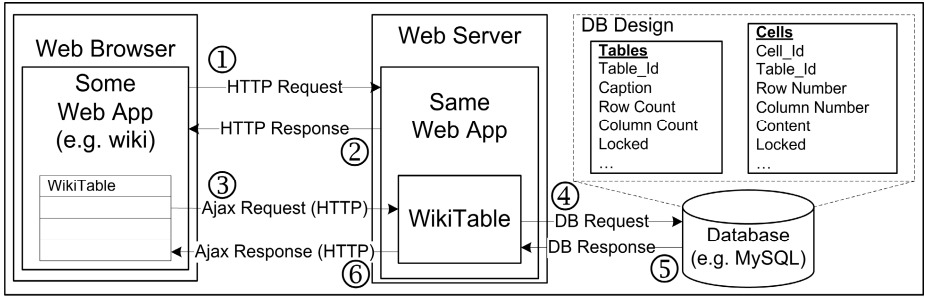


Fig. 1. The architecture and the database design of the WikiTable tool

Figure 1 shows a typical configuration of the WikiTable. The numbers indicate the order of the steps to display WikiTable in a web page. A web browser makes a request (1) to a web server for a web page, which contains one or more WikiTables. The web server returns the contents of the web page (2), which, for now, contains only a reference to a WikiPage. Once the web page is loaded by the browser, a JavaScript function is started. This function uses the DOM-functionality to find all the html-elements having the following structure `<div id="wt_?????">some text</div>` and extracts the following string: "?????". This string is the WikiTable's unique ID in the database. For every ID found, an AJAX request (3) is started. The server-side script of the WikiTable receives the request, checks if the ID is correct, and holds the appropriate data from the database (4 and 5). Once the WikiTable script has received the data, it constructs the HTML-Table-code, which is then sent to the browser (6). A Javascript function injects this piece of code in the appropriate place on the web page.

A key feature of WikiTable is to support collaborative authoring. To achieve this, we increased the granularity of the locked content while editing, from the usual whole-table granularity to a single cell. This makes a table editable by many users simultaneously, as long as every user works on a different cell. This feature has shaped the database design of the application (see Figure 1, the dashed rectangle in the top right position), which basically consists of two database tables. The first one contains metadata for every WikiTable, such as its caption, unique ID, number of columns and rows, and other information. Every row of the second database table represents a single cell in a WikiTable. It contains, among other things, the unique ID of the cell, its contents, the unique id of the WikiTable to which the cell belongs, and its position in that table (i.e., the number of the row and the column).

3.2 Integration

Another important characteristic of this tool is its easy integration with currently popular web applications, such as Wikis, Content Management Systems, Forums and Blogs. By using AJAX-technology, we implemented WikiTable as a thin, lightweight layer that could be easily integrated in the aforementioned types of web applications.

Currently, we have a MediaWiki integration. MediaWiki[5] is the wiki engine serving the popular Wikipedia Project.

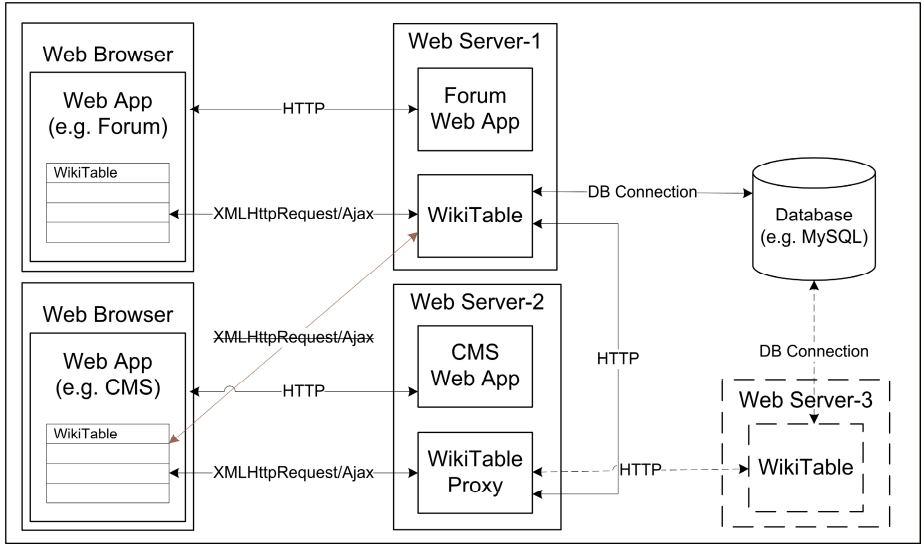


Fig. 2. The architecture of the WikiTable integration with different web applications

Figure 2 shows an integration of the WikiTable tool in two web applications, running on different web servers. Owing to security restrictions, most of the current web browsers do not allow their Javascript implementation to make a connection to a web server located on a different domain than the original one. There is a workaround for this restriction, which is known as Cross Domain AJAX Calls, and which is discussed in [8]. This is the reason we utilized a script called ‘WikiTable Proxy’ on every further web server that hosts web applications with WikiTables. This allows us to manage all the WikiTables from a central place, which, in Figure 2, is web server 1. An optional configuration could be the use of a dedicated web server (web server 3 in Figure 2), which hosts the WikiTable tool and manages all the requests from the WikiTable proxies hosted by web servers such as web server 2 in Figure 2.

