VOLUME 2 NUMBER 1

JOURNAL OF CYBER SECURITY & INFORMATION SYSTEMS

A Knowledge Management (KM) Primer

By Mark Addleson, PhD

It can be difficult to navigate your way around the field of knowledge management (KM). Whether you are just starting out and thinking about putting ideas about KM into practice, or you work in a organization that has had a KM initiative in place for years, at times it may be hard to see what KM is about, what people are doing, and why and how they are doing it. For, although the field has been evolving for at least twenty years, there is a very broad spectrum of ideas about what KM is (the theory and principles), how to do it (the practices) and what *not* to do (Fahey and Prusak, 1998; Snowden, 2007).

One sign of trouble in this field is that there are many definitions of KM.[1] Another is that, while lots of organizations claim to be 'doing KM,' their strategies often have little in common. A third is that KM can be very technical, so KM initiatives become complicated, often unnecessarily so.

In this primer, I want to answer three questions.

- Why KM?
- What is KM about?
- How do organizations undertake KM initiatives?

If you are involved in KM, I hope my perspectives will help orient you and I would be pleased to receive your questions, comments, or suggestions.

In the beginning there was management without knowledge

Management practices, as we know them today, began in factories, machine shops, and foundries towards the end of the Nineteenth Century, when mass-production methods became more prevalent (see Crainer, 2000; Witzel, 2012). The generally recognized starting point for contemporary management practices is 'scientific management,' indelibly linked to the name of Fredrick Taylor (1911), the inventor of time and motion studies and founder of the management consulting industry. By 'eliminating waste,' his object was to improve the productivity of manual workers and cut costs to make industrial organizations more efficient and more profitable.

Today, whether they work in government agencies or accounting firms and whether they are involved in aerospace engineering, health care legislation, or web design, most people are *knowledge workers* (Addleson, 2011). As Table 1, below, reveals, knowledge-work and factory-work are completely different. Because the different kinds of work have nothing in common, you can't manage knowledge workers – or their work – as if they were assembly line workers. In most organizations, however, you find principles and practices that evolved in factories, similar to those advocated by Taylor. It is hardly surprising, perhaps, that these conventional management practices are obstacles to doing knowledge-work.

Factory-work	Knowledge-work
Physical	Mental
Solitary (think 'production line')	Social (think 'network')
Routine and repetitive	Complex and dynamic
Talk is a distraction	Talk ('sharing knowledge') is necessary
Tools (like blueprints, machines, and breakeven charts) are essential.	Tools are needed, but all work starts with and is guided by 'talk' – people in conversation.

TABLE 1: Comparing factory-work with knowledge-work

One answer to the questions, where did KM come from and why are organizations doing it, is that KM provides tools and techniques to bring management into the Twenty-First Century. A well thought-out, fully implemented KM initiative can help to eliminate out-of-date industrial management practices. A good KM initiative will enable

people to organize and run today's organizations – government departments and agencies, for-profit businesses, as well as non-profits – as knowledge organizations need to be run, with employees – who often work in teams – collaborating and sharing knowledge (Bryan and Joyce, 2005; Linder, 2005; Sandow and Allen, 2005).

Knowledge management evolved from earlier 'change management' efforts that included 'Total Quality Management' (TQM) (Martínez-Lorente, *et al*, 1998) and 'Process Reengineering' (Macdonald, 1998) and 'Organizational Learning' (Yeo, 2005). These emerged about midway through of the Twentieth Century (Prusak, 2001; Lambe, 2011). The goal in each case was to *improve the way organizations worked* – to make them more effective and/or efficient. In retrospect, we can see that each of these efforts shared at least two important features, which contradicted old-style factory-management practices.

1. They advocated decentralization: greater reliance on 'local' knowledge and experience, instead of trying to run everything from the top with rigid rules, plans, and structures. The thinking here is that the people doing the work, with *practical knowledge based on their experience* of *how things work*, know most about how work processes can be improved and are the first to see problems when they arise. Workers 'on the ground' are usually in the best position to respond to changing circumstances, but they need to have *the authority to use their knowledge*, make decisions, and take action when necessary.

