

Performance Support in Internet Time

The state of the practice on the eve of the new millennium.

Gary J. Dickelman

Abstract

December 27, 1999 marked a reunion of sorts for a group of professional colleagues who have paved the way for, defined, and are actively engaged in the performance support practice. Gloria Gery, Stan Malcolm, Janet Cichelli, Hal Christensen, Barry Raybould, and Marc Rosenberg spoke with me via teleconference about the state of performance support (PS). My goal was to uncover the latest trends - associated with the latest organizational performance pain - borne of the e-revolution, the monster enterprise systems (SAP, Siebel, Peoplesoft, etc), and the emerging awareness that *usability means business* (pun intended). A number of questions were posed on the topics of learning vs. performance support, knowledge management, performance support models, the role of the internet in performance support, enterprise application integration, and intrinsic performance support. This lively, insightful discussion will surely add to your personal knowledge base and enhance your performance support activities.

December 27, 1999 marked a reunion of sorts for a group of professional colleagues who have paved the way for, defined, and are actively engaged in the performance support practice. Gloria Gery, Stan Malcolm, Janet Cichelli, Hal Christensen, Barry Raybould, and Marc Rosenberg spoke with me via teleconference about the state of performance support (PS). My goal was to uncover the latest trends - associated with the latest organizational performance pain - borne of the e-revolution, the monster enterprise systems (SAP, Siebel, Peoplesoft, etc), and the emerging awareness that *usability means business* (pun intended). Application integration practices have emerged to address the engineering challenges around enterprise systems that need to talk to one another. A potential bonus in these activities is the burden it removes from users as a single interface is enabled for complex, diverse systems. I asked the panel to what extent the PS practice benefits from such integration activities.

I also wanted to uncover what hasn't changed - or worse - where things may have reverted. Novelist Ayn Rand once wrote, "Thousands of years ago the first man created fire. He was probably burned at the stake that he taught his brothers to light." (Rand, 1943) These pioneers, who years earlier established sound electronic learning principles and protocol - the forerunners of PS - and have evolved them to a new paradigm that addresses performance, have seen their work subverted. Web-based training *commodities* have emerged that address no learning or performance needs, are bereft of the richness we saw 15 to 20 years ago in the computer-based training (CBT) revolution, but assume the labels of performance support. Even *good* technology-based learning materials are being labeled performance support. Our experts acknowledge and provide insight to these market-motivated inclinations.

The e-revolution has also spawned a flurry of knowledge management (KM) activities, so I asked what role KM is playing in the PS practice. How has KM influenced our thinking about PS systems engineering, the formal process of developing performance-centered systems? Have PS systems moved beyond simply anticipating performance problems with hard-coded supports, toward truly *dynamic* and *adaptive* systems that continuously support performance as environmental factors change? And is there a valid model for PS that is dynamic and adaptive as it includes both engineered systems and human beings harmoniously converging to expert performance and wisdom?

I hope you enjoy this lively, wisdom-filled discussion. Most important, I hope that its lessons enrich your knowledge of performance support in a new context that is a world changing in internet time, generating performance problems at the same blistering rate, and begging for the very relief that these pioneers provide.

Learning/Training versus Performance Support, and the role of Learning Technologies in PS

Gary: There is a trend to create computer-based training - *web*-based training - instead of getting to the root of performance problems. Such conventional learning programs, unfortunately, are labeled performance support. Clearly the learning technologies play a role in PS, but are not inherently PS.

What is the proper role of the learning technologies in the PS practice? What skills and techniques does a PS practitioner need that transcend those of the learning technologist?

Hal: Any activity that begins and ends with putting things in people's heads rather than with directly improving their performance is not PS. This is true of both learning/training and knowledge management. PS calls for human, financial, technological, informational, and organizational resources to be directed toward performing rather than on learning how to perform. The point-of-performance interface, which delivers the precise amount and type of support at precisely the right time *is* the PS system.

Focusing on learning or knowledge transfer rather than performance results in *people who know what to do but never do it* (Pfeffer and Sutton, 1999). It also leads to training as a solution to problems for which it is inappropriate. Such ideas were laid down by people like Tom Gilbert and Robert Mager years ago, long before PS was articulated and understood.

Learning technologists must recognize that their jobs do not end with training. Learning must be turned into performance, shared with the entire organization, then cycled back into the next iteration of training. Learning technologists must take an interest in - if not responsibility for- the full cycle. They need to embed PS and knowledge sharing into learning activities so they become second nature to the learners/performers.

Stan: I've always thought it a shame that vendors of traditional interventions like CBT and documentation have found *currency* (pun intended) in the PS movement. Many have simply relabeled their stock offerings as performance support. This purely market-driven tactic has confused our client base. Performance support *components* are equated to the whole, ignoring the entire notion of integration and the focus on business performance that is the PS hallmark.

Practitioners become confused because many activities of PS development resemble those of CBT or on-line documentation development - *analysis, structuring, programming*, etc. The difference is the goal. CBT's goal is *learning*, with improved business performance an obscure *assumption*. With PS, *performance* is the goal. In a well-designed PS system, learning is likely, desirable, even inevitable - but it's not the point. **Performance** is the point.

