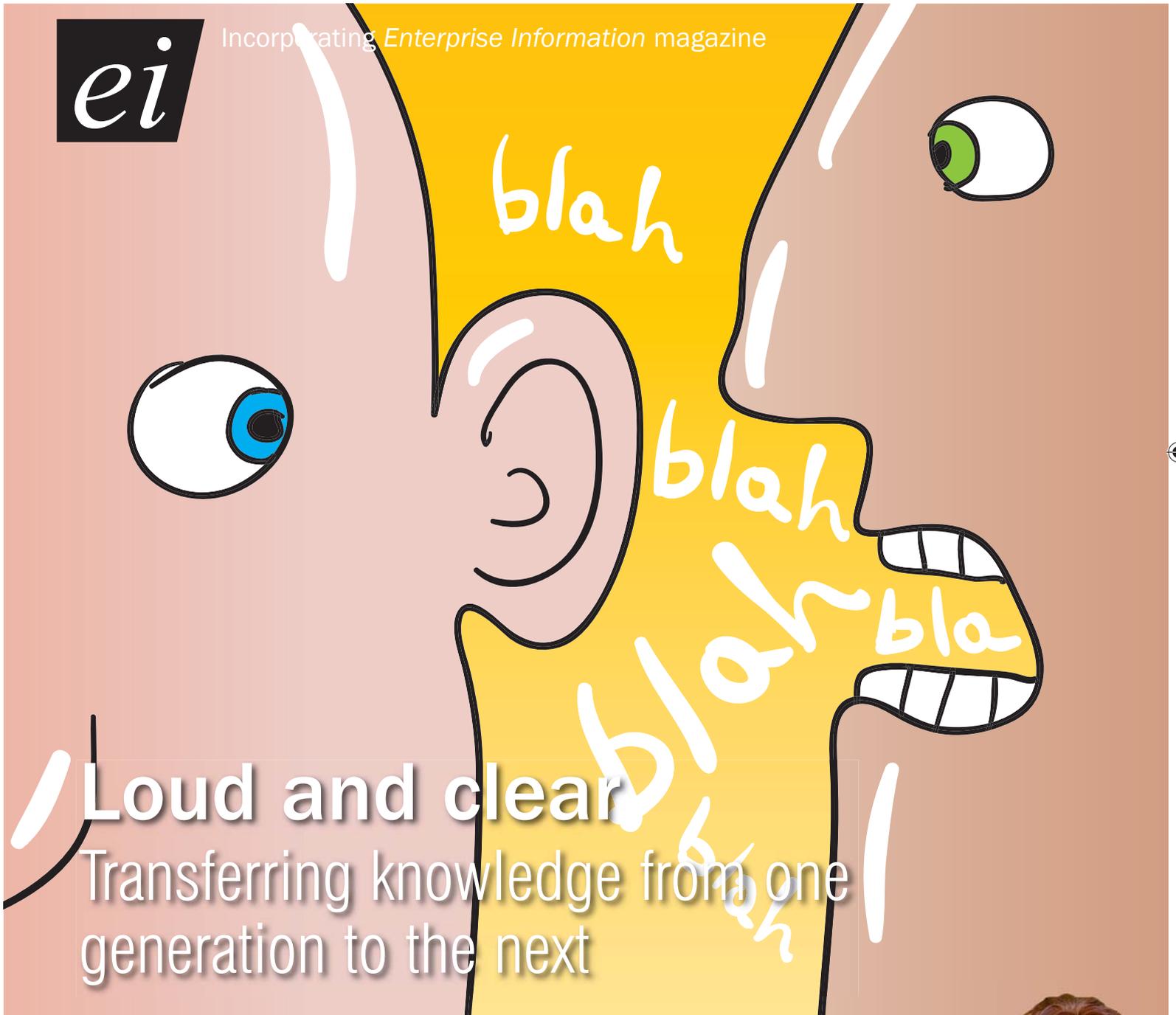


Inside Knowledge



Incorporating *Enterprise Information* magazine



Loud and clear
Transferring knowledge from one generation to the next

CASE REPORT

Jerry Ash investigates Guerrilla KM at Hewlett-Packard Services.

MASTERCLASS

Andrew MacNeil looks at the peculiar challenges of change management.

NEWS UPDATE

Facebook privacy warnings... David Gurteen goes conferencing... plus the latest events...

THE KNOWLEDGE

How Debra Amidon went from Digital to KM guru.



HP Services Consulting & Integration

Sowing the seeds of KM

'Guerrilla knowledge management' may be an unconventional and imperfect approach, but it does provide fast and cheap innovation.

By Jerry Ash

No matter how advanced and widely heralded a knowledge management (KM) programme might be, it's frustrating to the architects to find that a large portion of the internal audience doesn't have a clue. The architects do everything they can to improve usability, add features, promote and provide KM training, but improvement is both slow and incremental.