2. People need to 'share their knowledge'. If you want to devolve decision-making down to the local level, it is no good isolating individuals and groups in organizational silos (e.g. separating them by department) or behind top-down structures, which make it difficult for subordinates to communicate with superiors. You need to devise systems, structures, and cultures that make it easy to share knowledge, or 'move it around'.[2]

Knowledge management represents the further evolution of these ideas. Where TQM and Reengineering were devised originally with the object of designing new work practices and processes to make *industrial* firms more efficient, knowledge management is a creature of the *information age*.

These are some basic premises of KM:

a. people need knowledge and information to do their work;

b. today they have access to lots of information (and some people work almost exclusively with information);

c. accessing information and sharing knowledge enables them to do a better job – solve problems, work smarter, and produce better results;

d. there is technology available to help people access and analyze information and share knowledge;

e. conventional (factory-style) management practices don't pay attention to knowledge or information: to what knowledge/information people need, how they get it, whether they share it, and so on; and

f. in most organizations there are many barriers to accessing information and sharing knowledge

There isn't much agreement about what KM is

It is difficult to find a definition of KM that two people agree on and many fields, from IT to lawyers and librarians, claim KM as their own. As each has different knowledge-related needs, they advocate different practices.

Here is my attempt at a brief explanation of what KM is and why we need it.

When people do anything they *use knowledge and they often access information* (e.g. reading a book because they are doing research, mining a database for information about customers' buying habits). In most situations it requires more than one person to get something done and *people share knowledge*. At one end of the knowledge-sharing spectrum, in a person-to-person phone conversation, you might find a mechanic at a car dealership ordering brake rotors from a parts supplier. At the other end, where hundreds or even thousands of people, with different roles, responsibilities, and expertise are involved in a large-scale defense contract, at any moment, working in teams, various groups may be planning or reviewing some aspect of the design or testing of hardware or software.[3] Focusing on the interconnections between *work* (getting things done),

information, and *knowledge*, KM revolves around fundamental questions like:

- what information and knowledge do people need,
- how do they use it,
- where do they get it,
- do they have it,
- what obstacles are there to getting, sharing, and using it, and
- what will help them get it, share it, and use it more effectively

Wherever there are knowledge workers, questions like these help them accomplish whatever they are doing. The last one is most closely related to action, but, from the standpoint of KM, *all are practical questions,* in the sense that informed answers contribute to a better, more functional workplace.

The reason for asking these questions is usually framed in management language; for example, 'improving efficiency', 'making the organization more competitive (or more profitable)', 'getting things done quickly and cheaply', but the goal is the same. Unless people can do effectively what they aim to do (control air traffic, make new policies, protect critical infrastructure, care for the sick, and so on) organizational objectives won't be met. So, it is helpful, when doing KM, to keep the fundamental questions and people's work firmly in mind and *stay focused on the connection between these questions and the work people are doing*.

> *KM is about improving the experience and quality of knowledge-work, recognizing the importance of information and knowledge for getting work done well.*

Information and knowledge: is there a difference?

One of the considerations that trips up people doing KM is a lack of clarity about information and knowledge and their differences. Although a great deal has been written on questions like, is there a difference and, if so, does it really matter, the responses, unfortunately, often generate more heat than light. While they are philosophical, these questions are also intensely practical, because how you answer them shapes not only the way you think about KM, but also how you practice it.[4] To individuals who treat information and knowledge as interchangeable, the main purpose of KM, typically, is to provide employees with access to the right *technical information*. When they see knowledge and information as different, however, recognizing that people collaborate to get things done, the object of KM first and foremost is to create an environment where colleagues can readily share their knowledge with one another. In the former case, KM usually falls under IT; while, in the latter, instead of being subsumed under IT, KM may be the responsibility of a group in human resources or organization development.

Steering clear of philosophical debate about what knowledge is and how people acquire it, I will explain briefly why knowledge and information are different, although they are closely related in a symbiotic way. Anything you regard as *inform*ation informs – so, is useful – because you can and do place the material in the context of what you *already know*. Information 'fits' your (pre-existing) understanding. If something is beyond your knowledge and comprehension it is non-sense; it cannot inform.[5]

Knowledge

Knowledge is what you, or other people, know. If you have children you have knowledge about them: their ages, their likes and dislikes, their personalities, and so on. If you are a materials fabricator, you probably know what it takes to bend and cut and how to join metals and composites. Some of this you've probably acquired from books, the web, or from talking to colleagues.

