Collaborative ORM Data Modeling: Educational Experience using a Wiki

Gordon C. Everest
Professor Emeritus of MIS & DBMS
Carlson School of Management, University of Minnesota
geroverest@umn.edu

Abstract. This case study reports on a classroom experience using a Wiki to design an ORM data model. Student teams developed ORM diagrams and were to present them in a top down unfolding fashion along with an accompanying narrative description. Data modeling is typically a group effort involving several user domain subject matter experts. No one individual knows everything, but collaboration can capture their collective knowledge. In a large metropolitan area, the students were widely dispersed and most worked full time. This made face-to-face meetings difficult. Increasingly in practice, data modeling is done by virtual teams where members cannot meet to generate and review design ideas. Hence, there is a need for tools and an infrastructure to facilitate collaboration in virtual teams. Wikis have been used successfully for mass collaboration. Students are increasingly likely to encounter Wikis in their working world. Hence, the purpose of the assignment -- use a Wiki to develop and document an ORM data model. While there was considerable variation in the organization of Wiki pages across four teams, the most striking similarity was the inclusion of a Wiki page to collect elementary facts. This would attest to the central importance of this step in any data modeling effort, to gather information requirements and business rules. However, this classroom experience showed some serious limitations using a Wiki for small team collaboration in which there is some specific end product. The result goes beyond just the documentation itself to produce a well-defined database.

1. PREAMBLE

This paper explores the suitability of a Wiki to support virtual teams doing ORM data modeling, first in an educational setting, then with some discussion of what might be different in a real world working organizational environment. This paper does not purport to have all the answers, nor even suggestions for best practice. It simply reports on one experiment to use a particular collaboration mechanism -- a Wiki -- and some suggestions for future such projects.

Darleen DeRosa put forth one of seven key challenges to success in virtual teams: "Technology makes virtual teaming possible, but isn't a perfect substitute for human interaction. Teams must be careful to use the appropriate technology for various tasks." [Dempster, 2005]. This paper explores the appropriateness of Wiki technology for ORM data modeling.

Feedback and other ideas are welcome. Hopefully, this paper will spark some discussion as we try to find better mechanisms to support collaboration in virtual teams to do data modeling.

2. BACKGROUND

Last year, I attended a hands-on workshop "Introduction to Wikis." I had few expectations except perhaps that I would learn something about Wikis. Wikis are increasing in popularity, most notably Wikipedia. I wanted to find out what was really different here. While listening to the instructor, I realized that a Wiki might be a good way to facilitate team development of a data model. Subsequently, I got hold of Wikinomics: How Mass Collaboration Changes Everything by Don Tapscott and Anthony Williams. It opened my eyes to a future my students would be facing. I owed them some sort of experience with a collaboration tool.

Several years ago, after many student complaints, I dropped the team data modeling project in my database design class. It was simply too difficult for the students to get together for face-to-face meetings, being geographically dispersed in a large metropolitan area, and most of them working full time. Minnesota is the largest university in the United States and over 90% of the students commute to campus. Both time and space (geography) are significant barriers to face-to-face group meetings.

With the availability of collaboration tools such as Wikis, it was feasible to bring back the group project, at least on an experimental basis. This year I have begun to develop an online version of my database design class. If there is to be a data modeling group project, some collaborative mechanism is required to support virtual teams.

3. THE DATA MODELING IMPERATIVE: VIRTUAL TEAM COLLABORATION

WHEREAS: The objective of data modeling is to produce a single, consistent, comprehensive, unified, and accurate data model of some portion of the real working world of interest to a community of users.

WHEREAS: In practice, data modeling is a group effort -- seldom, if ever, done by a single person in isolation. Multiple people -- user domain subject matter experts -- often work with a database design "expert" or facilitator. No one individual knows everything about the subject domain. Our objective is to capture their collective wisdom, resolving any conflicting perceptions through some form of negotiation and mediation by others with domain knowledge. Such design can then be the basis for implementing a database (define and populate), perhaps in multiple stages.

WHEREAS: Domain experts are often geographically dispersed, and hence must form virtual teams. Also, they likely have time commitments and work schedules that make it nearly impossible to get together for face-to-face meetings, even if they were co-located in one place.

WHEREAS: Virtual teams require some collaborative infrastructure of technical support for the team effort. Recently we have witnessed the rapid emergence of many types of collaborative tools on the World Wide Web. In particular, Wikis have been used successfully for mass collaboration.

WHEREAS: A student's educational experience should be consistent with the working environment of IS/IT professionals, and prepare them for what they can expect to encounter in practice. The reality is: Students will find themselves working in teams, and they will find themselves using increasingly sophisticated collaboration mechanisms.

==> THEREFORE, to be properly prepared for such a working environment, students must have an educational experience in a data modeling team project using some collaborative tool.
4. TYPES OF COLLABORATIVE MECHANISMS

To facilitate collaboration in virtual teams, several tools and technologies are available, ranging from traditional to more recent innovations using the Internet.

* telephone -- synchronous; audio; point to point (poor support for a team effort)
* telephone voice mail (answering machines) -- asynchronous; point to point.
* telephone conference call -- synchronous; group; audio communication only.
* video conferencing -- synchronous video+audio group communication, capture non-verbal cues.
* mail/FAX -- asynchronous written communication.
* email -- asynchronous written communication, with file attachments.
* (threaded) discussion board -- asynchronous, but more "interactive" written communication.
* chat room -- synchronous, written communication.
* blogs -- asynchronous; essentially one-way communication, from one to many.
* "narrow"cast mechanisms, such as listserv or groups -- asynchronous; many-to-many.