Gloria: A reason for confusion in the marketplace is because many of the large players - with all due respect, like IBM Global Services - are still producing CBT. Enterprise Resource Planning system (ERP) vendors address the acknowledged pain by forming relationships with electronic training vendors or specialty vendors who can support products. Thus we have *so-called* PS software that interacts with a specific ERP, but doesn't address the greatest pain: supporting tasks that require integration with other applications. The training department is told by the information systems department (IS) to create CBT - which is largely ineffective and will otherwise be obsolete in six months. The ERP vendors sell to their own in-bred, co-opted group, making it a technician-to-technician sale. The default that I see everywhere is that the training department has no credibility. They are pushed aside by people who don't want them touching the code.

Stan: And the training departments that you're referring to are the *enlightened* ones! There are all the others who see web-based training as...

Gloria: ... the solution...

Stan: ... but without really doing anything different! The web offers these folks the opportunity to *appear* to be doing something *cutting edge* when in fact it's the same old ineffective training in a new package.

We need to be clear about the role of learning in PS. People insist that learning is noble in some abstract value-driven sense. The same people tend to fear that PS will dumb-down jobs to the order of, say, fast-food cashiers. Well, the fact is that much – most – of what training programs, CBT, and Web-based training have been requiring employees to learn is not enriching their lives. I'm not a better person for knowing the arcane codes associated with medical conditions in insurance claim handling. If a PS system can supply those codes, or, better still, hide them entirely, I'd rather not learn them, thank you. Instead, let me learn skills that apply, for example, to making better claim decisions.

Marc: Don't get me started. There are tons of cash and very aggressive vendors in this business who are selling substandard courses and *expectations* about what they can do. Calling it performance support gets the attention of corporate buyers.

There is a great role for learning technologies within PS, if done right. Bridging PS with learning technologies helps PS practitioners get to the table. The skills necessary to create PS include information design, human factors, interactivity, measurement, and performance analysis. It's a mistake to focus on programming -- HTML, etc. The number one skill is perhaps writing. We get excellent programmers who create great looking sites that do interesting things, but the content is just so much babble. Give me people who can write to users' needs using just the right words.

Janet: A big part of that problem is that training organizations have not forged relationships with and have not earned the respect of IT. And so they move ahead on their own, creating solutions that can be deployed as training programs, with which they are most familiar. Training departments are making quite a commitment in staffing and budgets for developing PS tools, performance-centered applications, and what many organizations are calling a *tool kit* - performance tools that are integrated with CBT, introduced through the learning context of CBT, and then uncoupled from training and made available to performers on the job.

Gloria: And it's just like a typical tool kit: a jumbled mess.

Janet: That depends on its accessibility, whether it's organized around a task structure, and the context in which it's used. There is a strong acknowledgement of PS in many training organizations. Some develop CBT exclusively, but call it performance support. Others create very good PS tools and integrated PS systems, in some cases rolling them out within the framework of CBT.

Gloria: That goes back to control. It's not necessarily ill-formed, but simply inadequate. We see many little *junk* utilities - and I call them "junk" because they are fine for what they do but every one looks different, every one has a different architecture, and every one can do very narrowly focused things - never getting integrated over time. It's the enterprise systems into which they have to be integrated. You can ease a sharp pain that someone has expressed, but it's like giving financial calculators to investment specialists. If selecting an appropriate investment is not supported, then what good is the calculator?

Stan: Another way to look at these is: *Do they have the potential to lead to something better?* Do they have potential to build credibility and understanding of what PS is capable of doing in an organization? - and, ultimately, lead to a higher level of cooperation between those building PS tools and those who own the enterprise systems? Rather than seeing it as a nice tool that, unfortunately, is not integrated, are there some longer-term consequences to having built the PS tool?

Gloria: Unless those who build the tools have the motivation to learn more about the technologies and where integration is necessary, they will never develop the necessary credibility to influence IS.

One reason why there's so much relief with the web is because it simplifies implementing CBT. So we see a trend of reverting back to sequences like: *text...text...question*; *text...text...picture...question* - in the same way that it was done in mainframe/3270 days, when it was not possible to have a dynamic display.

Stan: That's exacerbated by curriculum vendors of web-based training who are trying to turn their wares into a commodity.

Gary: In the process of making commodities of web-based training, the development tools have severely limited instructional strategies.

Gloria: People are trading effectiveness for ubiquity, period.

Gary: Ubiquity is a precursor to market success these days. That's what investors are looking for. Ubiquity - even in the absence of revenue - can be considered good if market capitalization is sufficiently high.

Barry: PS is a continuum, with training a part. We should build as much support as possible intrinsically, but also provide some training and other supports. Many companies today doing web business, for example, are recognizing that sales transactions cannot always be completed entirely electronically. They are letting consumers speak to a customer service representative at a key point in the transaction. They have recognized the need for multiple supports: (1) an intrinsic performance-centered web application, and (2) a human support structure to complete the transaction. Rather than try to engineer only intrinsic performance support, I think we should be engineering a continuum of support, actively designing each component along that continuum, rather than viewing the situation as either/or. Of course you should provide as much support as possible intrinsically and only do externally or extrinsically (via training or telephone hotline, for example) what you cannot do within the system.