It is common, nowadays, to distinguish between two types of knowledge: *explicit*, in the form of principles, theories, and facts about the world, lots of which fall under the heading 'technical knowledge'; and *tacit*, acquired largely from experience. These are sometimes referred to, respectively, as 'know *what*' (explicit) and '*know how*' (tacit). I know about the tensile strength of metals and the number of instructions a microprocessor is capable of handling every second and I know that the Empire State Building is 450 meters high, even though I've never experienced (seen) these directly. 'Know-how' implies an ability to get things done and to deal with problems or issues. I know how my children respond to different situations, I know how to jump-start a car, and I know how to stay upright on a bicycle without having to think about it.

It is also widely acknowledged that most of what we know is tacit and, among KM practitioners, there a fairly widely held belief that it is desirable, as well as practical, to turn tacit knowledge into explicit knowledge. Once 'captured,' they argue, it can be transferred to others (who will be able to access it as information) (Nonaka and Konno, 1998; Nonaka and

Toyama, 2003). Organizations, concerned about protecting their 'intellectual capital,' for example, are prompted by consultants to prevent useful knowledge 'walking out of the door' when employees resign or retire. They may then go to considerable lengths and incur significant costs to capture the knowledge of retirees, gained from years of experience on the job, then make it available to others

There are several reasons why, rather than rushing to embrace them, these kinds of initiatives, that include transferring knowledge in the form of lessons learned from people in one project team to another, should be treated with some skepticism and approached with caution. One reason is the growing recognition that tacit and explicit knowledge are different *types* of knowledge. It isn't practical to turn one into the other. Each is important and useful in its own way and they are complementary, not substitutes (Cook and Brown, 1999). There are also question marks over the knowledge that organizations manage to capture. Is it useful to others – either contemporaries or future generations – and, if so, in what form and under what circumstances?

The problem is that *knowledge always has a context* and you can't take it from its original context – the varied circumstances and life-experiences of the knowers – and put it into files or databases without it losing at least some of its *meaning*. One way to understand this problem is to consider how difficult it is to explain to someone who has never experienced a different culture how natives of the culture express their feelings. This is the kind of tacit knowledge you acquire through experience. You can explain to a stranger 'facts of the situation,' for example what people say and do when they greet one another, but this doesn't allow him or her to 'get' the culture. To know it, they have to experience for it for themselves, by participating in it.[6]

To clarify my position on the difficulty of capturing and transferring knowledge, it is time to return to the distinction between information and knowledge. There are many situations where people need and – as long as someone provides it – can *acquire information* that helps them either to do something they otherwise could not do, or to become more proficient at doing it. If they already have a common context of technical and other know how, doctors, engineers, lawyers, software developers, plumbers, or musicians can learn a lot from the information in instructions or other documents created by colleagues. But, with different backgrounds or fundamentally different experiences – when they have different ways of knowing and have to find common ground in order to proceed, when they have to *discover* what is going on, what others mean or intend, or what to do, when, and with whom – people's ability and willingness to collaborate and make sense of the situation (coming to understand it) together is paramount. Now, *sharing knowledge* takes priority over 'transferring information'. The information they can access is, at this point, less important than their ability to 'find a way forward' *together*.

Information

In contrast to knowledge, which people possess – they 'have knowledge,' think of information as 'out there' on websites, in databases, on menus, and in instruction manuals and blueprints. What was once someone's knowledge in the form of ideas, perspectives, or points of view, information is now in a kind of limbo waiting to be found.

It's not what is out there that is information. Whatever is out there becomes information only *when someone, seeing it as useful*, 'adopts' it and uses it. Whether they stumble upon it serendipitously or are consciously looking for ideas, a reference, or 'additional information' to help them with something they are working on, at the point at which they 'connect' with it, finding it interesting or believing it is useful, it becomes part of their knowledge (i.e. what they know) for a time. It is a common mistake to treat knowledge and information as if they are completely separate things. Knowledge – what we know and – information – which we acquire – are complementary. We find and use information because we have knowledge of how and where to look for it, plus an understanding of what we are looking for and some sense of what is likely to be useful and why.

