Developing an Intranet for a Small Unit

Capt. Dan Stormont
U.S. Air Force

The Theater Air Command and Control Simulation Facility (TACCSF) is a fairly small unit that works closely with contractors to develop simulation software and scenarios. The small size of the unit and the interaction with contractors prompted us to develop efficient ways to share data and documentation. This article discusses the approaches the TACCSF has taken to set up an Intranet using limited resources. It addresses the use of shared memory devices and the problems associated with that approach as well as the efforts undertaken to develop a browser-based (Intranet) paradigm. The lessons learned from these attempts are presented, as are our current efforts and future plans to develop the most cost-effective solution that will satisfy our needs.

HOW DO YOU SHARE information in your office? Pin pages on a bulletin board? Send E-mail out to everyone you think would be interested? Attach a routing slip to something and drop it in someone's in-box, never to be seen again? Write important meetings on a centrally located calendar? Although there is a need at times for formal routing of correspondence and face-to-face coordination of meetings, much of the information passed around in the typical office would best be disseminated by having it readily available on everyone's desktop computer. Recognizing these issues at the TACCSF motivated us to develop an efficient, usable Intranet for our organization. This article discusses the trials, tribulations, and lessons learned from these efforts.

About TACCSF

The TACCSF is an Air Combat Command-operated advanced distributed simulation facility. We are physically separated from our parent units: the 505th Command and Control Evaluation Group (CCEG) and the Air and Space Command and Control Agency.

The TACCSF team is uniquely composed of military members of Detachment 4, 505th CCEG, development contractors working for Lockheed Martin Corporation, and support contractors working for Scientific Research Corporation. The team works in the same building and must work closely on development projects for external customers interested in using our simulation facility. We have two PC networks—one operated by Detachment 4

and one operated by Lockheed Martin—that are used to share project information but must limit access to competition-sensitive or government-only information.

Another constraint is that Intranet site development and maintenance is only an additional duty for the people working on it, and funding for equipment and software is usually based on the amount of customer funds we receive in a year. We have a small unit with part-time Intranet employees operating with limited funds. Does this sound like your organization?

Why an Intranet?

Why did we want to develop an Intranet? We wanted to make as much information as possible available to people in the organization: information such as conference room schedules, important dates for the organization (Commander's Calls, etc.), and project information (schedules, funding charts, test plans, etc.). Most people are familiar and comfortable with using a Web browser, so an Intranet seemed to be a natural medium to share information.

Implementation Goals

What should an Intranet offer to be useful to a small unit like TACCSF? Most important, it needs to be easy to use. "Ease of use" is a phrase that tends to be ill-defined and overused; I define it as the ability to find the desired information (if it is available) with a minimal number of page changes. To make an Intranet easy to use requires a great deal of design and planning to logically organize and present the available data.

The Intranet also needs to limit access to sensitive information and be accessible to everyone who has a right to the information contained in it. This requires the use of password protection and possibly secure HyperText Markup Language (HTML) to protect the sensitive information.

Finally, the Intranet needs to be easy to maintain. The best Intranet site in the world is useless if it takes too long to update or demands that all changes go through a single choke point.

Using a Shared Memory Approach

When we first tried to share information at TACCSF, we used what I call a "shared memory" approach. This meant that people who wanted to share information, i.e., a document or a briefing, would place it in a shared memory area on the local area network (LAN) server and advertise its availability to people who might want to view the data.

There are a number of shortcomings with this approach. The biggest challenge can be to find the desired data. It does not take long for directories to proliferate in the shared area, which makes files extremely difficult to find unless you know the exact file path. Also, unless you have a file plan for your shared memory, you start to get several variations in organization and labeling of directories and files.

The shared memory approach is also a burden on the LAN administrator who must determine appropriate permissions on a directory-by-directory or often a file-by-file basis. The LAN administrator also has to juggle server

memory management since some users post more shared files than others.

First Attempt

Difficulties with the shared memory approach led to our first Intranet attempt—the TACCSF Intranet Software Engineering Environment (ISEE). The intent of ISEE was to share information concerning TACCSF projects. The site was organized by project and would contain information such as on-line documentation, project management data (schedules, financial reports), and briefings.