Gary: Unfortunately, classifying PS activities as intrinsic, extrinsic, or external reflects the *tactics* around the degree to which you can influence IT.

Barry: In such cases PS is not designed actively but instead opportunistically.

Hal: It's very important to blend activities that involve other human beings and establish protocol to help the organization adapt and grow knowledge as it evolves.

Barry: Terminology may change, but the interesting thing is that the approaches and methodologies that the PS community has been developing over the past ten years are exactly what is needed to ensure knowledge management systems and web-based e-commerce systems will be successful, and result in improved human and business performance. That's the exciting thing; the terminology is not important. What is important is that there is now a community of professionals who have a structured set of design principles and methodologies that together form a repeatable process for designing systems that achieve business results.

Gloria: To be sure, we're not suggesting that the term PS will become *passee*. But, like all worthwhile innovation, it will disappear into the fabric of best-practice. But we're still a long way from seeing that.

Stan: The concern is that over time, as the term fades, that the lessons we've learned over the past ten years or so are not passed on. It would be unfortunate if people had to re-learn those painful lessons associated with the term.

Hal: That's happened with CBT. The lessons of ten to fifteen years ago have been forgotten.

Barry: Maybe we need a term like *performance-centered e-business* to keep up the momentum.

Janet: The trend I've been seeing over the past two years is a separation in the camps of extrinsic PS, where it remains in the training camp with integration into learning. On the other hand, performance-centered design (PCD) is residing in the IT camp and getting some support in the collaboration between training, HR, and the line of business. But the real interest in performance-centered design within the IT community is mainstreaming PCD activities and techniques in their software development lifecycles.

Gary: I see that even in the graduate programs. As an advisor to a team of instructional designer graduate students who are creating a PS system, I see performance-centered design as part their development lifecycle. The program also requires a certain degree of technological savvy. The university is not producing instructional designers who are capable of creating only extrinsic or external PS. Performance-centered system development capabilities is the goal.

Janet: I see that as a trend. I don't necessarily see that a division is desirable. There's a need for the role and a need for the collaboration. We need people who have a human performance background if we are to produce interfaces that address the performance issues - the understanding of the work and of the workers, for example, and its translation to effective interfaces.

Knowledge Management (KM)

Gary: Over the past several years organizations have been striving to make documents and other knowledge assets universally available. Those very assets form the basis of performance-enabling content if delivered in the right context. Further, Internet technology appears to enable ubiquity of knowledge assets like no other time in history. It seems, therefore, that KM is a significant player in PS activities.

What is the role of knowledge management (KM) in performance support systems?

Barry: Improving business performance, which is a goal of PS, requires knowledge capture, maintenance, and management, plus a performance-centered interface to a knowledge base. Building performance-centered systems requires systems thinking, which focuses PS on knowledge acquisition. Systems thinking forces you to uncover key goals, barriers to achieving those goals and which knowledge is relevant so you don't end up trying to capture everything. It also forces you to discover how a worker gets feedback on job performance, and to look for ways to build that feedback into the system

A PS system is the infrastructure for knowledge management. Performance-centered design (PCD) transforms knowledge into performance by creating an interface to the knowledge base. Unfortunately there are many KM efforts that lack a performance-centered interface, thus there is no means of turning knowledge into performance. Examples include the many *document* management efforts labeled KM. Making documents available electronically does not improve performance. PS system design emphasizes how to display knowledge but often overlooks how to capture and maintain it.

Gary: So what are some of the elements of knowledge capture and interface design in PS engineering methodology?

Barry: Capturing and structuring knowledge are still difficult to automate and remain primarily manual processes, with a few exceptions (such as neural networks - still in their infancy regarding their learning ability compared with the human brain). Since it is a human process, there must be incentives for maintaining the key knowledge elements that you've identified for your system.

Marc: KM requires communities of interest/practice, and must be perceived by knowledge seekers as rewarding. The metaphor for KM is not the classroom but the *library*, because knowledge structure, relevancy, and authenticity of content are what matter.

Stan: It's true that KM seems to be the cure du jour, but knowledge is a critical component of PS nonetheless. After all, if you don't have access to knowledge when you need it, you're surely (editor's substitution) out of luck (SOL).

There's so much smoke around *capital K*, *capital M*, that it obscures the real issues we face in performance centered design. We need to be asking, "What knowledge?" and "When?" to make it instantly accessible in the task context. We need to be addressing the dynamic, mutable nature of knowledge so that our PS systems adapt to changing business conditions and performers' process innovations more rapidly than with typical software releases. We need to be *anticipating* knowledge, not just capturing it. We need to be building environmental *sensors* and *predictors* into our systems.

Gary: *What are the barriers to KM?*

Barry: The largest barriers to KM are cultural. It takes considerable organizational commitment to plan and budget for capturing knowledge and structuring it in a form that is useful to someone other than the person who created it. Also, people have a tendency to hoard knowledge to be considered the expert.