Without a *context* of existing knowledge (i.e. what you already know), information is useless. In fact, without that context it is wrong to call it information, because it does not inform. Telling you the Empire State Building is 450 meters high is literally meaningless to you unless you know numbers, understand what a meter is, know what a building is and, more specifically, are interested in the height of buildings and the Empire State Building in particular. This is to say that 'stuff' is not information *unless people can make some meaning of it and,* when they do, *it is knowledge (i.e. it is what you, or they, know).* You are surely familiar with stories about inventors who, initially, were unable sell what later turned out to be very practical ideas (the invention of Xerography - photocopying technology – is one example), because potential investors who they tried to convince couldn't 'see' the significance of their ideas. They had no context for appreciating the information they were given.

They didn't have the knowledge to assimilate it.

Much of the knowledge that we use to first find information and then use it is tacit. If I am in a restaurant and want to know what there is to eat, I know to look at the menu, or to ask the person who comes to serve me, particularly if I can't understand (make meaning of) the menu because it is in a foreign language or it describes dishes from a country and culture I don't know. I know, too, that a search engine is my door to lots of potentially useful information, but, until I learn (and know) how to use it, all this information is 'hidden', as if it doesn't exist. When I buy a new piece of technology, I look for instructions on how to use it, but if the technology is far from what I already know, because I don't have a context of existing knowledge, I might not be able to understand the instructions. They won't provide me with useful information until I call a friend for help or ask an expert to help me.

Knowledge is social

These examples point to an important consideration about knowledge. Much of what we know isn't in our heads. It is social – held and shared in groups or communities (McDermott, 2002). Because knowledge (or knowing) is social, because we share experiences and the meaning of ideas, experiences, values, and beliefs, we're able to communicate, share knowledge, and collaborate.

As I'm sure you have discovered, however, shared experiences and shared meaning only go so far. You have been working on a project, with the same people, for some months and, just when you think you 'know how another person thinks' or believe 'you're all on the same page', someone's actions suggest that you really don't know what motivates them or, perhaps, that they haven't understood what you said or what you expected from them.

One of the complexities of organizational life is that we work with and are expected to share knowledge with people who have very different interests and experiences, even when they are from the same organization. Nowadays, the people we work with are often from different, even competing organizations (Addleson, 2011). When there are two or more prime contractors and many more subcontractors on a very large project – as you find, for example, with any Major Defense Acquisition Project (MDAP) – innumerable organizational, occupational, and interpersonal boundaries exist in the multiple networks of professionals who must interact and share knowledge in order to do the work. These differences contribute to breakdowns, when work gets done badly and the whole project may run into difficulties, which is one important reason why we have to pay attention to knowledge and really work at ensuring we are sharing it effectively.

Two approaches to knowledge management

It is important to understand the relationship between knowledge and information, because this has a bearing on how organizations approach KM and it also explains why many KM initiatives fail to live up to expectations.

Organizations undertake KM initiatives in order to improve efficiency, because they see KM as a way of becoming more competitive, of reducing costs, and so on. KM will have these benefits *if* it enables people to be more creative, to work smarter, to be more productive and, generally, to do better work. Earlier, I said that in order to make sense of KM – to appreciate what it is about and also to understand what works and doesn't work – it is necessary to keep an eye on the relationship between work, knowledge, and information. Now, we can begin to see why.[7]

The nature of knowledge-work

Here are a few examples of knowledge-work

- Organizing and coordinating teams designing the hardware for the navigation system of a surveillance drone.
- Deciding what kind of information to extract from a huge database of customers' purchases collected by a supermarket chain, then writing algorithms to extract the information.
- Tracking down the people responsible for committing a bank robbery.
- Assisting customers who are subscribers to your cloud-based hosting service to set up their sites.
- Designing a guidance system for an air-to-air missile.
- Developing a training program for employees in your HR department.
- Testing the security of a large government agency's information systems.

