



Don Bishop ©1997 Artville, LLC

Strategic Development of the Usability Engineering Function

Let's imagine that you have decided to act as the usability champion in your product development organization. Whatever your professional discipline and your current job, if your goal is to introduce usability engineering into a product development organization that does not currently practice it, you must view yourself first and foremost as a *change agent*. Whether you are a usability practitioner, engineer, technical writer, project manager, business manager, or marketer, you must function primarily as an agent of organizational change. Failing to take this view of your role will most likely result (and indeed often has resulted) in a failure to introduce usability engineering into your organization in a lasting, integrated way. All the technical skills, good intentions, and sound logic in the world will not necessarily cause organizational change. Understanding what motivates organizations and causes them to change is key.

In my experience, there seem to be three distinct phases involved in establishing and evolving usability engineering to become a routine function in any software development organization:

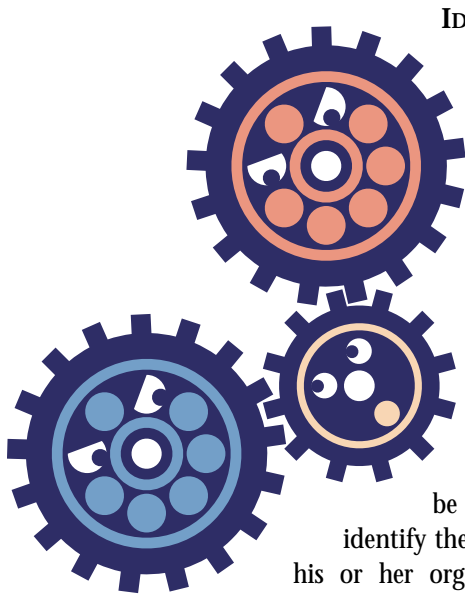
1. Promotion
2. Implementation
3. Institutionalization

In this column I offer thoughts and lessons from my own experience on how to successfully steer usability engineering through each of these phases.

Promotion

The first phase involves selling the organization on the very idea of usability engineering. Your focus here is influencing *people*. Your goal is to win the resources necessary to move into the next phase, implementing a usability engineering function. To do this, you need to

- ★ Identify and address organizational obstacles to change,
- ★ Exploit potential motivators, and
- ★ Apply success factors.



IDENTIFYING AND ADDRESSING ORGANIZATIONAL OBSTACLES TO CHANGE

As you attempt to facilitate organizational change, it is important to understand the forces in your organization that work to maintain the status quo. Anyone who aspires to be a change agent must identify the obstacles particular to his or her organization and address them directly and specifically. Failing to do so will usually result in failure to effect the desired results. You must understand the obstacles to change in order to overcome them.

Possible obstacles fall into several categories, including the following.

- ◆ **Prevalent myths, beliefs, and atti-**

tudes, for example, the attitude that usability is not really critical or the belief that usability engineering means nothing more than usability testing. You must continually educate your organization to address these obstacles.

- ◆ **Organizational incentives**, for example, an incentive system in which development teams are rewarded for staying on schedule and within budget but are neither rewarded nor punished for product quality, including usability. One way to address this obstacle is to make your target audience the group that stands to benefit the most from usability engineering (e.g., users), rather than the group that must incur the costs but has no accountability (e.g., development).

- ◆ **Organizational practices**, for example, limiting contact between developers and customers or users, a practice that makes user-centered design difficult. One way to address an organizational practice obstacle is to start with the easiest sell—usually usability testing. Organizations can usually be sold on usability testing, and it demonstrates the importance and utility of involving users. Then you can move on to persuading the organization to let you involve users in, for example, task analysis.

- ◆ **Organizational structures**, for example, organizational structures that make achieving integration and consistency across the functions of a software product difficult because of the division of labor across the whole project team. Short of convincing management to reorganize—which is the real solution—to address this obstacle you must be at least a good communicator and facilitator, acting as a liaison and communication pipeline among diverse organizational groups.

None of these obstacles is easy to address. However, failing to recognize and address them will doom your efforts to failure and yourself to a great deal of professional frustra-

tion. Again, to succeed as a usability champion, you must succeed as a change agent.

EXPLOITING POTENTIAL MOTIVATORS

Even though inertia and resistance to change are natural in most organizations, they often respond to strong incentives to make changes. When I look back on my years of experience as a consultant being brought into development organizations as their first attempt to use the discipline of usability engineering, I can ask myself why each organization chose that particular time to bring in someone with my expertise. From my experience, I can identify a variety of motivators, which have included the following.

✕ **A high-visibility disaster.** Most com-

success, including the following.