Gloria: I don't see hoarding these days because of the rate of change of processes within organizations. Time is the real barrier. Mechanisms for capturing things in a natural way - while you're doing them - are absent, versus waiting until some end-point in the process to reconstitute the knowledge, when it is clearly burdensome and there's no time to do it.

Barry: It depends on the performers. I agree with you for many routine jobs, but for professional jobs such as consulting and management, - where promotion is based on your level of knowledge - the situation is different.

Stan: There must be a *value proposition* for sharing knowledge if capturing knowledge is to take place continuously. It's not so much a matter of hoarding as it is, "What's in it for me? Am I likely to get anything back if I share?" - like an investment, or insurance.

Gloria: I don't see knowledge hoarding in medicine, consulting, engineering, or sales. People in sales protect their leads, but that's different from the common problem of how to capture the market or, in medicine, how to treat a new disease. In these cases if knowledge is shared, jobs become easier thus sharing becomes a cultural norm. Organizations have systems in place where you specify your project and they retrieve, filter, and focus the gamut of external and internal resources (like proposals, project plans, PowerPoint presentations). The business driver is reducing cycle time by reusing everything possible. They don't want to generate another new proposal or create a new course; they want to make use of existing resources, knowledge, etc., to meet the customer need in the shortest, most efficient time.

I've seen PS tools that structure thinking through the analytical process, then prompt you to add metadata to the resulting project files so that they are available and easily retrieved worldwide for similar projects. These are not only associated with the large consulting firms, but also with organizations like the IRS for complaint resolution and general taxpayer problems. Because of market conditions, these organizations are looking to find the solution fast; control is no longer the issue.

Hal: We still see a lot of hoarding, particularly related to sales. The model we're using gathers and presents a blended solution of training/learning, performance support, and knowledge management, creating a *knowledge community*. The idea is to plant the seeds of knowledge sharing and management into training and PS through team activities, making them a natural part of doing work. Knowledge sharing fails when we announce that we're going to set up a

- knowledge community: "Here's it is, folks, do it and you'll enjoy it..." without participants first experiencing the benefits.
- Janet: Other cultural issues have emerged, such as the need to convey trust, validity of information, validity of its source, and the like. When we infuse KM thinking into PS and vice-versa, we expand the notion of PS as human and computer into a larger enterprise. This is particularly true when you add all of the *other* human beings to the PS system that an individual performer interacts with to achieve a goal - like expert peers. In the context of PS, raising the level of conversation within an organization is the essence of KM thinking.
- Gloria: One of the goals for any PS system - for example, when you think of growing it dynamically over time - is to capture new knowledge. When knowledge growth is a goal, some of it may be things like automatically generating call documentation (customer service) or capturing new symptoms (equipment maintenance). It can be as simple as creating *process application kits* where new symptoms are added by each practitioner who doesn't find his or her symptoms in an existing list. Capture becomes organic, but the new problem to solve is *validating* newly acquired knowledge. A goal of PS is therefore growing knowledge and institutionalizing it through validation - not just including existing knowledge at times anticipated by the system developers.
- Barry: As you add more and more knowledge it often ends up in distinct knowledge structures that overlap. To make knowledge useful there must be a means to integrate it into a single knowledge base and PS structure, so that the performer doesn't have to search multiple places to get the complete picture.
- Gary: Unfortunately, many organizations capture knowledge through something like Lotus Notes. It falls short of being able to grow knowledge that addresses performance. At best, the strongest features are underutilized or improperly implemented. The result is what Barry spoke of: A large collection of databases and a performer population bereft of tools to filter and focus the information into knowledge and performance in context with the work process.
- Barry: This is another example of the problem I talked about earlier. Lotus Notes is often used as a field and record database structure for creating a knowledge base. Unless you build a performance-centered interface to that knowledge base, it is not very useful. Useful knowledge requires algorithms that link its chunks to the appropriate part of a performance-centered user interface so that the performer obtains complete knowledge at the moment of need and in the right context. This is a much more complicated design task than just presenting different views of records and fields in a database structure.
- Stan: Thought this conversation on KM, Barry outlined four key elements: (1) A process by which you solicit knowledge; (2) another by which you receive it; (3) you must then assess it; and finally (4) somehow disseminate it. We've done these things in the past in PS systems, but through a release strategy, where a version appears that addresses current understanding, then weeks, months, or years later another version appears that addresses a changed understanding. We need to make the PS system dynamic in *real time* and not just in discrete releases. We ought to be moving away from the iterative approach - of releases separated by relatively long periods of time - to something dynamic that evolves continuously to support performance, without the need for significant developer intervention.

Internet

- Gary: *Are the Internet and web-based applications advancing PS in any way? Do they, for example, reduce or eliminate the need for training?*
- Marc: Web-based applications make some tasks easier to perform and people are becoming very web-savvy. Check out companies like ARIBA (<http://www.ariba.com>) and COMMERCE ONE

(<http://www.commerceone.com>) to see how you can conduct business with no training and only a little learning on task. The need for training is reduced, but generally not eliminated.