All of these, used in any combination, are intended to support communication when people are not face-to-face. Of the traditional mechanisms, video conferencing (generally using the Internet) is the only one that simulates face-to-face interaction. Most people have some idea of how these various types of collaborative mechanisms are used (see Appendix A for a more detailed discussion).

None of these mechanisms has direct support for creating, reviewing, and revising a single, common, integrated end product of the team effort, such as a database design document, nor does any directly support the process of producing the end product.

4.1 Software Tools for Virtual Team Collaboration

With the World Wide Web, we are able to go beyond some of the more traditional collaborative tools, overcoming some of their limitations and more directly supporting the work of virtual teams. The use of newer collaborative technologies can help virtual teams become as efficient and effective as co-located (same-location) teams.

Software that aids virtual teamwork encompasses two main functions -- providing ease of communication, and providing task [process] and document [product] organization [Wikipedia, "Virtual Team"]. Both of these functions would be part of a comprehensive virtual team collaboration support system. Software facilitating communication includes features such as presence announcement, instant messaging, chat rooms, and web conferencing (supplementing telephone and video conferencing). With these tools, team members can make themselves available anytime. Members can have real time conversations and do not have to correspond back and forth as with some of the more traditional asynchronous mechanisms.

Software systems to organize team tasks and documents provide a central location where members can access documents, track progress, assign tasks, and provide calendars with key dates, timelines, and milestones. One example is "group" software from Yahoo. With Google-docs, team members can be online at the same time and all edits appear in near realtime, since the documents are refreshed every few seconds.

In the context of database design, the goal of the team is to develop a single end product - a data model with supporting documentation. This goes a step beyond many of our collaborative tools available today. Wiki is one example of a new breed of collaborative mechanisms.
4.2 Characteristics of a Wiki

The distinguishing characteristic of a Wiki compared to other virtual team collaborative mechanisms is: the create/edit function is decentralized to a population of peers. This is made possible by storing a single version of the content on a common server. All users access the Wiki content using a client web browser. Use of a Wiki can avoid the traditional cycle of writing, disseminating it to others for review, and integrating the feedback from multiple reviewers into a new version of the original document. A Wiki essentially provides the readers/reviewers the ability to actually edit what they are reading at that very moment, while it is fresh in their mind. This can be very powerful, and motivating. Knowing that they can revise what they are reading, they are more likely to read more critically. The reader can think beyond what they are reading, similar to brainstorming except that it is done asynchronously, at each viewer's convenience.

Theoretically, a Wiki is open to the whole world for anyone to read and revise add to the content. In practice, however, a Wiki is often limited to a specific population. While everyone in the population has some responsibility for the extent and quality of the Wiki content, it is not unusual to designate one member or a small group to review and approve all changes and additions to ensure consistency and accuracy. In a data modeling context, the population would be limited to those with some knowledge and experience in the subject area being modeled, including any facilitators.

An important characteristic of a Wiki is the historical record of all revisions. This makes it possible to track changes, additions, and deletions, and to recover any content that was inadvertently or deliberately deleted or modified. In an educational setting, it also allows the instructor to assess the degree of timely participation by each student. Likewise, it allows students to view the relative contributions of other team members. In the team dynamics, it may encourage the more passive students to step up and the more aggressive students to hold back and allow others to take some responsibility for the work.

In the organization of Wiki webpages, it is wrong to think of them as arranged in some sort of hierarchy. This is a common misunderstanding, especially since we are so conditioned to seeing hierarchies, such as the directory of folders on a computer. Thinking hierarchically leads to confusion. The organization of a Wiki must be thought of as a web of pages, where any page can point to any number of other pages. A Wiki web starts out as a single webpage, and, as webpages are added, it grows to include many pages all linked together. (See Appendix B for additional detail on the Characteristics of a Wiki Environment).

5. THE DATA MODELING ASSIGNMENT

My objective in giving this assignment is to prepare the students for what they are likely to encounter in a professional work setting. I also attempt to make the assignment as realistic as possible and avoid the artificiality of the classroom setting as much as possible.

The students were given one large database design problem for Holly Hotel. [Everest in Hunter] The database was to support the operation of a large hotel complex. The problem was broken down into four subareas and assigned to four teams, focusing on the basic operations:

1. Rooms and Facilities (Sleeping and Meeting)
2. Customers (Guests, Hosts, and Billed Parties)
3. Reservations (for Rooms/Facilities by Guests/Hosts for an Event or Visit)
4. Availability, Assignment, and Usage (of Rooms/Facilities to Guests/Hosts)
The assignment excluded personnel, inventories, non-rentable rooms, food service, housekeeping service, and financial accounting (general ledger, payroll, purchasing, accounts payable, vendors).

The learning objectives included:
* practice designing databases using ORM.
* using an ORM data modeling tool (VisioEA in Microsoft Visual Studio.NET).
* dealing with a realistic problem (large and complex, with a fuzzy, incomplete, and ambiguous statement of the problem and system requirements, such as you might get from users and managers in a real organizational context).
* working in virtual teams.
* collaborating using a Wiki (the University of Minnesota Wiki infrastructure).
* each team integrating their design with the designs of other teams.

The project assignment spelled out the required deliverables in some detail:
* Model diagram as produced by the design tool, manually annotated as necessary, and presented in a top down fashion unfolding successive levels of detail.
* Narrative to accompany the diagram(s) with overview and assumptions.
* Selected output reports from the design tool.
* Relational tables generated from the ORM data model and discussion of implementation considerations.
* Three feedback memos on: the project, use of the Wiki, and the ORM data modeling tool.

6. CONDUCT OF THE DESIGN PROJECT

The students were a mix of undergraduate seniors majoring in MIS, and working professionals in Information Systems. Being an advanced database design class, students were expected to have some experience setting up and using a database with a DBMS. Many of the students, particularly the practitioners, had substantial experience with databases.

