Tips on Giving a Good Demo

Mary Fernández
SIGMOD 2005 Demonstrations Chair
mff@research.att.com

For the first time this year, a "Best Demonstrations" session was included in the SIGMOD program. The first two days of the program included 24 demonstrations, each of which was presented during two of six interactive demo sessions. During the first two days, panels of three judges visited each demo group, each of whom was allotted 15 minutes to present their system to the judges. The friendly competition made for very exciting and noisy demo sessions!

The judges selected three demonstrations [1, 2, 3] as the "best" based on five criteria that we share with you here. The selected demos were showcased in front of an audience in a special session on the final day of the conference.

Giving a good demonstration is like giving a good "elevator talk", but it is presented interactively. If you are unfamiliar with the classic elevator talk, here is the demonstration variant: Imagine that you get stuck in an elevator with your favorite high-tech mogul (e.g., Gates, Jobs, Ellison, Page & Brin), and you have fifteen minutes to sell your new database systems technology. You start to describe your idea, and the mogul says, "Don't tell me! Show me!". So you unsuspend your laptop (you are always prepared), and start to present your demonstration system.

Since giving a good demo is like telling a good story, we use a literary metaphor to describe each criteria. Do not underestimate the importance of telling a compelling story, because it helps people better remember your technical contribution.

User scenario : The characters

Introducing the users, or characters, in your story answers the key questions: *Who* is your target user and *why* are they important?

Your demonstration should answer these questions by describing a compelling user scenario or experience. If you are going to work very hard to build a system, it is likely that you have a particular user in mind. Database systems have numerous potential users: from end-users at their desktop, to database administrators, to analysts and statisticians, to animators and gamers—the list goes on and on. Know your user and put them at the center of your demo story.

Technical problem : The setting

Introducing the technical problem, or setting, answers the key question: *Why* does my system exist?

Your demonstration should answer this question by identifying the core technical problem. Ideally, your system should demonstrate the problem *interactively*. For example, if your system implements a query optimizer, you could demonstrate the problem by running queries without the optimization enabled, let the user see how slow it is, then possibly present a graph of the unoptimized query speed. As another example, if your system implements a graphical/high-level interface to a query language, you could show the queries that a user might have to write by hand in the absence of your interface.

If you cannot find a way to demonstrate the technical problem interactively, an effective alternative is to describe the problem graphically in a poster or in slides.

Technical solution : The plot

Presenting the technical solution, or plot, answers the key questions: *What* does my demonstration system do and *how* does it work?

Demonstrating your core technical solution is the centerpiece of your presentation. Ideally, your demo should include both the system's "dashboard" (i.e., its user interface) and a look "under its hood" (i.e., its internals). You should now return to your user scenario and show how it is handled using your system's dashboard.

Demonstrating the internal functionality of your system may be more difficult. Your technical solution, for example, may be an algorithm embedded deeply inside a query engine. Some effective techniques from the 2005 demo program included dynamic graphs that plotted relevant system variables such as CPU load and locations of distributed query plans; output of each phase in a query translation process; and displaying the intermediate representations of mediated data sources.

Making this internal functionality visible externally may require "extra" work that would not otherwise be necessary in a real system, as users typically only care about the system's dashboard, not what is going on under the hood. Visitors to your demo, however, may be more interested in the internals, so demonstrating the internals interactively will help visitors remember your technical contributions.

If your system solves multiple problems, an effective presentation technique is to introduce one problem at a time (using poster or slides), immediately followed by a demo that shows its solution. This technique draws the visitor into your demo quickly and avoids the common mistake of using all your time to describe the problems and not having sufficient time to show your system.

Integration : The sub-plots

Describing how your system is integrated with other technologies, or sub-plots, answers the key questions: *What* are the components of my demo system and *how* does my technical solution, interact with, support, or enhance related technologies?

The most interesting database systems are part of integrated systems. You should clearly identify the components of your demonstration system that you invented and engi-

neered entirely yourself, those that were integrated without any changes, and those that were modified to integrate with your technology. Some examples from the 2005 demo program included an open-source, 3-D game that was re-engineered to support interactive, streaming queries of character location and action, and a personal information system that used a desktop file metaphor to browse inferred relationships between heterogeneous data sources.

If your technology integrates easily or seamlessly with existing technologies, that is a sign of good design and engineering. You can illustrate this aspect, for example, by showing how much code had to be changed to integrate the components.

Impact : The resolution and insight

Describing the potential impact of your database systems research, or the resolution of your story, answers the key question: *How* and *when* might my technology have an impact?

If you are lucky, you might have one or two minutes to try to put your demo system in perspective. Making a guess about how your own work or similar systems might change the world may seem speculative and a bit egotistical, but informed speculation is very interesting to your visitors and helps them relate your technical contribution to what they already know.

Lastly, this is a good time to talk about the lessons that you learned. Often we start building a system with assumptions about what the "hard" problems are only to discover that the genuinely hard problems are something entirely different. Experiential knowledge of this kind is valuable and rarely shared in written form, but also of great interest to your visitors.

Pointers on Demo Form

Your demonstration will have a bigger impact if your visitors can see and hear you! Here are a few pointers on form to improve your presentation.

- *All* your visuals (posters, displays, slides etc.) should be legible to your visitors, who may have to stand 5-10 feet away from the display. Fonts should be at least 20 point.
- Bring or rent a monitor or video projector so that many people can view the demo at the same time. Hanging over a laptop is uncomfortable for everyone involved and limits the number of visitors at any one time.
- A poster is an advertisement for your demo, so it should be visually interesting. One or two large posters is more effective than hard copy print-outs of slides hung on the wall or an easel. If you prefer slides, present them on your laptop.
- *Point* directly to your visuals. Gesturing at a poster, slide, or display that is 10 feet away is ineffective and confusing. If you use a poster, stand next to it and point directly to the poster while describing the user scenario and technical problem.
- Demo rooms are *noisy*, so ask visitors to stand close to you so they can hear you. Speak directly to them, not at the poster or laptop.
- *Practice* your demonstration, more than once. Visitors will ask questions and things will go wrong, so do not expect to get through your entire demonstration every time. Don't leave the most important points to the end, because you may not get to them!

Putting it all together

You may find all these suggestions overwhelming and difficult to satisfy simultaneously, but consider them as opportunities to improve your communication skills. Learning research[1] has shown that different people learn best through different modalities: verbal (say it!), aural (hear it!), visual (see it!), and tactile (touch it!). Every person has a modality in which they best express themselves and in which they learn most effectively. Demonstrations are an excellent way to develop skills in multiple modalities, because they can incorporate many forms of communication. Giving demos will not only make you a better communicator, but will make your work accessible to more people.

Giving a demo is an immediate, intimate, and exciting way to share your technical work with others. We hope that you find these suggestions helpful in giving demonstrations of your database systems research.

References

[1] Y. Ahmad, B. Berg, U. Cetintemel, et al. Distributed Operation in the Borealis Stream Processing Engine. In *SIGMOD*, Demonstrations Program, June 2005.

[2] Y. Cai, X. Dong, A. Halevy, J. Liu, J. Madhavan. Personal Information Management with SEMEX In *SIGMOD*, Demonstrations Program, June 2005.

[3] Y. Li, H. Yang, H.V. Jagadish. NaLIX: An Interactive Natural Language Interface for Querying XML. In *SIGMOD*, Demonstrations Program, June 2005.

[1]Howard Gardner, Harvard University, http://www.howardgardner.com/