- Janet: The Internet is simply a different medium and doesn't necessarily foster performance-centered interfaces. I've watched interfaces evolve from character-based mainframe systems to GUI / client-server applications with no significant improvements in usability. They keep exactly the same screen layout, process flow, and functionality - then move into a browser environment *still* looking like their mainframe predecessors. Today the performance-centered web-based systems are those driven by electronic commerce, where if people can't easily buy, revenues plunge, market share is lost, and businesses fail.
- Barry: The most exciting thing about the Internet is that PS and PCD used to be luxuries, but now they are necessities. Since business transactions are moving directly to consumers and bypassing intermediaries such as customer service representatives, a performance-centered interface is critical to completing transactions. Whereas a customer service representative can compensate for poor systems design by getting answers from someone sitting in the next cubicle, the consumer doesn't have that option.
- Stan: I concur with your comments and Janet's, and bring to bear my experience with an ad agency that is explicitly using PCD to enable e-commerce. These people are interested in the corporation's entire image, and their clients are marketing people concerned with business performance. In hindsight it's a very natural place to expect PCD to emerge.
- Gloria: We're seeing performance support viewpoints pop up everywhere. As advertising has turned to the Internet we've seen PS even in IT departments; for example, in Siebel. Customer relationship management has taken off because it fills the void created by the large ERP systems. Siebel is also going wild because it focuses on the essence of sales automation - which is *customer relationship management* - instead of time and territory management. So the goals are becoming more explicit because of data availability and changes in architecture. I personally think that PS is going to be what the information center was in the days of data processing, before end-user computing. It will be a temporary advocacy until the infrastructure to enable and support it is in place. We will be maintaining the PS advocacy for a few more years before it disappears...
- Stan: ...into business-as-usual, without a need for a special term.
- Gloria: I'm buying a new car today and was able to do all of my preparatory work on the Internet. I will walk into the dealership knowing configurations and comparisons - by price, by feature, by engine, by whatever is important to me. I call it *unbelievable* when I can create generative charts - including things like the final assembly location, get all of the third-party evaluations, incident reports, and the like. I can find what I need to achieve the goal and accomplish the task. *Unbelievable* because you can do so for just about anything. It's an information democratization of sorts. The only reason we had intermediaries in the past is because they had access to the data and could perform transactions. Now the data and the transactions are universally available, whether it's purchasing a car or buying stock or doing research. The value proposition of dealing with an intermediary has diminished substantially by information and transaction access; the web does that.
- This goes back to Janet's point about the necessity for performance support on the Internet. We'll leave the confusing web site as fast as we'll abandon a boring person at a cocktail party. It's the compelling context where you really have no choice.
- Hal: The intermediaries Gloria just referred to are *web portals*. They are the interfaces to complex processes and job functions. They guide a person to the right information and tools to achieve a goal. Acquiring knowledge via a PS portal is not simply a matter of using search engines. Rather, relevant sites, tools, and resources are tied together based on the job context and the goal. For example, they enable transitions from selling products to selling customer relationship

management services. Portals exist to drive sales people through the process, helping them to acquire the necessary knowledge and skills within the context of their jobs, and thus accomplish new goal effectively.

Dynamics and Human Activities in PS

Gary: In his PI article Performance Support in Perspective (Banerji, 1999), Dr. Ashok Banerji described a PS system as a human activity system, where the human being strives to achieve a goal or complete a task. The system includes dynamic elements to ensure that the permeability of barriers to achieving the goal increases until the goal is met and until the human being is able to accomplish tasks in the most efficient way. So, according to Banerji, a PS system includes the computer and the human being, and it is *adaptive* with respect to completing business tasks and achieving business goals.

Is Banerji's description of PS consistent with how you approach performance issues in your respective practices? If so, please elaborate; if not, then how is your approach different?

Gloria: The model that incorporates the human is the right one, but I do not see dynamic human-computer interaction completely implemented. An early stage of dynamic PS dominates, where existing knowledge is organized and linked to context. Almost 23 years ago we were creating field help and screen help at the same level of sophistication that we see today, except now there may be some form of task or process help. PS developers are still trying to *hard wire* support to the context that is procedural or conceptual, not state-sensitive or behavior-sensitive.

Gary: An exception, perhaps, is the way menu items appear or disappear in Windows 2000, based on usage. Ironically, such dynamics occur in GUI menus, which were designed for predictability. Designers have therefore introduced conflicting goals, and at the expense of the user.

Marc: If PS is successful, people perform better but also learn. Consequently their reliance on PS may decrease, so the PS system must adapt if it to continue helping performers with increasingly challenging tasks. It would be nice, for example, if MS Word's *Mr. Paper Clip* - who I no longer need when crafting a letter - would assist me in the difficult area of styles; I wish he would adapt to my needs.

Gloria: The new agent technology is fundamental to dynamic PS. As the logic, the tools, and the ability to program them improve, dynamic PS will progress. Today I see even the best software engineers on PS development teams being barely able to hard-wire support, never mind introduce dynamic support via agents and based on sets of conditions. So Banerji's is a good model, but very far from what most organizations are able to even begin to achieve.