The class used the Wiki infrastructure provided by the University of Minnesota based on TWiki. We set up a Wiki web for the whole class and one for each team. A Wiki web starts out as a single webpage, and, as webpages are added, it grows to include many pages all linked together. Each webpage in a UMN Wiki is called a Topic.

To get them started using the Wiki, students were asked to create and populate a group bios topic page to share personal information (contact information, background, and work schedules). This gave them something easy to do while learning how to access the UMN Wiki.

After receiving the project assignment and the problem description for Holly Hotel, students were asked to use the class Wiki to state their preferred subareas to work on. This gave them another opportunity to use the Wiki. Teams were formed to balance the background and skill level of the students. All but one was assigned to their first or second choice of subarea. Then we set up a separate Wiki for each team and permitted each student to access their own team's Wiki. Each team consisted of three or four members.

Initially each team would have access to its own Wiki only (and the class Wiki). The idea was to provide a little privacy as they initially explored what to do. Floundering and false steps would be hidden from members of other teams. This gave them some freedom to make mistakes as they figured out what to do, how to structure their work, and begin creating content.
Then on some designated date, the individual team Wikis would be opened up to the rest of the class. The initial closed period would give the team a chance to develop a preliminary design for their assigned part of the problem. Once opened, other teams could provide feedback, comment on, and actually make changes to anything contained in their webpage or any part of their design. Opening up the group Wikis is like opening the preliminary design to a design review.

After having an opportunity to produce a preliminary design, and seeing the preliminary designs of the other teams, there was a "meet the user" session where they could ask questions of the CEO of Holly Hotel (actually the instructor).

Then toward the end of the project time period, the individual team Wikis would be closed, giving each team an opportunity to finalize their deliverables for submission without interference from the other teams.

Separate team submissions was the one artificiality introduced in the educational setting. In a corporate setting, the individual sub-models would have to be integrated into a single, comprehensive model, which would then be the basis for implementation. The need to assign grades based on individual (or team) performance precludes completing the integration process and grading a single submission (for the whole system) from the class. To increase the realism of the assignment, each team was told to consider their submission a preliminary design. (See Appendix C for a deeper comparison of the Educational vs. the Business / Organizational Settings).

7. OUTCOMES and OBSERVATIONS

Preliminary results from this initial educational experience using Wikis were mixed. Hopefully, with what we have learned from this initial experience, subsequent efforts can be more effective. This will also provide some useful guidance to others undertaking a similar collaborative virtual team effort to do data modeling, perhaps using other collaborative mechanisms.

7.1 Overall Comments

The use of a Wiki was new to everyone. Consequently, the students had to learn about Wikis while they were also learning about the project assignment and the ORM design tool (and the course material, and some small ORM design exercises along the way). While we had Usage Notes for the ORM design tool (VisioEA), Usage Notes for the Wiki would have helped to jumpstart students. Nevertheless, several students commented that they appreciated the opportunity to learn about Wikis. Some of their comments in the feedback memos:
- "great learning experience for teamwork"
- "Wiki technology is a great fit for a group assignment"
- "our team was synergistic, able to accomplish far more together"
- "Once we got going, I felt we turned quite productive. It was just in getting it started."

7.2 Face-to-Face Meeting

Once the students were assigned to teams they were left on their own to figure out how to work together. Given that the class itself was a weekly face-to-face meeting, it was possible for the team members to make contact and touch bases to arrange meetings and followup on tasks, etc. However, this seemed insufficient. Perhaps it would be helpful to set aside some class time immediately after assigning students to teams. This would provide an opportunity for the groups to meet together and get to know one another better. They could exchange contact information, discuss strategies for tackling the assigned problem, and how the team members will work together. Periodic in-class team meetings after that might also be helpful for team functioning. Face-to-face meeting may not be possible in a purely online learning environment.
7.3 Where and How to Start Using a Wiki

With the Wiki, the students did not know where to start. This was true across all four teams. As one student observed, "We were given the project and given the Wiki and were told to use the Wiki to do the project with little grasp of how the Wiki could help us to do the project." Another said, "The biggest problem to getting work started on the project was getting used to the Wiki."

A bare Wiki is so freeform, that there is essentially no guidance for how to set it up, no outline or structure for this particular type of task. A Wiki starts with a single home webpage. From there, participants create additional webpages, adding content and adding links to additional pages they create. While the Wiki infrastructure may have great potential to facilitate collaboration in virtual teams, students were lost. They had many resources explaining what a Wiki was and the mechanics of how it worked, including extensive online documentation, and a class presentation and demo. However, a Wiki is so freeform that the students had trouble figuring out how to get started and use it for the task at hand. Much more guidance is needed.

It would have been helpful for students to have one or more suggested templates for the types and structure of the webpages, at least as a place to start. Using a Wiki was initially considered part of the learning experience. However, without some idea of how to set up the Wiki for a data modeling project, using a Wiki is daunting for the novice.

7.4 Early Generation of a Page of Elementary Facts

The most striking similarity across all four team submissions was the inclusion of a webpage to collect many elementary facts. This was quite gratifying to see. Without any prodding, the teams all started with the first step -- verbalize -- in the conceptual schema design procedure (CSDP) [Halpin, 2001, p. 58ff]. This reinforced the notion that verbalization is an important, and natural, first step in developing ORM data models, indeed, data models using any data modeling scheme. With a reasonably complete set of elementary facts, an appropriate tool can render them in a diagram for human review, and then generate the relational data model for implementation in some target DBMS.