Now, here are a few of the characteristics of this kind of work:

Many people are involved in getting things done: employees of the organizations, their customers and clients, contractors, suppliers, and so on.

Typically, much the work is done by an assortment of project groups or teams. These may be comprised of people with different skills and technical qualifications. From time to time teams need to interact with other teams, both from the same or different organizations, who may be spread out across the globe.

The problems people deal with in order to get things done are often ill defined. They don't have clear-cut objectives, a predetermined time-line, and in some cases they don't even know who they are going to work with, as people are assigned and reassigned while the project or task is in progress.

Even before they begin to 'solve problems' their work involves 'setting' the problem: deciding what they are doing or what issues they are dealing with, then deciding what they're going to do about the situation and who should be involved, setting schedules, and getting commitments (Schön. 1983).

This work is what we call *'organizing'*. Knowledge workers spend a lot of time organizing.

They do this by interacting and talking to one another on the phone, in person, and by email. In fact, much of their work consists of conversations. Before a defense project is funded there are rounds of discussions and negotiations, among a multitude of stakeholders, including potential contractors, politicians, and senior officers, offering proposals, doing evaluations, and providing counterproposals. At different times these groups draw on individuals with a variety of skills, from negotiators to cost estimators to proposal writers. And, with the object of deciding what comes next as well as assessing what's been done, the pattern of *sharing knowledge* – talking, asking questions, offering advice, listening to what clients and colleagues have to say, getting commitments, providing updates on what they have accomplished, and so on, continues throughout the project until the contract is eventually put to bed, perhaps a decade or more later.

From these few points we conclude that:

- Work is very social. It involves people continuously interacting with one another.
- Knowledge-work is also cooperative in the sense that people *need to collaborate* in order to define and solve problems together.
- Conversations are central and, when you observe them at work, you realize how much time knowledge workers spend on the phone, on email, or talking to others in conference rooms and corridors. They can get little done unless they talk to each other, sharing their knowledge; and unless they are *willing* to collaborate they won't share knowledge. By talking to one another they find out what the issues are, what has been done so far, what needs to be done, what kinds of problems people are experiencing, and so on.

Now, these behaviors – working together collaboratively and talking to one another, sharing knowledge, are not the norm in most organizations. Under 'old' rules of management, which evolved in the factory system:

- Competition, rather than collaboration, is expected. People compete with one another in order to climb the ladder to the top or to earn bonuses and bigger paychecks.
- Action is valued more than talk. In fact, employees are generally discouraged from talking.
- Employees are expected to work alone, rather than cooperate.

The design of office space illustrates the last two points. You find employees sitting behind cubicle walls, isolated from each other.

KM version 1

One approach to knowledge management says the real purpose of knowledge management is to correct the deficiencies of conventional management practices. Knowledge workers can't function effectively in a factory-management culture. People need to talk to one another, they need to cooperate (collaborate) more than they need to compete, and as it takes a team (even teams of teams) to do the work, we should be rewarding team-effort rather than individuals.

From this standpoint, the fact that they are doing knowledge-work, not assembly-line-work, changes everything (Allee, 2000). KM, viewed as any and all actions that encourage and enable people to collaborate and, in the process, co-create and share knowledge, should be as ubiquitous, necessary and natural for organizations as breathing is for humans.

Adopting KM version 1 means recognizing that knowledge management is potentially deeply subversive. Its *purpose is to change the way we manage work* – making any and all changes necessary to ensure that knowledge workers are able to do and produce good work. In the interests of creating a culture where teams really do work as teams, where people

are able to leverage their combined knowledge to solve wicked problems (Conklin, 2006; Marshak, 2009; Rittel and Webber, 1978), produce good software, or deliver excellent services, we need to examine every practice to see if it stands in the way of generating and sharing knowledge. Nothing should be sacred.

KM version 1 begins with questions like:

- Our work depends on collaborating and sharing knowledge, what does it take to do it well?
- How well are we doing and what are the obstacles?
- How do we deal with them?