Stan: We see systems that are, at best, responsive based on the extent to which developers can *anticipate* user needs. Systems need to be adaptive based on external environmental sensors or predictors - such as financial management systems adapting to what's going on in the stock market. These are things *external* to built-in system rules and external to the user of the system, but are out in the world affecting business performance. It is ultimately business performance that's being supported, which we hope to achieve by making human performance faster, more accurate, and more adaptable to changing circumstances.

Janet: The ability of PS systems to be adaptive today is limited by *hard-wired* content and supports. Users can access tools or content using different views, levels of detail, and sequences, but still within the pre-defined framework. For example, *guide me* versus *remind me* reflects the novice versus expert view. While these are rudimentary forms of PS, they acknowledge performer diversity with limited adaptability.

Gary: In certain enterprise resource planning (ERP) systems, vendors have eliminated the need for users to navigate dozens of screens and sort through hundreds of data fields to input three or four data

items by changing the interface dynamically based on role. The user sees a single panel containing only those data relevant to the job.

Do you see changing the view of a system according to job or role as supporting dynamics?

Gloria: That's just a filtered view of the system based on task or role...

Stan: ...again reflecting what developers can anticipate and not true dynamics.

Hal: The adaptive attribute of PS I'm dealing with is less for computer systems but more around the jobs of knowledge workers (e.g., sales people, branch managers), where accessing a system is only a small part of the work activity. The opportunity for PS to be adaptive presents itself as a means to drive and sustain competent performance once someone is already trained, skilled at some level, and engaged in a process. As the worker improves performance by applying knowledge in new ways, the challenge is to capture that event and feed it back into the work context - which is the PS system. The goal is to grow both the knowledge base *and* organizational competency - which is how I interpret Banerji's vision. So far PS has not been adaptive such that it expands organizational wisdom.

Gloria: The systems I see are fixed, do not match tasks, do not match mental models, and do not filter information significantly. They are becoming more dynamic or adaptive as rule based-, behavior based- or data based systems. To be *performance based* they must be openly modifiable and adaptive to ensure that performance goals are met in a changing business context.

The development activities I'm engaged in mostly *narrow* human behavior rather than keep it open. Why? Because we're not dealing with highly generative, creative work. We're creating a purchase order - a task that was once accomplished by *everyone* but is now too complex to be done by *anyone*. It is because the systems, in their goals to manage data and processes have introduced so much complexity and have failed to consider the human being in the process.

Janet: 'Adaptive' is about being able to migrate from what exists today to some necessary, perhaps unanticipated future state. It can be in terms of a performer's progression from being less competent to more competent. It can reflect changes in work style or in the work environment. It can reflect changes in how job tasks are accomplished, how business processes evolve, and generally how the current state should change to a more efficient future state. It may be in terms of the knowledge that is currently available to support the work versus the knowledge that may become available in the future. As Gloria said about the changes in human behavior, the performer is doing something today that may need to change; performance support can facilitate that change.

Janet: We've heard *day-one-performance* as a goal of performance support, but what about *day-two efficiency*? A colleague tells us of a PS system that was a resounding success in terms of day-one performance but a failure in terms of day-two efficiency. The lesson is that PS must adapt to business shifts, individual diversity, and other dynamic factors that affect performance. In this sense, I agree with Banerji's description.

Enterprise Application Integration (EAI)

Gary: Integrating diverse technologies often fosters PS and KM. In the past there were silos and gatekeepers around the billing system, the payroll systems, the general ledger system, etc. They were separate not only by vendor and in functionality, but also in their data structures and reporting software, and most likely even with respect to the people that maintained them. Now there are companies devoted to getting one system to talk to another in the emerging field called *Enterprise Application Integration* (EAI). EAI companies create custom bridges (adapters) that connect systems (like Siebel and SAP). The user is unburdened with understanding different data models or distinguishing functionality, and can get the otherwise cross-system task done very simply through a browser interface.

How do you see EAI activities affecting the PS practice?