7.5 Using the Wiki for Communication

In most cases, the participants did not use the Wiki for communication among themselves. They preferred to use the familiar traditional modes of communication: mobile phone, email, or an online (threaded) discussion board. It was just one more place that needed to be accessed to see if there was anything new there. The online discussion available to the students was not used much either, probably for the same reasons the Wiki was not used much for communication, yet another place to have to check. People are so used to using email to communicate, collaborate, and share files, that, even if they started by using the Wiki, most reverted back to email and sharing design files as attachments. The Wiki seems to have little or no value added (and even a potential hindrance) when used for communication. One student commented that group software, such as from Yahoo or Google, "facilitated communication better than a Wiki."
7.6 Using the Wiki as a Repository for Storing and Sharing Files

Several students commented that the Wiki was most useful for storing shared files, rather than emailing files back and forth. In fact, one team had none of the deliverables stored in the Wiki; they were all stored as individual attached files. A second team did not even store their deliverables as attached files in the Wiki -- they only embedded links to the files which were stored on a completely separate server maintained for general student use. A third team had separate Wiki web pages for each of the deliverables. A fourth team actually incorporated all the deliverables into a single Wiki web page, making it particularly easy to obtain a hard copy of their submission. The ability to store attached files on the Wiki server was especially useful since the diagrams and documentation from the design tool were in their own files, and would have to remain an attached file in order to be opened with the tool by the recipient. Each student had their own copy of the data modeling tool software.

7.7 Using the Wiki to Monitor Progress

Several students commented that the Wiki was a convenient way to monitor the progress of the team effort. It provided a "visual representation of the project's completion status." "It was useful as a visual representation of what had been accomplished" as well as what remained to be completed.

7.8 Not Meeting Timelines

Not surprisingly, most students failed to meet the suggested time deadlines for various tasks. More than half the class missed the suggested timelines for entering their bios, starting a list of tasks, and setting up pages to store the deliverables. For the "meet the user" session, most students were not well prepared, which would have meant having carefully read the case problem statement, developed some initial thoughts, and even produced a preliminary design. Thus, when the individual team Wikis were opened up to the rest of the class there was little substance to review. Interestingly, they all met the deadline for final submission, since getting a grade depended on that. Student feedback included:
- "Not enough weight put on participation"
- "Use of the Wiki will not take off unless enforced"
- "Needed some incentive through rewards"

Some participants were eager to get started as soon as they were assigned to a team (the tasks and deliverables having already been defined in the project assignment). That would lead to frustration and discouragement when their other team members did not respond. One student asked for the email addresses of the other team members so he could contact them directly. This information should have been on the Wiki bios topic page but was not.

7.9 Not Closing the Individual Team Wikis at the End

The students expressed strong sentiment for keeping the individual group Wikis open at the end of the course. Closing the group Wikis would set an arbitrary deadline for looking at the other teams' designs. It was generally felt that nothing would be gained by closing them again, and that nothing would be lost by leaving them open. In the final days, the teams would be concentrating on preparing their own final submissions, rather than concern themselves with what the other teams were doing. I agreed with them and left the Wikis open to the end.
7.10 Overlap in the Subparts

The database designs of the individual teams had considerable overlap. This is not surprising since the various functions in each of the teams will involve the same underlying entities. For example, while the subareas of Rooms/Facilities and Customers are relatively independent, Reservations involves Customers making reservations of Rooms/Facilities. The last group involved the assignment of specific Rooms/Facilities to Customers and their events, which would determine Usage, and which would in turn be used to determine Availability for new assignments and Reservations. There are dependencies here which would suggest a phased approach, working on Rooms/Facilities and Customers first. This is most likely what would happen in practice. One approach would be that the dependent groups (3 and 4) do not try to design the independent subparts (1 and 2) but rather simply ensure that the information they need is included in the designs of the independent subparts. In an educational setting, we are limited in the ability to have subarea designs done in phases, since all the teams need to be working in parallel. Overlapping parts of the data models presents a particular challenge to the individual groups as they undertake to develop their own designs in parallel.

8. USING A WIKI FOR DATA MODELING PROJECTS: Conclusions & Recommendations

Based on the outcomes and observations, we can draw several conclusions and make some recommendations when using a Wiki for a data modeling project.

A Wiki can be used to contain material relating to both the process and the products of data modeling. Process information includes background information, participant contact information and their calendars (though organizations often have separate systems for maintaining individual and event calendars), organization of the team, roles and responsibilities of the team members, task lists, timelines, completion status of tasks and deliverables, intermediate products, outline of deliverables, discussion and communication among team members, and a place to store shared files and reference materials.

Product information includes the final deliverables for the project and any supporting materials. For a data modeling project, this would include data model diagrams, accompanying narrative description, assumptions made in developing the data model, output reports from the data modeling tool, generated relational tables, implementation considerations, and any memos associated with the project. As one student observed, the Wiki was "useful for packaging up the work product ready for submission."

8.1 Providing some Initial Structure to Set Up a Wiki for Data Modeling

Just as students needed some guidance for how to initially structure their Wikis, the same would be true in an organizational setting. Preparing some sort of template -- ideas for the types of topic pages to create in the Wiki and how they should be organized -- would provide a very useful starting point for a virtual team charged with a data modeling task.

In an organizational setting, a Wiki can harness the power of collective wisdom to gather facts, but subject matter experts are needed to review and validate those facts. Unfortunately, the totally freeform nature of a Wiki runs counter to the need for some pre-specified structure to organize content, a fundamental principle of data modeling [Chisholm, 2007].
8.2 Importance of an Initial Face-to-Face Meeting

DeRosa [in Dempster, 2005] emphasizes the importance of building familiarity and trust among team members. An initial face-to-face meeting can be invaluable to create a bond between members and gain an understanding of each other's individual makeup. Subsequent, periodic face-to-face meetings can also help to sustain and strengthen team interaction, even if only a couple of times a year.

