In the last issue of KMR, Dave Snowden looked at three common mistakes companies make when they implement KM programs. Here, in Part 2, he explains how to get it right, through "just-in-time" solutions that use natural knowledge flows that already exist in the organization.

THE KNOWLEDGE YOU NEED, RIGHT WHEN YOU NEED IT

"Just-in-time" points the way to next generation KM

By Dave Snowden, IBM's Cynefin Center for Organizational Complexity

Dave Snowden is Director of IBM's Cynefin Center, which focuses on the application of complexity science to management, including knowledge management.

The time has come to look at positive interventions that can avoid the errors identified in the last issue of KMR. My overall argument is not that things can't be managed, but that they can't be managed in the sense that a machine can be managed. One origin of the word manage in English is a French word which means the ability to ride a horse in dressage. This is a more appropriate understanding than command, control and engineering meanings.

The following list establishes a set of principles that are not intended to be prescriptive or universal. Each is illustrated by a current knowledge management practice, both established and experimental.

Ensure a diversity of approach

Technology is a useful tool, but should not become a fetish. A good tool fits the hand and is largely forgotten despite its usefulness; a bad tool in the technology field too often requires not only the hand, but also the brain, to be bio-reengineered to enable its use.

Mechanical systems have a tendency for conformity, and creation of "universal" good practice, as there is, in theory, an ideal design. Organic systems in contrast encourage diversity. One of the main implications of this is to avoid purchasing a knowledge management system in the way that enterprise-wide resource planning systems are purchased. Each organization is unique; and individuals work in different ways. Some have messy desks, while others are neat and tidy; some enjoy virtual chat, while others despise it; some multi-task, some work on one task at a time. Attempts to impose uniformity on the rich diversity of human behavior will damage the knowledge base, and will not succeed. A knowledge management system should be a sound, resilient architecture with many different tools for communication and collaboration. Different communities and individuals can then gravitate to those tools that most naturally support their modus operandi.

In one study at IBM[1] we identified over 60,000 private collaboration spaces for just over 50 formal knowledge communities. Those numbers relate only to those who used technology for collaboration, so the ratio in practice is even more extreme. Not only is it impossible to formally manage the content of over 60,000 private virtual spaces, but it is also not desirable. A fully structured system with taxonomy, etc., would only discover or manage knowledge that we currently know that we need. It would not discover the knowledge that we might need to know in the future.

For example, IBM did not know that it needed to know about narrative or storytelling skills when it originally designed its formal knowledge spaces,

IBM's Cynefin Center for Organizational Complexity builds on and extends traditional knowledge management approaches through a complexity-informed approach called "Organic KM." For more information, visit www.ibm.com/services/cynefin.

KEYPOINTS

- There are three common mistakes in KM system design: assuming that human beings are always rational, that experts automatically share all of their knowledge on request, and that workplaces can be managed systematically, like a machine.
- "Just-in-time" KM addresses these mistakes by offering solutions that enhance natural knowledge flows that already exist.
- Examples of "just-in-time" KM are narrative databases, social network stimulation, limiting e-mail use, apprentice systems, and expertise locators.

but the existence of a private collaboration meant that the knowledge was self-organized at little or no cost. As a result it was available when needed, in the right context. The fact that it was easy and non-intrusive to set up a private collaborative space meant that it was done. Also, the selective membership was built on existing or newly formed social obligations. I participated in a private space where I had an obligation to fellow "believers" in narrative long before it was respectable because they were fellow believers, whereas participation in a formal community was a chore.

Don't impact on people's time

Time is the main enemy of knowledge management, followed closely by fears of abuse, and, at a distance, by the more common desire to retain power.

In the modern organization, e-mail traffic based on one-to-one communication has risen to levels where it has a growing negative impact on knowledge worker productivity. We can see this in common e-mail practices, such as mass copies, blind copying, task avoidance through a request for new data just to get an e-mail "out of the system," and many others.

Many aspects of e-mail use are showing all the evidence of addiction. Within the Cynefin Centre we have started to devise programs in which we use some of the techniques developed in rehabilitation clinics to break addictive patterns of one-to-one communication, often by cold turkey techniques: closing e-mail down for a period before bringing up collaborative systems, habituating people to many-to-many tools before allowing one-to-one but then banning all copies and all attachments.

To allow true cooperation for knowledge tasks, a good general rule to apply is this: any knowledge task for a knowledge worker requiring a time commitment, requires a prior explicit gift of at least twice that time. If you want a group of consultants to devote 40 minutes a day to knowledge sharing, then give them a low paid clerk to handle expense forms and timesheets – then they will help. Simply appeal to their sense of duty and loyalty to the organization and you are doomed.