- Janet: The opportunities that EAI present for PS practitioners include designing interfaces that help the performer maintain context, focus on tasks, and focus on goals without worrying about data and data integration.
- Gloria: Some time back, usability focused on an individual using one or two systems to accomplish tasks. Recent trends in *process* integration reveal that the number of systems that an individual has to touch has increased substantially. Whether it's a call center, a manufacturing process, or a sales activity, the data is everywhere and the task activities that comprise a process require people to touch *between four and twelve systems*. Every one looks different, every one has different labels, and the labels have different meanings. The performance problem - the one that the business people nod affirmatively about - is: *In order to do the work the performer has to develop a mental model of the entire corporate system architecture, where the data resides, what screen it resides on, how to navigate to that screen, and how to integrate information that is in multiple places*. Not one system was designed to support the view that the performer needs. Cognitive processing, navigating, and synthesis necessary to bring information together to answer the critical business question and proceed with the task are bringing organizations to their knees. At one client site it requires *five* systems to generate a simple purchase order. It's not just the ERP that's a problem; they have to go, for example, to a different system for the part number. And these underlying systems are redesigned, requiring performers to relearn the tasks. There's nobody in the IS department who knows what the \$12,000 per year clerk has to know. So the motivation is to create a new glue through a user interface designed around the work that masks the interactions with the multiple systems.
- Gary: There are the business processes, system functionality, and data models. Organizations are asking us to help reduce user pain around the latter two, where they are forced to mentally map system functionality and data to the business process. If organizations could take those two burdens away and allow users to focus only on the business tasks, work would get done expediently and accurately without training.
- Gloria: The irony is that the pain creates incentives to *bypass* the systems. One client was telling me that workers buy parts by credit card through external means, then submit expense reimbursements - all to bypass the complexities of and avoid having to learn the internal systems. Ultimately, bypassing the systems is faster, cheaper, and easier than using them because of the complexity and related pain. And while the technology issues are certainly important, the political issues around system departments and silos remain equally important.
- Gary: I see more and more that the organization that takes ownership of PS - whether it be training, documentation, or human factors engineering - includes more IT involvement. The teams that are tasked with supporting performance are better integrated in this respect than I have seen in past years.
- Gloria: I see more integration, too, but instances are few and the process is still very painful. I don't disagree with you, but it's just not enough.
- Gary: But the complexion is changing. I see entire IT groups tasked with solving a performance support problem - even if it's around technology infrastructure. The performance support issue has gotten someone's attention to the extent that the integration problem, for example, has visibility and support from the right level. Perhaps they don't see the pain the same way that a performance support practitioner does, but they see it nonetheless, and the result is we're getting closer to a common goal.

Intrinsic PS and the Future

Gary: *What, in your opinion, are the things necessary to achieving good intrinsic performance support (e.g., relationships between IT and PS professionals; tools; top-down support; cross-discipline cooperation; cost-benefit; etc.)? What does the future hold?*

Stan: Well I certainly feel that the more intrinsic the performance supports are, the better, but it's rare that we achieve anything more than extrinsic levels. That goes for most of the entrants in the EPSS / Performance-Centered Design competition¹, which ought to represent a forum for showcasing innovation.

I think it's a pipe-dream to expect the IT- and technology-based learning stovepipes to suddenly figure out what each offers to the other and to begin working closely to achieve good intrinsic PS. It happens, but it's rare and will remain so. Intrinsic performance support requires a single vision and, pragmatically, I think that vision needs a single organizational source.

Since I don't see IT handing over "the vision thing" – much less responsibility for system development - to learning technologists, I expect we'll see more intrinsic performance support when internal IT departments understand the value – or necessity – of performance support, and when they gather the skills necessary to implement the intrinsic PS vision. The key to IT getting it, in my opinion, is convincing business executives that PS offers a viable solution to their problems, thus creating a demand that IT would be crazy not to try and meet. Within IT, a natural place for the performance support vision to take root is with human factors professionals. I'd look for them to demonstrate some leadership – and I'd suggest that we in the performance support community target them for our attention.

A wild card in all this is the potential of outside IT vendors to supplant internal IT providers. I think there's a niche for mid-sized external vendors with extraordinary skills to present viable solutions to business executives, thus taking business away from internal IT departments who do not adapt.

A wilder card still comes from a completely different direction. I've spent some time lately with the staff of an advertising agency – specifically with their interactive media group. These folks are doing intrinsic performance support applications that are embedded in their clients' web sites, serving customers. Their entrée into the systems development arena comes through their involvement with the overall corporate image, which must be reflected in the client's web site. They seem to be doing a successful end-run around IT because their focus is on e-commerce while IT is distracted by Y2K and ERP. Business clients are lining up for their services because a) they see the direct connection to business performance, and b) by and large, these systems are being built from scratch (though they certainly connect to back-end legacy systems).

Finally, the other way I see intrinsic performance support happening is the way it first happened at Aetna over 10 years ago: By attacking smaller performance chunks and building tools – from scratch – rather than systems. They can be built quickly, they can target critical tasks (and thus demonstrate dramatic results), and they can be built without significantly different skill sets from those of the best learning technologists.

Marc: Don't forget about meeting needs and building relationships. It's all the *human* stuff we tend to ignore in our quest for the PS holy grail. The key to building intrinsic performance support is to get the customer and the builder of the system to understand *early* the value of PS - even get the customer to *tell* the builder to do it. We need to educate the customer who is paying for the system. It doesn't matter if you use cost-benefit, a really cool demo, or a directive from on high-- as long as you get the support you need early on. I think the future is promising in this area as front line people become more savvy and thus demand of performance-centered solutions - as we see happening in e-commerce. They will go right to the supplier they believe will do the best job. Stay tuned....

¹ See http://www.epssinfosite.com/dd_index.htm

So what has changed? This is the first time in history that we see organizations *demanding* performance-centered systems - in the e-commerce world. Such systems first compete for attention, then must enable immediate business transactions. We've said for years that the *performance* in PS is *business* performance, enabled through human performance; now we see why. The e-business world may not be using the term performance support, but *usability* - a necessary condition for PS - has become the mantra and a common demand (Nielsen, 2000). Thus e-commerce systems provide powerful examples to help shape the demands of business owners around their need for performance-centered systems. The web alone does not enable performance, but it's role in e-commerce helps to surface PS as a significant business enabler.