- ★ **Know your audiences.** Product development organizations have many potential stakeholders, for example, developers, project managers, high-level management, marketers, users, and user management. All have different motivators, and all can potentially benefit in different ways from usability. You need to understand the specific motivators and benefits for each and focus on them when communicating with any stakeholder.
- ★ **Clarify value added.** At least initially it can be useful to justify the costs of usability engineering tasks. The potential payoff of employing usability engineering techniques is not usually obvious to development engineers and managers, at least for the bottom-line. They will be reluctant to spend money, resources, and time for some vaguely defined benefit, especially given tight budgets and schedules. Sometimes you must make a good business case to clarify the bottom-line value of adding usability tasks to the overall project budget.
- ★ **Manage expectations.** One of the easiest mistakes you can make as a usability practitioner trying to gain respect and acceptance in a development organization is to feel you must have all the answers. Credibility is seriously damaged when unrealistic expectations are encouraged. For example, it is important to make the limitations—as well as the value—of prototype testing clear. Developers led to believe that prototype testing is the answer to user interface design will inevitably feel disappointed and disillusioned with the field. You must carefully point out, for example, that testing *identifies* problems but does not *solve* them, that testing will not necessarily predict sales, and other misconceptions.
- ★ **Test whenever possible.** At least initially, data are always better than expert opinion. This is a political, not an

objective, truth. We are talking about effecting change in a development organization, not about promoting objectively optimal methods in a non-political, completely accepting environment. In reality, simply soliciting an expert opinion can often be a much more cost-effective technique than formal usability testing. However, to get a development organization on the usability bandwagon, there is usually no substitute for formal, objective usability testing.

If you happen to be a usability practitioner, success factors also include the following.

- ◆ **Cast yourself as an ally, not an enemy.** It is easy to become viewed as the “usability police.” You need to cast yourself as an ally to development projects, as an invaluable resource that can help the whole team succeed, rather than as an enemy who will reveal their flaws and shortcomings to others. You must work toward creating a process through which you can have an impact, and this involves getting other engineers in the development organization invested in your skills, methods, techniques, knowledge, and design ideas. Getting “buy-in” is crucial.
- ◆ **Establish credibility.** You must move quickly to establish personal credibility, as well as credibility for the usability engineering discipline. Most likely the engineers and managers you work with will not be aware of your special skills and will imagine that you are simply someone with a different set of opinions who has been, for some unclear reason, assigned the job of designing a user interface. They may be resentful that this job has been taken away from them or that someone has been put in the position of critiquing their work and may be looking for validation of their skepticism. Therefore, it is critical that you choose tasks that clearly demonstrate the special skills you bring to the team.
- ◆ **Produce well-defined work products.**

BUSINESS COLUMN EDITOR

Susan Dray

Dray & Associates, Inc.

2007 Kenwood Parkway

Minneapolis, MN 55405,

USA

+1-612-377-1980

fax: +1-612-377-0363

dray@acm.org

You should certainly participate in design meetings, but rather than just throw in one more set of opinions, you should use these meetings to identify opportunities to define and conduct brief requirements analysis or evaluation tasks aimed at answering questions being raised and debated in these meetings. You should highlight these tasks, rather than your expert design advice, as your primary role. This will cast you as an expert with specialized skills who can be used as a resource in making design decisions, rather than as an adversary or competitor for turf with differing opinions.

❖ **Communicate effectively.** You need to be both articulate in your oral presentations and effective in your written communications to the developers you are trying to influence—who are, in a sense, your users. You need to write “developer-friendly” user interface design specification documents and to find even more effective ways to communicate design standards than specification documents. For example, an oral presentation, well illustrated, may more effectively communicate design standards than any product style guide will. A prototype embodying design standards will be better yet. Best of all might be design standards embedded directly in development tools.

❖ **Be an engineer, not an artist.** Software developers are engineers. They are trained to think and work in certain ways, and they relate to and work best with other engineers who think and work similarly. They view psychologists and artists as very different kinds of thinkers and workers and are often put off by the language and cultural differences of these professions. To be successful in an engineering environment, you must think and work like engineers. The more usability engineering techniques seem familiar to the engineer, the more likely they are to be respected and accepted.

Implementation

As a usability champion, succeeding in the first phase (promotion) means that you have won the opportunity to establish and develop a usability engineering function. Now your focus changes from influencing *people* to influencing *projects and products*. You are no longer just a salesman. Now you must be a manager as well (or you must find a good one).

The managerial issues that must be tackled during this phase will include the following.

⌘ **Finding the right person to manage the function.**

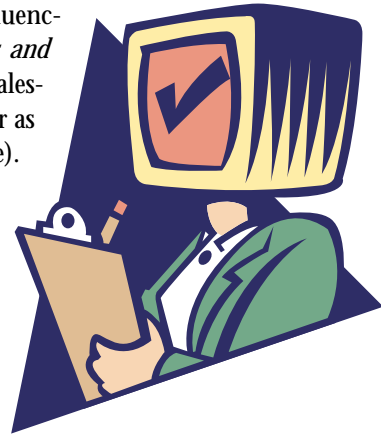
This may or may not be you, the initial usability champion who succeeded in promoting usability engineering in phase one. It should be someone who has not only an advanced level of usability engineering skills but also managerial and change management skills. If this is not you, have the good sense to step aside and support someone more qualified to help the new function succeed in this phase. Let's imagine, though, that you do indeed fit this description and that you do in fact become the manager of the new function.