A developing practice in the just-in-time field is the use of narrative databases. Faced with a choice between drawing down best practice from a knowledge management "system" and hearing the stories of eight or nine trusted individuals about their experiences, the vast majority would choose the stories. Narrative works in the same way, recording experiences as they occur, and accessing them through high abstraction criteria such as archetypes and themes to reflect the natural process of enquiry. As I enquire of the system I can record new stories, and hot link both new and old stories to more explicit knowledge material. A word of warning though. There are a lot of amateurs operating in the field of organizational storytelling (many of them from story telling professions such as journalism, who think the skills transfer without amendment), and secondly, too many organizations can't resist telling people what the stories mean and which they should read.

Another interesting approach in this area is the growing use of apprentice systems. Borrowing unashamedly from medieval practice, apprentice schemes evolved to transfer tacit knowledge through observation, coaching, and practice. These techniques are not only more effective than manuals and computer-based training, but they are also one of the only ways of embedding and validating knowledge transfer between people.

Ironically, the full life cycle costs of knowledge acquisition and deployment of apprentice schemes are also generally cheaper than approaches based on rigid codification. Self-selecting apprentice schemes can also be created, using some of the social network stimulation techniques outlined below.

"THE BETTER EXPERTISE LOCATORS RESPECT THE PARADOX OF PRIVACY: IF I ALLOW SOMEONE TO KEEP THEIR KNOWLEDGE PRIVATE THEN THEY WILL SHARE IT, IF I TELL THEM TO SHARE IT THEY WILL KEEP IT PRIVATE."

Connect people, build networks

The number of respected pioneers in knowledge management who emphasize connecting people over codification is only matched by the number of managers who read and praise their thinking, then proceed to do the exact opposite. We resist the concept of arranged marriages, so why should anyone think the same approach can apply

universally to community formation?

The following are two examples of new practice:

1. Don't build 'Yellow Pages' use an expertise locator

Yellow pages rely on people to codify their expertise and maintain data. In the early days, they are novel, so people participate. But usually within a year the systems are neglected or abused, as people either do not have the time, or manipulate their entries based on perceived futures within the organization they work for. Expertise locators, on the other hand are non-intrusive in that they either map affinities based on access to knowledge or trawl e-mail to indicate evidence of expertise. The better ones respect the paradox of privacy: if I allow someone to keep their knowledge private then they will share it; if I tell them to share it they will keep it private. To be asked if I know something in the context of the need, is more likely to elicit a response than to be asked to codify knowledge in the absence of that context.

How narrative databases work

Narrative databases attempt to recreate as closely as possible the ways people share knowledge in face-to-face encounters at work or socially. There are two basic steps to create a narrative database. First is "capturing" the stories and second is creating an index that makes them useful. Capturing the story uses tape recorders or interviews in the field.

Below is an example of a narrative, provided by IBM, which has a patent on the anecdotes and the indexing. Archetypes for this story would be "apathy" and "principled." The stakeholders for this narrative would be "staff," "bankers," and "students." The theme would be "harmful effects."

The database might produce this narrative from various searches. For example, someone in the marketing department might be searching for material to produce a new ethical product to differentiate the organization from the competition. They would ask the database to call up stories around the theme that products can be harmful, told by students. Or a trainer could be looking for material for a new training course for branch managers and ask the database to call up stories around the theme that products are harmful. The story would also be useful for a manager assigned to a new location who is trying to understand the issues he or she will be involved in.

The value of narrative databases is that queries produce several anecdotes, giving users many different perspectives.

Name of narrative: "Someone was killed years ago"
Transcript: "The first time in my life I met a banker, any banker, I mean, the first I saw in my life was while we were in Australia a month ago. We actually wanted to talk to this banker about student activities and other stuff, so credit cards were not the topic of the story. But then he, somehow we talked about the range, and then he came up with a story, that – no, he doesn't sell credit cards to students. Why? They are bad. Why? Because in his university, when he went to school, somebody was killed by using credit cards. (Killed?) Killed, yeah. (How killed?) Yeah, he killed himself, because he couldn't face his parents. Oh and by the way, he was depressed. When did it happen? Ah, maybe twenty years ago. Who still knows about it? Yeah, maybe the doctor in the infirmary. (But he's retired.) But he's retired, yeah – you were – two of us. (People who don't know.) So."