The biggest challenge with ISEE was the lack of good Web development tools at the time. Most of the HTML pages for ISEE were developed by hand using HTML editors that require the author to work directly with the HTML source code and refer to an HTML reference guide. This made page development tedious and time-consuming.

There were, however, some good ideas that came from the ISEE experience. We discovered that the best way to minimize page changes was to nest lists of data from the most general to the most specific. (This may seem obvious, but there are many examples on the Web where this principle is not followed.) The other good idea was linking related data in documents. For example, this allowed us to create a hyperlink requirements traceability and verification matrix.

One person created the ISEE in about 80 hours using a freeware HTML editor. It was hosted on our LAN server, which, at the time, was a Compaq Proliant 1000 running Novell NetWare 3.1.2.

Next Implementation

The next attempt at implementing an Intranet site started with the format (at that time) of the TACCSF Web page and extended it to provide project information, information of general interest, and on-line administrative information.

This attempt also used a "what you see is what you get" (WYSIWYG) graphical editor (Netscape Navigator Gold); however, the WYSIWYG fea-

tures of the editor were somewhat lacking, and a lot of custom page adjustments were required.

Another problem with this implementation was the lack of user involvement and requirements input, which meant that the system was never widely used.

The second attempt took about 40 hours to create and was also hosted on the Compaq server running Novell NetWare.

Current Efforts

We currently are experimenting with an Intranet implementation that will encompass the entire range of our daily operations and incorporate most of the elements that were present in previous attempts. This Intranet is intended to be more than a mere copy of the Web page or a source of project information.

One of the more important objectives of our current implementation is to increase usage of the Intranet. To accomplish this, it is essential to keep the data referenced in the Intranet pages current and to make the system easy to use and update. Fortunately, improvements in Web authoring tools and network operating system support for Intranets, along with increasing comfort levels with the Web browsing paradigm, make this objective easier to accomplish.

The organization of the current Intranet is being improved by designing the site map in advance (similar to doing a file plan for paper records) and assigning an "owner" for each of the sections of the Intranet site.

The latest Intranet sites were authored by one person at a time, with each iteration taking between eight and 16 hours. This illustrates how much authoring tools have improved since our first Intranet attempts. The latest iterations were created with AOLPress or Microsoft FrontPage and are hosted on a pair of Dell PowerEdge 2100s running Windows NT 4.0.

Future Plans

Once the current Intranet efforts are complete, we plan to start work on more interactive content, such as Java

database connectivity, interactive forms, and on-line scheduling of conference rooms and test-bed resources. Other possibilities for growth are collaborative design and reviews with off-site customers and password-protected Internet access to the Intranet.

Our preference is to host future versions of the Intranet on its own server that runs Windows NT Intranet Information Server.

Selecting Tools

The quality and affordability of Intranet tools has improved considerably since we first started experimenting with an Intranet site. The text-oriented Web page editors in common use then have been replaced by graphical tools; network operating systems now routinely bundle Intranet capabilities as part of the package, and highly capable browsers are freely available.

The current range and quality of Web development tools is impressive. Most of the popular office suites have the capability to save documents in HTML format, and the appearance of these documents has improved greatly. Of course, to develop a comprehensive site and not just an HTML page requires a Web development environment. There are a large number of Web development tools available at a reasonable price, either stand-alone (Microsoft's FrontPage) or as part of a browser package (Netscape's Communicator). One tool that is worth consideration is AOLPress, which is a free package available at http://www.aolpress.com. There is also a free server package that complements AOLPress. There are better tools available; however, until you have more experience with Intranet development and can better determine your development environment needs, AOLPress can help you get your Intranet project started.

Lessons Learned

The most important Intranet development lesson learned is that your Intranet must be easy to use. This requires taking time to lay out the site in advance to ensure that the format and organization is logical and aesthetically pleasing.

You must also ensure that the Intranet users feel they have a voice in its development and use. If people feel the Intranet is being forced on them and are not given an opportunity to express their opinions about its design, they are not likely to use it.

On a similar note, it is important to identify "owners" for the sections of the site. We approached this by assigning a portion of the site to each of the directorates in the organization, and the directors assigned a person to maintain their section of the Intranet.