Only when people understand the relevance of these questions and have good answers should they address more 'technical' ones related to 'intellectual capital' and 'talent management' such as:

- What kinds of knowledge/experience do we need?
- Who has this knowledge?
- How do we ensure that the people who have it are connecting with those who need it and vice versa?
- What kinds of tools will help people collaborate and, how do we encourage people to use them in ways that foster collaboration? (see Wenger, White, and Smith, 2009)

KM version 2

A fundamentally different approach to KM, which is very popular, KM version 2 focuses on tools and data (or 'content') more than, and in many cases instead of, people and practices. Most organizations with KM initiatives actually do KM version 2, even if they talk as though they are doing version 1. There are probably two reasons for this. First, KM version 2 is not subversive. It fits well with conventional management practices. The other reason is that people who are responsible for KM often have not thought deeply enough about knowledge and work and haven't asked the deeper questions, about why they are doing KM, what they hope to accomplish, and what it takes to get there.

It is fairly easy to tell whether organizations are doing KM version 1 or 2. Version 2 is characterized by a highly technical KM language and by budgets that are heavily oriented to IT, to technologies like portals, databases, and search engines, and to activities like 'knowledge engineering,' knowledge capture and retrieval, information retrieval, enterprise architecting, data mining, and categorizing information (creating taxonomies or developing ontologies). KM version 2 is an approach that tends to see and treat knowledge and information either as completely separate (and to focus on information, mistaking it for knowledge) or to blur their differences. So, when people doing KM version 2 talk about 'collaboration', they often mean moving information or data around, rather than people interacting and sharing knowledge as they make meaning together (Addleson, 2013).[8] KM version 2 should probably be called 'information or data management' rather than KM.

Why it is necessary to keep these two approaches to KM separate

At the end of the day, doing knowledge-work well – *producing good results* – depends on people collaborating and sharing knowledge. This is the bottom line of knowledge-work and the object of KM version 1. People need to share knowledge and, no matter how sophisticated your technology, no matter how good your search engines, or how detailed your taxonomies (i.e. no matter how hard you pursue KM version 2), if they won't share knowledge or don't do so effectively you have a problem: your teams and project groups become dysfunctional and projects run into trouble and fall short or fail.

Most organizations struggle with the problem of sharing knowledge, but few are tuned into the reason for the struggle; they manage knowledge workers using outdated, high-control factory-management practices and KM version 2 is compatible with these practices. For example, knowledge workers need to network – and networks are loose and flexible – but organizations rely on rigid, top-down reporting structures. Instead of paying attention to these issues, to the culture the enables people to organize their work in fluid networks, with flexible plans (and deadlines if necessary) and agile practices – clearly this is a tough nut to crack because it requires everyone to think and act differently – organizations focus attention on KM version 2, opting for 'technological fixes.' Here, they get assistance from vendors who claim, misleadingly, to sell KM in a can (i.e. a computer/server). When they install this software, purchase this search engine, create a portal, build the right workflow processes, and so on, 'information rich,' employees will work smarter, quicker, be more productive, and organizations will be more innovative, more competitive, and more profitable.

Community of practice or community of interest?

Alongside portals, document repositories, directories, and search engines communities of practice (CoP) are an integral part of many organizations' KM initiatives. There is good reason to be pleased that word about CoP spread quickly (the term was coined by Jean Lave and Etienne Wenger in 1991) but there is a downside too. CoP is an over-used buzzword. What organizations call 'CoP' are often communities of interest (CoI). The difference is significant, as I will explain.

Knowledge-work is collaborative, creative, and synergistic. Knowledge workers get things done by interacting and sharing knowledge, when they draw on their experience to answer colleagues' questions or give advice, when they swap stories or try to fathom out – together – what is going on, what they need to do and how to do it. Communities of practice, as the name suggests, have to do with the way people do their work (i.e. with their practices) but, to appreciate what makes a CoP and why they are important for KM, the words 'community' and 'practice' must carry equal weight.

The word 'community' suggests a group of people who are quite intimate with one another; connected not only by intellectual interests (e.g. rocket scientists who design missiles) or formal work requirements (e.g. individuals designated as 'members of the "Blue Team"') but also by interpersonal relationships like friendship and/or loyalty and/or collegiality. Perhaps they show genuine affection for one another, with each one caring about what the others do or don't do. When people 'live in community,' they tend to see each other quite often, know quite a lot about what the others are doing, and be generous in helping and supporting one another when they need it. This describes the relationships among people in a CoP.