⌘ **Staffing the function.** Ideally you will be able to recruit trained and experienced usability practitioners.

Alternatively, or in addition, you will need to recruit from within the organization, and provide training, coaching, and mentoring to develop your staff.

⌘ **Organizing the function.** You will need to decide whether to have a centralized or a decentralized function. This choice involves a number of trade-offs and must be tailored to each organization.

⌘ **Defining roles.** You will need to clarify the roles of the new function within the usual mix of roles on project teams within your development organization. You will need to match skill sets and roles. And you will need to provide



ongoing support to succeed in getting these new roles accepted.

❁ **Providing career paths.** You will need to think about providing a career path for the usability practitioners within their new function, so that the organization can attract and retain talented people.

❁ **Planning and budgeting.** You will need to conform to the planning and budgeting conventions of the development organization. You may need to act as a profit center or somehow justify the existence of the function. You should write an organizational plan each year in which you lay out the goals for the function and how you plan to deploy the resources of the function in pursuit of these goals. This will provide an agreed on general direction and a way to measure the function's performance for management at the end of the year.

In the implementation phase, it is important for the function manager to be strategic in the use of resources. A window of opportunity to establish a usability engineering function has been won, and early successes are critical for long-term survival and for moving to the third phase, institutionalization.

In my experience, what has most often worked best is introducing usability engineering techniques one at a time, on one project at a time. Usually usability testing is a good place to start. On a given project, applying usability engineering techniques at the usability testing stage is too late for them to work most effectively, but doing it almost always demonstrates the value of the techniques. It can help win the support and funding to do more and earlier usability engineering work (e.g., requirements analysis and design tasks) on later projects.

Similarly, the function might start out focusing efforts on the projects of one particular section of the overall development orga-

nization—the one that seems most receptive. Later, when successes have become visible and convincing, the function can expand the usability engineering resource and introduce usability engineering techniques to other parts of the organization.

Finally, find high-visibility, high-impact projects to which you can apply usability engineering techniques first; then, expand to cover all projects.

In order to win support for the next phase, it is imperative during this phase to make sure the function *publicizes* its successes and uses them to educate the rest of the organization on how to incorporate usability engineering into the development process. If no one else knows about the great work done on one project, its potential wider impact is lost.


Institutionalization

When you have succeeded in implementing a usability engineering function, you win the opportunity to develop the function to become a standard or institutionalized part of the way your whole development organization does business. Now your focus changes from influencing *individual projects and products* to influencing the development *process* in your organization. You no longer need just a salesman and a manager. Now you also need a methodologist.

Many usability engineering organizations go successfully from phase one to phase two and never get to phase three. I think this is because they fail to be strategic in phase two. Some simply stay in phase two, having an impact on certain projects but never becoming an institutionalized part of the overall development process. Some go along in phase two for awhile, only to be eliminated eventually in a downsizing. They are sacrificed in a downsizing precisely because they have not been strategic and have never achieved a phase three status in their development organization.

I believe the key to achieving phase three is integrating usability engineering with the development organization's standard methodology. In turn, the key to this achievement is not only working with the methodology keepers and getting written





into the formal methodology documentation, but also finding or creating the opportunities to put the new methodology into practice on projects and then publicizing these efforts and learning from them. Many methodology organizations are "ivory tower" organizations that never actually affect how projects are run. Conversely, influencing individual projects does not accomplish institutionalization. Only the combination works: one must get usability engineering into *both* the formal, documented methodology *and* into the actual, living, practiced methodology. This must be done in a highly visible and well publicized manner, so that the whole organization learns from the gradual introduction of usability engineering into the underlying development process.

I believe that it is inevitable that sooner or later, the software engineering discipline will embrace and incorporate usability engineering and it will become widely institutionalized in

development organizations, similarly to how software engineering methodologies in general have become institutionalized. But this will happen sooner rather than later if usability champions and practitioners understand the importance of moving from phase two to phase three and become more strategic in their role as organizational change agents.

Change is slow. Be patient. Be strategic!

[Parts of this column are excerpted from or based on Chapter 18 in Mayhew 1999 and Chapter 13 in Bias and Mayhew 1994.]

Bibliography

Bias, R.G. and Mayhew, D.J. *Cost-Justifying Usability*.

Academic Press, 1994, Boston, MA.

Mayhew, D.J. *The Usability Engineering Lifecycle*.

Morgan Kaufmann Publishers, 1999, San Francisco, CA.

Trenner, L. and Bawa, J. *The Politics of Usability*.

Springer-Verlag, 1998, London, UK.

Usability in Practice (M.E. Wiklund, ed.). Academic

Press, Inc., 1994, Boston, MA. *✓*