In a purely online learning environment, or in widely dispersed teams, face-to-face contact may not be feasible. However, it would help to have an initial video conference for team members to see each other and pick up on the non-verbal cues in their communication. This could be done using video conferencing or webcams.

8.3 Need Incentives for Timely Participation

In retrospect, if there are no meaningful rewards (or penalties), most students will delay doing something until the last minute. That seems to be the nature of an educational environment. The project ought to be broken down into little steps with points given for meeting each deadline (and an increasing penalty if they don't). The ability to assign points which will be used to determine the final grade is actually a strength of the educational environment.

This level of refinement may not be needed in practice. A student's goal is to complete the course, satisfying all the requirements, and, getting a good grade. However, in an organizational setting, the objective is to complete the assigned project and to do that well. We often set deadlines for intermediate tasks/deliverables but it is difficult to come up with appropriate sanctions when those deadlines are missed.

8.4 Notification of Changes to Wiki Content

Perhaps the real reason the Wiki was seen as not helpful for communication is that participants don't receive notice when something has changed or something new has been added. When you go into your email, you know immediately if there are any new messages waiting. In fact, many email systems will produce a sound or popup when new messages hit your inbox (and you have the email application running). Some say "You've got mail" or "16 New Messages" when one logs on to the system.

Though evidently none of the students found it, the UM Wiki does, in fact, allow you to sign up to receive notice of changes to the Wiki web. With the subscription service you are automatically notified by email when topics (individual webpages) change. The email contains links to all of the topic pages modified since the last alert. You can sign up for all Wiki web pages or just the ones you wish to monitor, or list exceptions when there are some pages for which you do not wish to be notified of changes. You can choose to receive the complete topic webpage, or just a change summary. This is a very comprehensive and flexible service. If the students had discovered and used it, perhaps it would have become the primary mode of communication among team members. This experience reinforces the need for usage notes to help the students get in to using the Wiki.
8.5 Creating, Formatting, and Manipulating Webpage Content

Creating was more difficult than reviewing, editing, and revising webpage content. Perhaps this was due to the unstructured nature of a Wiki. A Wiki seems better for reviewing than for initial creation of material or designs.

The formatting capabilities using the Wiki markup language were similar to those generally available in HTML. Consequently, they were difficult and awkward to use, as several students noted. It was also difficult to move pieces of content around, using drag and drop on selected portions. It would be much easier if the formatting, editing, and manipulation were like those in a word processor, such as Word or WordPerfect. Some found it easier to maintain text files in their word processor and simply email them as attachments. This defeats a major strength of a Wiki.

When a participant is reading/reviewing some content, there is a need to distinguish revisions from comments, as is generally possible in the more advanced word processing programs. In this way, a team could retain control over revisions, but would be able to receive comments from a wider population on specific content at specific points in a webpage. This is important when the team will be evaluated on the basis of their "final" product. They want and need control over revisions as they enter the final stages.

8.6 Linking the ORM Model Diagrams with the Wiki Content - Version Control

When a Wiki is used for ORM data modeling, the product deliverables are more than just formatted text. The deliverable also includes data model diagrams from the data modeling tool. In addition, it would implicitly include any reports which the tool could produce based on the model.

For some teams, the ORM diagrams were integrated (copied) into the webpages, so that one could see and print directly the entire deliverable for the team without having to open attached files. This would render the diagrams read-only. At the other extreme was a team that included all the deliverables in files attached to one webpage. More usual was to include all the narrative content in Wiki webpages, and to attach ORM diagrams to those pages. Then it would be possible for a student to open the attached ORM diagram file with the data modeling tool, and possibly revise it. On the other hand, if a recipient did not have the appropriate design tool installed on their computer, they would be unable to open any attached diagram files. This would argue for copying the diagram onto the Wiki webpage, and attaching it as a file to the webpage.

Sharing ORM diagrams as attached files raises an interesting problem for concurrent update control. If one team member opened up a model diagram with the design tool (every student had their own copy of the software), made some revisions, and then posted the revised model to the Wiki, there is nothing to say whether or not another team member would be doing the same thing. Now there are two versions of the same model. Integrating the changes in both versions could be very difficult. In the absence of any system capability, each team is responsible for ensuring that concurrent updates do not happen, that only one person at a time be allowed to make revisions to a model. This brings us back to pre-Wiki mechanisms with the same inherent problems.

Even if the diagrams are integrated into the Wiki webpages there is a problem. The model diagrams cannot be manipulated directly on a webpage (in HTML). They must be in the file type used by the design tool. Hence, if a user wanted to make revisions to a model diagram, they would still have to get the latest file from some central library, and there would still have to be appropriate multiuser version controls.
It would be desirable for the design tool to have some version control capability, storing versions of the models in some central location, and controlling concurrent updates. Then a user would check a model out of the central "library" when they wanted to make some revisions (otherwise they would be permitted read only access).

8.7 Direct Manipulation of Text and the ORM Diagrams

A Wiki operates as a single repository of content -- text, read-only diagrams (copy and paste), and attached files. Even the textual content of a Wiki is not edited according to the principles of direct manipulation. When the reader desires to modify a Wiki page, they must click on an "Edit" tab which opens up the page showing the embedded markup language. The Wiki markup language is used to create/revise the content of a Wiki page.