A case in point is the very mature KM programme at Hewlett-Packard's HP Services Consulting & Integration (C&I) where the team reaches effectively beyond the business unit to the wider services organisation and other divisions within the corporation. Yet KM programmes there run into the same problems any initiative does within a large organisation. KM is just one of many, sometimes overlapping, programmes competing for employees' attention.

"Getting our message across to the rank and file in the cacophony of messages from human resources, training, the project-management office

– never mind memos from managers, senior managers and vice presidents – can be very difficult," says Andrew Gent, lead knowledge architect for HP C&I, part of a four-member team under Stan Garfield (see *Inside Knowledge*, December 2005/January 2006).

Gent had spent 25 years in large computer companies – Digital Equipment, Compaq and HP – starting in technical documentation and ending

the consulting organisation and finally ended up on the KM team.

While the team continues to promote and maintain the overall KM programme, Gent and the team have also found ways to innovate without taking resources away from ongoing programme management.

Over the past ten years, C&I has established a stable and mature KM programme run by Stan Garfield's

**"We identify a problem; someone says 'I have an idea'; and as long as it fits within the basic rule of 'fast and cheap' that person owns responsibility for the solution."
Andrew Gent, HP Services Consulting & Integration**

up doing information architecture within the domains of documentation, training, usability and corporate strategy.

It was along those lines that he was first introduced to the services area. Eventually he was asked to develop an overall collaboration strategy for

group. However, KM programmes (good or bad) run into the same problems any initiative does within a large corporation. "Sometimes you wish you could get one or two quick wins or do something innovative to help the troops," says Gent.

“Traditionally this would involve a proposal for a new project, justifying the ROI to get funding – and possibly head count – and then spending months or even years developing, testing, revising and finally releasing. This is neither quick nor – in the end – innovative since any risky parts are usually argued out of it during the proposal and testing phases.”

Thus, return on investment (ROI) and innovation are examples of the proverbial chicken and egg syndrome. You cannot prove the ROI without a demonstration of value in production. But the point of ROI is to justify getting something new into production. Guerrilla KM bypasses the twin problems of getting attention and getting going. It offers opportunities for small, quick wins while avoiding the entire ROI-justification cycle.

Streamlining

What makes guerrilla KM different from other forms of KM is the goal and the approach. There are three aspects:

1. Goal. Rather than tackling large, broad-brush, fuzzy problems, the goal is to tackle small, distinct, concrete but irritating issues important to the user;
2. Approach. It's one of simplicity; there's no fancy project plan, no reviews, no approval needed and the project is completed quickly (usually in two weeks or less);
3. Rollout. Unlike official projects that can enforce usage, guerrilla projects go for a quiet, grassroots rollout and voluntary adoption based on how useful it is to the user.

“Design on these projects tends to be ‘inspired,’” Gent says. “We identify a problem; someone says ‘I have an idea’; and as long as it fits within the basic

rule of ‘fast and cheap’ that person owns responsibility for the solution.”

The guerrillas will occasionally test out ideas on a limited audience – just to verify that it is a real problem they’re solving. But in general, they avoid user feedback in the design stage as that would take up too much time and bog the design team down in too much detail, too early on.

“This is counter-intuitive to most user-centric design approaches,” Gent says, “but quite frankly, more often than not users will push the design beyond the bounds of the original, simple scope. When that happens you have to stand your ground or else the entire effort will become a full-scale project with all of its incumbent strictures, requirements and controls in no time flat.”

Although the roll-out is low-key, it's not intended to be low-impact. The solution is offered to users, often by word of mouth, with no big announcements, no training and offered ‘as is’ with support only as time allows. “Adoption is limited at first,” Gent says. “On the other hand, a benefit is that we

rarely get the resistance you sometimes find in large programme roll-outs. Instead, we tend to get constructive criticism, suggestions and even an occasional offer of assistance.”

Another twist on rollout, Gent advocates that the solution be made as broadly accessible as possible, not just within his own business unit. With open access to the solution, you get a larger ‘base’ audience. Adoption occurs a little faster and sometimes other organisations pick up on the solution faster than the group it was originally intended for.

It also leads to simpler development. Since Gent is not concerned with restricting access, he can avoid complex security concerns that could significantly impact development costs. In summary:



- Attack a concrete, discrete problem;
- Design the simplest solution possible, using what you have at hand;
- Give it to the users to decide if it is useful/successful or not.

“That last item is key. Because of the approach, with none of the user input of normal development, there is a fairly high risk of missing the mark and you can expect at least some of the projects to fail. That is not a bad thing. It is just part of the process and part of the learnings. Better to fail after two weeks of development than six months to a year. As a consequence, when I announce a new project of this kind, often the first question I ask is not ‘What’s missing?’ or ‘Where can we make improvements?’, but ‘Is this useful?’ If yes, we have succeeded. If no, we try again – or move on to the next project,” says Gent.