2. Social network stimulation (SNS)

SNS[2] works by taking the natural process of network building and accelerating it. Normally, over several years, as I work on different projects in different departments, and meet people socially, I build a network, and hear stories that allow me to operate effectively in an organization. Mentor schemes, good induction, and narrative can all help. However, it is now possible to reduce five years of accidental networking to five months, or even five days in a task-based environment. In SNS we focus less on managing knowledge, but on managing the channels through which knowledge flows.

A common theme in the above examples is to allow communities to form based on natural preferences, although SNS does it within a set of top-down, determined heuristics. Telling people they should work together can clash with the basic chemistry of human interaction and is inefficient.

See where people walk before you build paths

A good designer observes patterns of human interaction before designing – for example, planting grass and observing where people walk before investing in a path. Of course you can also grow a hedge or build a bridge to encourage patterns.

Here we need to understand that in dealing with complex systems, all interventions are also a form of diagnosis and vice-versa; partly because any attempt to study a complex system changes the system being studied. This means that in managing a complex system (and by implication all knowledge management is predominantly about complex rather than ordered systems), we intervene with multiple small probes so that the most natural patterns of behavior are revealed. Some examples will illustrate this approach:

1. The way we represent human requirements is key here. There is an old adage amongst IT professions that users never really understand what they want by way of a system functionality until they get it, at which point they want something different. Equally, many business users are pretty much convinced that the IT profession created user requirement specifications to ensure that they would sign up to something they could only partially understand, but for which they can be held accountable later.[3]

This tendency has been all too present in knowledge management system design. Narrative techniques, particularly those which represent abstract features of organizational culture through archetypes and themes, provide a new way of creating a dialogue between system designers and users. Archetype families provide a representation of the different attitudes, experiences and belief

systems of a user community. Allowing a designer to put any feature into a system, subject to their being able to create a coherent story about how each archetype will respond to that feature, is far richer than esoteric interpretations of a formal document. In effect, this creates a fictional space of discourse, which allows ideas to be tested and evaluated in advance of implementation.

2. In forming communities, it is always better to use an existing, naturally occurring community – formal or informal – where working practices and trust are already developed, than to impose a utopian design. Most knowledge management practice has gone the utopian route, which is not to say that it has not worked, but when it has, the cost has been prohibitive.

Social Network Analysis techniques identify naturally occurring communities that can be formalized according to natural affinities. When a naturally occurring community cannot be identified, we can use an insect model – create the metaphorical equivalent of a bright light and see who swarms. The cohesive entity that results becomes the new community. These models will satisfy most organizational requirements for communities of practice.

3. Narrative databases provide a quick and easy way to see how a "lessons learned" program should be implemented.

Any knowledge holder can record into a tape recorder in two to three minutes what they would otherwise spend a few hours writing. The sheer volume of material that can be captured in narrative form is far higher than can ever be formally codified. Because such systems are self-indexing they can be built and populated quickly, allowing the central designer to observe patterns of use before committing to the cost of codification. The designer can see which stories are accessed, in what context, and how they are used in practice, rather than trying to hypothesize on the basis of interviews which do not reflect the real nature of acts of knowing in the first place.

Conclusion

All of the above examples rest on observing the natural process of knowledge creation and transfer, then using technology as a tool, if appropriate, to ensure that such natural processes can be accelerated and scaled. In all cases they focus on self-organization, albeit within imposed boundaries. In complex systems we embark on journeys and respond to the environment; in ordered systems we determine our goal and then engineer to achieve it. The results and the approaches are different and complementary.

Just-in-time is not just about embedding knowledge into process through artefacts, although that is valuable and often necessary, but expensive. It is also about creating a human ecology in which knowledge self-patterns, and being aware of when those patterns are stable enough to justify the cost of embedding. It is about diversity within boundaries, using technology as a supporting tool, not as an overriding fetish, and above all, about recognizing that humans are more than capable of achieving results without constant control. Very few senior executives would dream of treating themselves or their children in the way that they treat their employees, and it is time for less hypocrisy in KM and associated system design. KM

References

1. Published in Liberating Knowledge referenced above and updated in Complex Acts of Knowing.
2. A technique paper on SNS is available from the author.
3. This is a direct quote from a soon-to-be published article relating the use of archetypes in an advanced KM system design linking methods from both IBM and Microsoft.

Contact
Dave Snowden
IBM's Cynefin Center for Organizational Complexity
E-mail: snowded@uk.ibm.com