While a Wiki enables the immediate manipulation of text and the formatting of text, it does not allow for the immediate, direct manipulation of diagrams. The reader cannot read and edit a diagram in the same way they can edit text. This is understandable since the underlying application program, the data modeling tool, is not available within the Wiki. It is only possible to cut and paste the latest version of a diagram from the data modeling tool into a Wiki page. In this case, the diagram cannot be manipulated directly in the Wiki. It is necessary to revert back to the "compose, disseminate, revise, and integrate" process used in the traditional collaboration mechanisms. The reader must download an attached file (of a model diagram), open it up with the modeling tool (which is stored locally), edit and save the model, and upload it back to the Wiki server.

It is possible to use a screen sharing program where multiple users can be online concurrently, using the same instance of a data modeling tool, and taking turns to create or modify a single data model. One example would be Breeze Meeting from MacroMedia.
9. SUMMARY

This initial attempt to use a Wiki to facilitate the operation of virtual teams was challenging for me and somewhat frustrating for the students. We were both learning about the nature, operation, and effective use of a Wiki for ORM data modeling. The mixed results of this experience serve to temper the hype surrounding the use of Wikis.

Use of a Wiki effectively solves part of the problem for virtual teams -- handling the creation and manipulation of textual material. We still need effective mechanisms to handle other types of material, such as data model diagrams and accompanying information, that require specialized application software.

Wikis are of little help in handling non-textual materials which require special software tools to create and manipulate such artifacts as data model diagrams. It is back to the traditional methods of collaboration -- create, disseminate, review/revise, return, and integrate. Wiki participants are still on their own to control versions and concurrent updates.

A Wiki may be most suitable when there is no precise end product beyond the textual document itself. In ORM data modeling, the participants are trying to produce a model which is an accurate representation of some real world environment. They need to arrive at a single common understanding in the data model.

* The students certainly got, and appreciated getting, realistic exposure to the problems of integrating data models, particularly when there was much overlap in this project assignment.

* A Wiki works well as a repository for storing and sharing files.

* All teams built a page of elementary fact sentences, which is seen as a testament to the value of verbalization being the first step in conceptual data modeling, as in ORM.

* An initial face-to-face meeting is important, even if by video conferencing or using webcams.

* There was a problem knowing how to start and effectively use a Wiki for virtual team collaboration. Students and novice users of a Wiki need help using a Wiki to support a data modeling project with virtual teams. This could be partially overcome by providing usage notes and initial template suggestions for organizing the Wiki web.

* There was a lack of flexibility and functionality for editing Wiki webpages using the Wiki markup language (similar to HTML) which must be overcome before Wikis can become mainstream.

* Students did not use the Wiki for intra-team communication. Even if they tried, they still reverted back to the familiar traditional means, primarily email.

* Students need incentives to encourage timely participation.

* Use of the Wiki change notification capability may have triggered more timely responses, and increased the likelihood that the Wiki would be used for communication, particularly when hot links can be embedded in the email notice making it even easier.

* It remains difficult to share repeated revisions of ORM data models -- teams need version and concurrent update controls in the context of a Wiki.
REFERENCES

BOOCH, Grady, "Collaborative Development Environments," 2006 December 1,
http://www.booch.com/architecture


USEFUL WEBSITES

http://www.startwright.com/virtual.htm - "StartWright" is the leading reference site for the information needs of information technology project managers and support staff traveling to, starting up, and working on remote, dispersed or virtual projects.
http://www-users.cs.york.ac.uk/~kimble/teaching/mis/Distributed_Team_Work.html
http://www.collaboration-tools.com
APPENDIX

A. USING TRADITIONAL MECHANISMS TO COLLABORATE

An originating author writes something and sends it to the other members of the team, either as an attachment to an email, or posts it to some common location, often a web site. If not in a direct email, the recipients may receive notice of a new (or revised) posting, or have to periodically remember to check themselves. Each of recipient reviews it, and responds in any of the following ways.

1. A telephone call or email reply with comments and suggested revisions. In this case it is particularly difficult to reference the specific part/place in the source document to which the comment or revision applies. Alternatively, each reviewer can manually add annotations to a printed copy of the document and mail or fax it back. The originating author then has the task of integrating the feedback from multiple team members into a single revision. This can be difficult and time consuming, especially when the comments or suggested revisions conflict. This requires more communication and negotiation to resolve the differences.

2. Comments and revisions made directly to the electronic copy of the document. This is feasible if the system used to prepare the document has the ability to track changes by each reviewer, such as in Microsoft Word or Corel WordPerfect. The individual marked up copies of the original document are sent back to the author who uses the word processing software to examine and merge all the copies. The system maintains all the added comments and revisions in a single document. The original author can view all the comments and revisions together in one place. However, the author still has the task of integrating the feedback from the individual reviewers to produce a single revised document.

3. Another possibility is to have documents stored in a single location on the web, and a version control system to ensure that only one team member at a time has the document out for revision. This can be a time consuming process since the revisions can only be made serially. Any one reviewer can significantly delay the project.

4. Synchronous mechanisms (teleconferencing, or chat rooms) can be particularly useful when there is substantial disagreement that requires considerable negotiation to resolve differences. In such settings, other team members may be able to mediate the differing parties and bring them to some agreement.

Through this whole process it is still necessary to have someone as the primary coordinator of the work effort to oversee the process. The coordinator role need not be a single individual, rather responsibility for parts of the process or end products can be distributed among the team members.
B. CHARACTERISTICS OF A WIKI ENVIRONMENT

A Wiki can avoid the normal cycle of writing, disseminating, reviewing, revising, and integrating the feedback from multiple reviewers into a new version of the original document. A Wiki essentially provides the readers/reviewers the ability to actually edit what they are reading at that very moment, while it is fresh in their mind. This can be very powerful, and motivating. Knowing that they can revise what they are reading, they are more likely to read more critically. The reader can think beyond what they are reading, similar to brainstorming except that it is done asynchronously, at each viewer's convenience.