3.3 User Interface

The user interface design is at a very early stage. Figure 3 is a screenshot of the current user interface; here, the WikiTable is integrated in MediaWiki. Area number 1 in the Figure is the title of the WikiTable. The two buttons following the title are: [e] Editability, toggles the editablilty of the table; and [o], New Window, displays the table in a new window (e.g., for printing). Every column and row of a WikiTable has its own header (areas 2 and 6), which is only visible in the editing mode. In the column header, clicking '<<' will switch the position of the current column with the previous one, '-' will remove the column, '+' will add a new column after the current column and '>>' will switch the current column with the next one. By clicking on a cell

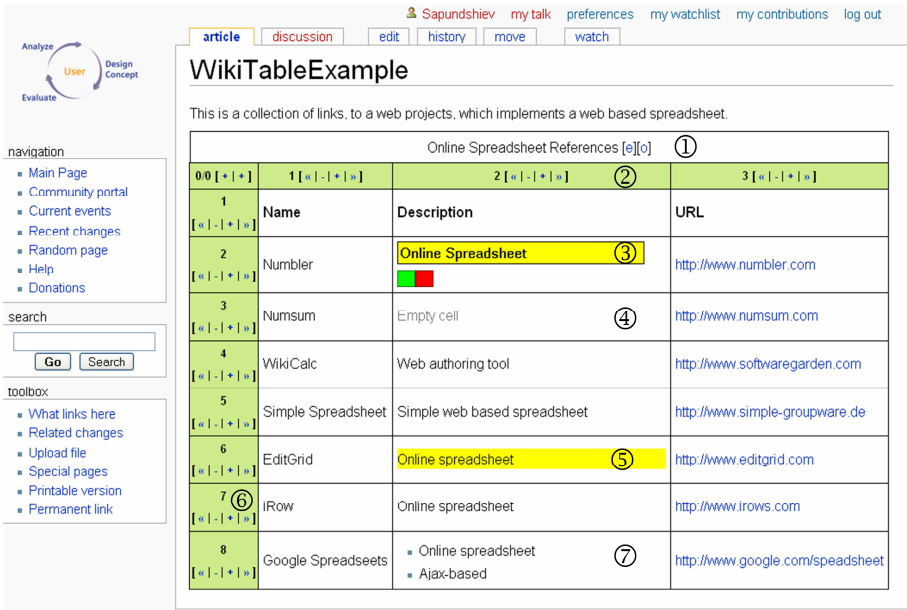


Fig. 3. An example of the UI of WikiTable integrated with the popular wiki engine, MediaWiki

from the table, the contents of cell become editable (cp. area 3). This functionality is known as “edit in place.” Clicking the green rectangle confirms the changes, the red one discards them. Cells with no content display a grayed out text, “Empty cell” (cp. area 4). Area 5 shows a cell with a yellow background; this denotes that the current cell is currently being edited by somebody else. A cell may contain MediaWiki syntax as well; this is the case for the cell labeled as area 7. Here we used “*Online spreadsheet *AJAX-based” to achieve the bullet points.

All the described manipulation happens dynamically while utilizing the AJAX functionality, which we mentioned in the Architecture section. No reloading of the hosting wiki page is needed (cp. steps (3)-(6) from Figure 1).

4 Application Scenario

WikiTable can potentially be used in many application domains that require collaborative documentation sharing and management, such as software development, medical information management, and building information management. A good example is the Global Studio Project [2], which was initiated by Siemens Corporate Research. This project comprises six development teams located around the globe (US, Germany, Brazil, China, etc.). A Wiki site has been set up to facilitate information sharing and collaboration among these teams. Many documents, such as project plans, status reports, software requirements, etc. are organized in tables. However, without the WikiTable, the coordination of data input and editing among multiple users is

challenging. Because there are thousands of documents, finding the right item in the table based on a keyword search is rather time consuming. There are also situations in which the contents of the tables needed to be used in other applications, such as a Content Management System, and current table implementations make transporting the table quite cumbersome. Yet, these challenges are easily resolved with the WikiTable. Currently, the application of the WikTable into this project is ongoing.

5 Conclusion

In sum, we have developed the WikiTable tool and also demonstrated its capability to support collaborative authoring and data management. We have developed the architecture, the integration framework, as well as the preliminary UI for WikiTable. There are a number of additional features that need to be implemented in order to make the WikiTable a mature tool. For instance, the tool should support table export and import (e.g., XML format), and the table should have some basic spreadsheet functionality.

In addition, a user validation session is also necessary to examine the effectiveness of the WikiTable.

Acknowledgments. We thank Claus Knapheide for his support on this project.

References

1. Bachmann, F., Merson, P.: Experience Using the Web-Based Tool Wiki for Architecture Documentation. Technical Note CMU/SEI-2005-TN-041
2. Ducasse, S., Renggli, L., Wuyts, R.: SmallWiki: a meta-described collaborative content management system. In: Proc. of the 2005 WikiSym, pp. 75–82. ACM Press, New York (2005)
3. Haake, A., Lukosch, S., Schümmer, T.: Wiki-templates: adding structure support to wikis on demand. In: Proc. of the 2005 WikiSym, pp. 41–51. ACM Press, New York (2005)
4. Leuf, B., Cunningham, W.: The Wiki Way. Addison-Wesley, Reading (2001)
5. MediaWiki – <http://www.mediawiki.org>
6. Prototype – JavaScript Framework <http://prototype.conio.net>
7. Silberhorn, H.: TabulaMagica – An Integrated Approach to Manage Complex Tables. In: Proc. of DocEng, Atlanta, Georgia (2001)
8. Use a Web Proxy for Cross-Domain XMLHttpRequest Calls <http://developer.yahoo.com/javascript/howto-proxy.html>
9. What Is Web 2.0. <http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html>
10. Wikipedia – <http://www.wikipedia.org>
11. Xia, S., Sun, D., Sun, C., Chen, D.A.: Collaborative Table Editing Technique Based on Transparent Adaptation. In: OTM Conferences pp. 576–592 (2005)