Turning to practices, Etienne Wenger identifies three elements of their work practices that give members of a CoP a sense of belonging to and participating in a shared enterprise (Wenger, 1998. 72-85. See also Wenger, 2004; Wenger *et al*, 2002).

- Their '*mutual engagement*,' or the fact that they are actively involved in doing the work together.
- The fact that they see their work as a *'joint enterprise*' and, as they interact, continuously discuss and clarify what they are doing, what constitutes 'good work' and whether what they've done is up to standard, and so on.
- '*A shared repertoire*' of resources in the form of shared routines, artifacts or tools, a common vocabulary, and, perhaps, similar ways of thinking, even dressing.

Julian Orr (1996) and others have written detailed accounts of CoP, explaining how they work and what makes them different from regular teams and the kinds of interactions people typically have in organizations. Some of the characteristics of CoP, which you might expect in a community, are:

- Limited hierarchy. Members treat one another as peers. Authority is based on age, experience, and expertise rather than rank.
- Limited competition. Relationships are collegial and cordial and competition is friendly (e.g. demonstrating problem-solving skills rather than rivalry for promotion).
- It is seldom 'strictly business'. When members chat they will talk about their families, share their concerns about bosses or colleagues, and so on.

From the standpoint of KM, CoP have virtues that are particularly important. One is that participants readily share knowledge. CoP are good – some might say ideal – knowledge sharing contexts. The other is that they are to a large extent self-organizing. Rather than compliance, relying on instructions and rules from above, it is participants' accountability to each other, as well as their mutual commitment to their 'joint enterprise,' that ensures the job gets done and gets done well. Self-organizing is a particular virtue in environments where things are constantly changing and experience is paramount. Rigid rules and formal structures impede rather than assist people in getting the work done.

As you might expect, organizations that understand and practice KM version 1, emphasizing flexible work processes, with groups sharing knowledge and organizing themselves, are generally better at supporting CoP. Sharing knowledge is everyone's business. This means a culture of openness. While CoP can emerge in all kinds of environments, they are more likely to thrive when there is openness rather than top-down control.

You'll often find groups called 'communities of practice' in organizations that have adopted KM version 2, emphasizing tools and technology ahead of people and practices. Most of the time, however, these are, at best, *communities of interest* (CoI). Members of CoI are interested in the same 'body of information' – not necessarily work-related – and, often, have little else in common. They may be members of the same profession (e.g. lawyers; scholars) and/or have a similar domain of expertise (tort reform; medieval religion). Sometimes a CoI is comprised of individuals with the same hobby such as sci-fi, model trains, or gardening.

As these examples suggest, the participants need not be in the same organization and CoI are often virtual groups of people who only contact one another online, as members of a user- or interest-group. In common with members of a CoP, sharing their ideas, knowledge, or information (in the form, say, of articles, drawings, or URLs) helps CoI members get things done, whether at work or play. This is valuable but, when it comes to KM, it is only part of the story. It is 'cooperation' rather than 'collaboration,' which means 'working together'.

Members of a CoP generally work together closely and, as joint contributors to the work, *co-create* the results, whether this is a

PowerPoint presentation or a piece of software. To understand the difference and why collaboration is highly desirable, you need to appreciate that knowledge-work is deeply creative. When software developers start a project, for example, they seldom know where they will end up. Their 'product' comes to life and evolves while they work, in the work, as they interact and talk among themselves and with their clients. Without the back-and-forth, the meetings, conversations, and networking, little would be accomplished.

The bottom line is that CoI are necessary, but not sufficient, for people to do good knowledge work. When a KM initiative is mainly tools and technologies (KM version 2), the IT department that takes the initiative in setting up SharePoint sites for team members to share ideas or ask for advice is helping the cause of KM; but by how *much* depends on a variety of factors. If the organization is hierarchical, their online contacts should help people work around the barriers to knowledge sharing (e.g. between superiors and subordinates) created by hierarchy. Yet, you can do this without affecting the culture and ultimately it is the culture (whether people do or don't want to share knowledge) that matters.