The four keys

Gent hasn’t developed the ‘principles of guerrilla KM’ yet, but believes that the following represent key attributes:

- *‘Fly under the radar’*
The project must fit within existing schedules and resources so it won’t be noticed by anyone until it is complete;

- *‘Fast, not perfect’*
If the saying ‘good, fast, cheap – pick two’ is true, guerrilla KM always goes for fast and cheap. The solution has to be sufficient and nothing more. The designs have to be as simple as possible. Bells and whistles are saved for the next version. In fact, there should always be a wish-list of desirable features. If not, you’ve probably failed to meet your original goal;
- *‘Stick to your guns’*
Everyone will have something to say about the effort and how it should be done. The goal of guerrilla KM is to choose a solution and complete it as fast as possible. Once it is finished, then you can decide if it is successful once people actually get a chance to try it.

Gent is a strong believer in what he calls ‘loose integration’. He tries to build each project so it has no absolute dependency on specific software. He favours use of public interfaces, preferably web services. Several years ago he became a strong advocate of RSS as a readable interface for users and, more importantly, as an integration point for other applications.

“This is the sort of loose integration I’m

users (name, address, phone and so on). What me@hp adds is a very simple structure and several additional content areas: a photo, a biography (clear or formatted text), a list of ‘friends’, and a list of areas of interest. This last is what attracts many people, because unlike the biography, which is free text, the ‘areas of interest’ feature is a ‘folksonomy’ that can be browsed. (A folksonomy, of course, is a user-generated taxonomy that can be used to categorise and retrieve web content such as web pages, photographs and web links).

The tool is very simple, was built quickly – and is very handy.

Gent is a strong believer in what he calls ‘loose integration’. He tries to build each project so it has no absolute dependency on specific software.

He favours use of public interfaces, preferably web services.

- *‘No new hardware’*
This is a corollary of the first principle. Don’t design anything that would require higher-level management approval. In other words, no additional resources or hardware beyond what can easily be expensed as a miscellaneous cost;

talking about,” Gent says. “Almost everything I build has RSS in it somewhere.”

One example of simple guerrilla-KM design was the development of a MySpace-like programme called me@hp, which takes advantage of the existing knowledge of internal

“Again, adoption is slow, but since there is almost no maintenance cost, it is actually a good platform for further experiments in social computing. Most recently we added ‘forum membership’ underneath ‘areas of interest’. This is automated and pulled dynamically from the forums server via RSS [really simple syndication] feed,” says Gent.



Willing to fail

To engage in guerrilla KM, you must also be prepared to fail. Projects do not always succeed and it is hard to predict success in advance. Here's a case in point: in me@hp, a member's profile shows a list of the forums she or he is subscribed to. On Gent's task list was a plan to do the reverse – create a list of forums where a person could easily find everyone who was subscribed.

The membership directory was created using Ajax and the forum RSS

To be fully accepted, knowledge management has to be an integral part of the work, not another layer of responsibility. It has to make the work easier, not harder. It has to help people be more efficient and effective.

feeds, where you could extend and collapse a list of the forums and their members; each member's name was linked to their me@hp profile. "It was greeted with a resounding 'meh'," Gent said. He blames the disinterest on the fact there was no problem being solved. A membership directory had been requested, but it was (and still is) unclear how it would be used. Overall, the membership directory can be considered a failure.

"It was perhaps our most 'expensive' failure since it took almost a week and a half to complete," Gent jests. But in consequence, Gent built several smaller components that are already in use in other places.

Sowing seeds

One of the key axioms of KM must surely be this: to be fully accepted, knowledge management has to be an integral part of the work, not another layer of responsibility. It has to make the work easier, not harder. It has to help people be more efficient and effective, not further burdened by more work than they can handle. And it has to show clear benefit for the organisation's critical success factors.

'Nice-to-haves' rarely survive; 'must-haves' are critical to survival.

Andrew Gent's low-key guerrilla tactics aren't acts of aggression, but shades of Johnny Appleseed, a piece of American folklore based on the real life of John Chapman, born in 1774 near Leominster, Massachusetts.

He turned up in the then Northwest Territory from which the states of Ohio, Michigan, Indiana and Illinois were formed following the Revolution. In the early 1800s, Chapman was among the first to explore the new territory. In anticipation of pioneer settlement, he entered the wilderness with a bag of apple seeds, chopped out the brush by hand and planted the seeds. He did all the work himself, living alone for weeks at a time with only the local Indians and wild animals for companionship.

When the first settlers arrived, John Chapman's young apple trees were ready for sale. He spent the rest of his life producing and distributing this staple of early life. It became a culture. Today, of course, there is nothing more American than apple pie.

Andrew Gent may not become a folk hero, but he is well on his way to sowing the simplest seeds of KM in the most fertile places. Watch them grow. ■

Jerry Ash is KM coach, founder of the Association of Knowledgework (www.kwork.org), special correspondent to Inside Knowledge and author of Next Generation Knowledge Management I & II. To order either of these, contact Adam Scrimshire at publishing@marketing.ark-group.com. Andrew Gent will be guest moderator for AOK's STAR Series Dialogues 17-28 September. AOK membership is free and you can join online at: www.kwork.org/explain_join.html.