PS practitioners can also take advantage of the emerging field of EAI because its outcomes enable us to create performance-centered interfaces. When the business processes, system navigation, and data of complex, diverse systems are merged to a single entity, greater opportunities exist for realizing PS in interface design. Today EAI concerns itself primarily with data integration. When tools emerge that integrate processes and superimpose user workflow onto communicating process- and data structures, then the opportunities for PS will be extremely rich and more results measurably rewarding.

While much of our discussion focused on enabling performance for work that is primarily system-mediated, we cannot forget just how much work is not. PS must step up to its proper role in providing a performance-based structure for a much broader range of tasks and jobs, such as bank relationship managers, high-end sales people, and brokerage branch managers. This highly generative work (that is not as simple as generating purchase orders on big systems) requires PS to embrace the notions of knowledge management - continuous capture, maintenance, validation - and dynamics, in terms of environmental sensors and agents of communication. In this context, PS manifests itself in bridging the gap between knowledge and performance.

Learning and the learning technologies remain important to PS, but it is clear that you cannot merely re-label training as PS simply because there is a related performance *issue*. Unless there is an explicit performance *outcome*, it's not PS. Business and IT departments have traditionally turned to training departments to close performance gaps, but have gotten training programs instead. It's time that the training function finds its place in the broader consortia of skills associated with performance - information design, human factors, knowledge management, interactivity, measurement, performance analysis, and the related technologies - and forges relationships with IT and business.

Finally, PS practitioners will continue struggling to remove the pain created by enterprise systems. Simple business tasks are rendered complex and difficult in the ERP world, where developers continue to forget the human being in the process. If the model that represents performance includes the human being along with the technology, then we will get closer to supporting performance: *business* performance, through human performance. In the meantime, many practitioners will continue to develop and bolt on extrinsic supports - both to enable performance but also to earn a seat at the development table. The seat alone gets you nothing, but as PS tools grow in sophistication and measurable effectiveness, the opportunities to create intrinsic supports and true performance-centered systems will continue to grow.

References

Nielsen, J. (2000). *Designing web usability*. Indianapolis, IN: New Riders Publishing

Pfeffer, J., and Sutton, R.I. (1999). *The knowing-doing gap: how smart companies turn knowledge into action*. Cambridge, MA: Harvard Business School Press

Rand, A. (1943). *The fountainhead*. New York, NY: Bobbs-Merrill

Bios

Gary J. Dickelman is Vice President of the Consulting Division for GURU, Inc. He specializes in the application of human factors engineering, learning technology, information technology, and business

process engineering to business systems analysis through implementation. Dickelman's work includes electronic performance support systems for human resources management, customer service, investments and pensions, health care, and business process re-engineering. Dickelman is a member of The Society for Applied Learning Technology (SALT), the American Society for Training and Development (ASTD), and the Association for Computing Machinery (ACM). He is a contributing author to *Using Computers in Human Resources* (Jossey-Bass, 1992) and *The Instructional Technology Handbook* (McGraw-Hill, 1993). Gary can be reached at gdickelman@pcd-innovations.com.

Gloria Gery is an independent consultant based in Tolland, Massachusetts. She developed the performance support concept -- and works with numerous sophisticated clients in developing performance centered software applications. Her emphasis is in strategy development and in creating the necessary managerial approaches. In addition, she assists in developing creative interface and program structures to assure learning and guarantee performance. Gloria's knowledge and experience in managing complex change combines with her understanding of computer technology to assist in critical systems implementation planning. She is the 1998 inductee into the HRD Hall of Fame sponsored by Training Magazine and is the author of [Electronic Performance Support Systems](#) (1991) and [Making CBT Happen](#) (1987) the respective seminal works on Performance Support and Computer-Based Training.

Stan Malcolm serves as a coach to businesses wishing to develop or advance their strategies for technology-based learning, electronic performance support, or corporate learning in general. He also manages the STEP consortium: senior professionals and executives at large corporations who share non-proprietary best practices and the costs of joint research efforts related to learning and performance. Before forming his own firm in 1995, Stan headed the learning technologies function of Aetna, Inc. for 11 years. For the third year, he is coordinating the performance-centered design competition. His web site is: <http://www.performance-vision.com> Please stop by and visit.

Barry Raybould is one of the founding pioneers in the field of EPSS and performance-centered design. He is a recognized leader in methodology in this field and pioneered the performance support engineering discipline for developing performance-centered systems within a knowledge management framework. He founded one of the leading consulting companies in this field and has just recently completed a two year effort to document and create a new Performance Support Mapping® training program to help organizations rapidly build up in-house expertise in performance support engineering and performance-centered design. Barry can be reached at BRaybould@arielpse.com.

Janet Cichelli has spent the past 8 years at WPI growing one of the industry's premiere performance support practices. Under Ms. Cichelli's direction, her award-winning group provides performance support consulting and development services to a wide range of government and commercial organizations worldwide. Ms. Cichelli specializes in performance-centered systems design, learning architectures, knowledge base design, and innovative approaches in performance support implementation.

Marc J. Rosenberg is

Hal Christensen is