In its deployment, a Wiki provides an infrastructure where "knowledge" is created around a certain topic by many participants. The key to a Wiki is the incremental building of knowledge by a community of peers. From a mechanistic perspective, a Wiki is simply a software system "that enables users to edit the content of web pages," [Wikinomics, p.13]. But, according to Tapscott, it is much more: "a metaphor for a new era of collaboration and participation -- something dramatically different. The new promise of collaboration is that with peer production we will harness human skill, ingenuity, and intelligence more efficiently and effectively than anything we have witnessed previously," [Wikinomics, p.18]. Some call it the new "Age of Participation" where producers and consumers of knowledge become one: "prosumers!"

Wikis (indeed the whole open source movement) represents a second wave in the producer-consumer relationship. Not too many decades ago, professionals such as lawyers, teachers, and even doctoral students, would scribble out their ideas on paper or speak them into a dictation machine. Specialists in the "typing pool" would produce typed copy from these inputs, the products of individual creation. With the advent of computers, functioning as electronic typewriters, individual authors took on the task of creating the original typewritten documents using word processing software. Creating a document became more efficient and more accurate because the originator was creating, proofreading, editing, and revising the machinable copy. The typing pools disappeared. In fact the creative process was able to blossom. Furthermore, the revision process was rendered even more efficient. Once an ORM data model diagram is mechanized (computerized, digitized) it becomes very easy to make small revisions and print out a new copy, without having to recreate it entirely from scratch. This characterized the first wave -- the focus being on work produced by individuals.

The second wave extends the technology to groups of people. Appropriate mechanisms for collaborative creation of knowledge can further increase the efficiency and accuracy of the resulting output product, be it an encyclopedia, or a data model. Collaboration can generally produce a better result than individual action. That is why we have design review meetings, in which others can critically review the work of an individual or group. Furthermore, in some cases, mass collaboration can produce a better result than a small group.

One of the basic default rules for a true Wiki is that it is open to the world and that anyone can add or revise anything. Hence, the result can be the product of the collective wisdom of all the participants. From this perspective, a Wiki is ideal for mass collaboration. Every reader can become a contributor. This leads one to question the accuracy of the resulting product. However, Wikis have proven to be self-correcting and self-policing. With their name attached to any additions or revisions, participants are less likely to deliberately enter erroneous or undesirable material. Furthermore, errors and undesirable content are quickly corrected or removed. "According to an MIT study, an obscenity randomly inserted on Wikipedia is removed in an average of 1.7 minutes!" [Wikinomics, page 75]. In another study, Nature magazine compared 42 science entries in the Encyclopedia Britannica and in Wikipedia. They found a comparable number of errors. More importantly, the errors in Wikipedia
are corrected quickly, whereas the Britannica errors remain until the next revision. This is not surprising. Britannica is produced by a small set of expert writers who are not themselves experts in what they write; they gather, interpret, and summarize the expertise of others who have written about the subject. With (mass) collaboration in a Wiki, experts themselves can be involved in the original creation of the recorded knowledge.

In practice, however, a Wiki is often limited to a specific population. The default of 'open' changes to 'closed' (to others) once you give permission for named individuals to access the Wiki. Then the rest of the world is excluded. Whether open or closed, we cannot assume that all contributors are competent. Hence, a successful Wiki requires some oversight, review, and policing of additions and changes to the content.

A closed Wiki requires an invitation, qualification, registration (and banishment) process, which implies that there is some central control for the Wiki. Permission to participate can rest upon the satisfaction of various qualifications. Sometimes a central authority is needed to test and validate the contributions from participants to ensure that they are consistent with and will integrate with some core product. This is particularly true in the case of open source software. The operation of the Wiki is still decentralized in the creation and revision of content.

Sometimes, it is desirable to limit editing and revision to the individual or group responsible for the result, but still obtain and encourage input from other interested parties. In a Wiki, it is possible to give read-only permission to some participants, although that is not in keeping with the basic philosophy of a Wiki.

An important characteristic of a Wiki is the historical record of all revisions. This makes it possible to track changes, additions, and deletions, and to recover any content that was inadvertently or deliberately deleted or modified. In an educational setting, it also allows the instructor to assess the degree of timely participation by each student. Likewise, it allows students to view the relative contributions of other team members. In the team dynamics, it may encourage the more passive students to step up and the more aggressive students to hold back and allow others to take some responsibility for the work.

In the organization of Wiki webpages, it is wrong to think of them as arranged in some sort of hierarchy. This is a common misunderstanding, especially since we are so conditioned to seeing hierarchies, such as the directory of folders on a computer. Thinking hierarchically leads to confusion. The organization of a Wiki must be thought of as a web of pages, where any page can point to any number of other pages. Our class used the Wiki infrastructure provided by the University of Minnesota based on TWiki. We set up a Wiki web for the whole class and one for each team. A Wiki web starts out as a single webpage, and, as webpages are added, it grows to include many pages all linked together. Each webpage in a UM Wiki is called a Topic.

In summary, the distinguishing characteristic of a Wiki compared to other virtual team collaborative mechanisms is: the create/edit function is decentralized to a population of peers.
C.1 UMWiki

The Wiki system is based on the TWiki system. A Wiki software system is used to create and maintain many Wiki webs. A Wiki web is set up for a community of (peer) users, which may include all those in the University directory. A Wiki web consists of one or more linked webpages. Each Wiki webpage is called a Topic. The name of each Topic webpage is generally a sequence of two or more words, initial caps with no spaces, e.g. DataModelDiagram

Every Topic page displays the following information about the last revision:
Revision#, date, time (down to the second), and who made the change.
In that way, a reader can see if there have been any changes to the Topic since it was last viewed.