Once the site is established, maintenance is much simpler; however, it is important to realize that developing an Intranet is a resource-intensive undertaking. You need to take the time to evaluate why you want to establish an Intranet and how your organization expects to benefit from it. Then you can determine if the expected benefits will outweigh the effort required to implement your Intranet site.

If you follow these recommendations, you should be able to set up a successful Intranet site for your organization, big or small. ◆

About the Author

Capt. Dan Stormont is the chief of project engineering at TACCSF at Kirtland Air Force Base, N.M. Prior to TACCSF, he was the computer engineer for the B-1B Conventional Mission Upgrade Program at Wright-Patterson Air Force Base, Ohio. He has an associate's degree in electronics engineering technology from the Community College of the Air Force, a bachelor's degree in computer engineering from the University of Arizona, and is pursuing a master's degree in computer engineering at the University of New Mexico. He is an active member of the Association for Computing Machinery and the Institute of Electrical and Electronics Engineers Computer Society.

Det 4, 505 CCEG/ENP
1655 First Street S.E.
Kirtland AFB, NM 87117-5617
Voice: 505-846-0539 DSN 246-0539
Fax: DSN 246-1486
E-mail: dstorm@taccsf.kirtland.af.mil

Coming Events

Software Quality Through Robust Testing
  Dates: May 21, 1998
  Location: Eatontown, N.J.
  Subject: Year 2000 fixes, reduce testing time and cost, improve coverage, and find defects early.
  Contact: Madhav Phadke, Voice: 732-577-2878; Fax: 732-577-2879; E-mail: Madhav_Phadke@compuserve.com

7th IEEE North Atlantic Test Workshop
  Dates: May 28-29, 1998
  Location: West Greenwich, R.I.
  Subject: Issues for the 21st Century: higher quality, more economical, and more efficient testing methodologies and designs.
  Sponsor: IEEE Computer Society, Test Technology Technical Committee, University of Rhode Island
  Contact: Jim Monzel, Voice: 802-769-6428; Fax: 802-769-7509, E-mail: jmonzel@vnet.ibm.com

Effective Methods of Defect Detection and Defect Prevention
  Dates: June 2-4, 1998
  Location: Seattle
  Subject: Software quality decomposed into defect detection and defect prevention.

Sponsor: Quality Assurance Institute
Contact: Voice: 407-363-1111; Fax: 407-363-1112; Internet: http://www.qaiusa.com

5th International Conference on Software Reuse
  Dates: June 2-5, 1998
  Location: Victoria, British Columbia
  Sponsor: IEEE Computer Society in cooperation with Association for Computing Machinery
  Contact: Jeffrey S. Poulin, program co-chairman; Voice: 607-751-6899; Fax: 607-751-6025; E-mail: Jeffrey.Poulin@lmco.com

Second Workshop on Software Architectures in Product Line Acquisitions
  Dates: June 8-10, 1998
  Location: Hawthorne Hotel, Salem, Mass.
  Subject: Applying software architecture technology to the acquisition of software-intensive product lines. Based on experiences, working groups will make recommendations to move to an architecture-based product line acquisition approach.
  Contact: Lt. Col. Gene Glasser, E-mail: glassere@issc.belvoir.army.mil

15th International Conference on Testing Computer Software
  Dates: June 8-12, 1998
  Location: Washington, D.C.
  Subject: "Testing Under Pressure," with emphasis on management strategies.
  Sponsor: U.S. Professional Development Institute
  Contact: Voice: 301-270-1033; Fax: 301-270-1040; E-mail: admin@uspdi.org; Internet: http://www.uspdi.org

4th Joint Avionics, Weapons, and Systems; Support, Software, and Simulation (JAWS S³) Symposium and Exhibition
  Dates: June 15-19, 1998
  Location: Riviera Hotel, Las Vegas, Nev.
  Subject: "Meeting the Technology Needs of the War Fighter in the Year 2000 and Beyond"
  Sponsors: Director, Test, Systems Engineering and Evaluation, Office of the Undersecretary of Defense; U.S. Navy, Avionics Department; U.S. Air Force, Embedded Computer Resources Support Improvement Program; JAWS S³ Working Group
  Contact: http://jawswg.org