Every Topic page is displayed with the following external links (top, leftside, or bottom):

**ACTING ON THE CURRENT TOPIC PAGE BEING VIEWED:**
* "You are here" - showing the path of links by which you arrived at this topic page.
* Edit - displays the page with the Wiki markup language (opens in the same window).
* Attach - to attach a file to the Topic. The file is then stored at the Wiki web server.
* Printable - displays the topic page ready for printing without the external links shown.
* PDF - to generate a PDF copy of the topic page.
* Banklinks - showing the names of all topic pages which link to this topic page, for this Wiki web, or for all webs in the Wiki system.
* History - displays the complete list of revisions to this Topic page. Each revision shows the date, who made the revision, the changes, additions, and deletions. With the change history, it is possible to revert back to any prior version by undoing selected changes.
* Edit topic preference settings for this user
* Rename/Move topic, changing all references from all webs, or changing references in this web only
* Delete topic
* Set new topic parent (default is 'none' - all topic pages considered peers)
* Compare revisions; view previous revision

**ACTING ON THE WHOLE WIKI WEB:** (noting that a Wiki system will contain many webs)
* Create a new Topic, or just type in a valid topic name (words with initial caps and no spaces) and the system will automatically create a link to an as yet undefined topic page.
* Index of all Topic pages in this in this Wiki web.
* Search for a substring match, within this web, or across all public Wikis in the UMWiki system.
* Display the last change (date, time, who, first part of the change) for each Topic page in the web.
* Notification - set up a subscription to receive an email every time a change is made to named Topic pages. Choose what you want to have included in the notification, including the complete change.
* Statistics - a timeline showing the number of views, saves, uploads, and the top contributors for each Topic page.

* Preferences for this web which are applied to all Topic pages; can be overridden by individual user preferences.

When editing the contents of a Topic page, the system uses a Wiki markup language, similar to HTML, the Webpage markup language. The following formatting options are available:

- Font - bold, italics, underscore, strikethrough, size, sub/superscript
- Paragraphs - justification
- Headings - level
- Tables of columns (no predefined number of rows)
- Links - internal or external
- Embedded image
- Preview changes
- Undo - abandon changes but continue editing
- Cancel editing, making no changes
- Save and continue editing; Save and Exit.

Regarding concurrent update control, one of the technical support people sent the following: "There is a 10 minute edit lock on wiki topics. During that time period a user receives a warning but can choose to edit anyway. If a save comes in that is based on an old revision (ie there have been changes since the topic was opened for editing) the TWiki engine puts notations around conflicting parts of the save. It is then up to a person to come in and merge the two sets of edits. This is messy and we do encourage folks to wait out the lock period whenever possible to at least partially ensure they aren't going to run into a merge scenario. There is a "Check Point" button on the edit screen that will extend a lock beyond the initial 10 minutes."

Access the UMWiki and view the main web by going to: [http://wiki.umn.edu](http://wiki.umn.edu)

However you will not be able to see any other webs without permission (being registered in the University wide user directory, which can include permitted guests).

[https://wiki.umn.edu/twiki/bin/viewauth/IdscDataDesign](https://wiki.umn.edu/twiki/bin/viewauth/IdscDataDesign)
C. EDUCATIONAL vs. BUSINESS / ORGANIZATIONAL SETTING

One similarity between an educational setting and an organizational setting is that participants in a team effort will generally be using the same data modeling scheme, in this case ORM.

In a real world working environment, a database design effort would lead to a single data model. When that design effort is broken down for simplicity and manageability, the subparts must be integrated into a single whole. At the very least, models will relate to and, therefore, must be consistent with each other. The user domain subject matter experts work together (usually with an expert data modeling facilitator) to produce a data model [for further discussion, see Pelkki, 1995]. Every effort is made to resolve differences among the participants, since ultimately only one database will be built.

A large, complex data modeling project is often broken down into subareas, with subarea experts working on different teams. Smaller sub-projects are more manageable, but the sub-models must be integrated into a single comprehensive data model. This is why it is useful to have some common members across the teams.

In a real world organizational setting, individual contributions on a team effort may appear not to be explicitly evaluated. However, individual performance is evaluated -- it is just usually done implicitly and subjectively. It may not be reported as a "grade," but it will be reflected in opportunities for career advancement, promotions, assignment to projects, salary and bonuses.

In a classroom setting we must ultimately assign a grade for each student's performance. For class team work, the product of each team is evaluated with the same score given to each member of the team (unless other evidence warrants a grade differential for either significantly greater or lesser contribution to the total team effort. This is revealed by administering a Peer Evaluation in which each student comments on and rates the contributions of each team member, including themselves). This need to assign grades based on individual performance precluded assigning a single grade for the whole integrated data model, in which case all students in the class would receive the same grade for the assignment!

In an educational context, we need to receive distinct deliverables from each team. Hence we cannot complete the normal process of working toward a single, common design. Since we cannot take the final step of integrating the designs of individual subgroups, we emphasized that their product should be considered a preliminary or tentative design proposal which must yet be integrated with the other pieces and is subject to design review.

One alternative would be to assign the same data modeling problem to each team and have them work completely independently. This is the approach often taken in a classroom setting. Teams could still use a Wiki for intragroup activity. However, this would deny the students exposure to the dynamics and reality that integration is always an issue in practice. No system ever stands alone; there will always be a larger context in which it must interface and interoperate with other systems. Stove pipe application/information systems are the bane of most organizations today. The cry in the industry is for greater integration, even across corporate boundaries. That is the underlying motivation for EAI, EII, ERP, SOA, ESB, etc., indeed "Enterprise <anything>" is a cry for integration, for enlarging the scope, and thinking/designing in a broader context.