If, for example, their business involves working with 'big data,' or if they have highly sensitive information that needs to be secure, organizations surely must have a heavy IT focus (i.e. KM version 2). Focusing on IT, however, is never the end of the story in terms of getting work done. In fact, in most cases it is just a small part of the story. Knowledge-work means people getting together, interacting, talking, sharing knowledge and creating new knowledge in order to solve problems, deciding what to and how to do it, then guiding and assisting one another in actually doing it. Making this happen takes KM version 1. Organizations that are mainly doing KM version 2 won't get the results they want. Poor knowledge management practices and limited collaboration will consistently hamper them. It is worth thinking about your organization's stance on KM? Are you doing KM version 1, version 2, or both? And, how well does KM serve you?

To answer the question, how should we organize knowledge-work, you need look no further than Agile methods, like Scrum (among many sources of information on Agile, see the articles on Ken Schwaber's website http://www.controlchaos.com and Mike Griffiths' blog, 'Leading Answers' at http://leadinganswers.typepad.com/leading_answers/). Although they are associated with software development and project management, agile methods should serve as an example for all knowledge-work. As the name indicates, these methods have evolved with flexibility at their core. Agile recognizes tacitly that, in spite of their best efforts to do so, people may not be able to see and plan very far ahead. Instead, they figure out what to do (and, possibly, where they went wrong) while actually doing the work – discussing, planning, designing, and building – with one another, with other teams, and their clients. So, Agile practices rely on stakeholders interacting (i.e. cooperating and collaborating) frequently, sometimes daily (even if only for a few minutes) to share knowledge; and they rest on the premise that, as those doing the work know better than anyone what is going on, it is best for them to organize themselves (see Addleson, 2011).

The Team Software Process℠ embodies many of the best practices for supporting knowledge-work including: coaching, team building, collaborative planning, and regular meetings to assess and report on the status of the work and to revise plans. Watts Humphrey (2000), the developer of TSP, says it was clear, early on, that the success of TSP depended on management providing broad support for the process. This is true of KM in general and, as good KM practices frequently run counter to 'old' management practices, letting go of the old ways may well be the main obstacle to implementing a successful KM initiative.

About the Author

Mark Addleson is an Associate Professor in George Mason University's School of Public Policy, where he teaches in the Organization Development and Knowledge Management Master's program (ODKM). Mark taught for more than 20 years in his native South Africa at the University of the Witwatersrand's Graduate School of Business Administration and was a director of Econometrix, a consulting firm. An Associate of The OCL Group, LLC, he has consulted with a range of organizations – for-profit, non-profit and government – both in South Africa and the USA. His areas of research include organizing knowledge work, knowledge management, and organizational change and he is widely published. His book, *Beyond Management*, about how to organize knowledge-work, was published by Palgrave Macmillan in 2011. Mark also has a blog 'Management is Dead', at http://www.managementisdead.com.

References

Addleson, Mark. 2013. "Will the Real Story of Collaboration Please Stand up so We Can See It Properly?" *Knowledge Management Research and Practice* 11 (1) (February): 32–40.

Addleson, Mark. 2011. *Beyond Management: Taking Charge at Work*. Houndmills, Basingstoke, Hampshire, UK: Palgrave Macmillan.

Allee, Verna. 2000. "Knowledge Networks and Communities of Practice." *OD Practitioner* 32, Fall/Winter (4). Available at: http://www.vernaallee.com/images/VAA-KnowledgeNetworksAndCommunitiesOfPractice.pdf

Bryan, Lowell L., and Claudia Joyce. 2005. "The 21st Century Organization." *The McKinsey Quarterly* (3): 24–33.

Conklin, E. Jeffrey. 2006. *Dialogue Mapping : Building Shared Understanding of Wicked Problems*. Chichester, England ; Hoboken, NJ: Wiley.

Cook, Scott D.N., and John Seely Brown. 1999. "Bridging Epistemologies: The Generative Dance Between Organizational Knowledge and Organizational Knowing." *Organization Science* 10, Jul/Aug (4): 381–400.

Crainer, Stuart. 2000. *The Management Century: A Critical Review of 20th Century Thought and Practice*. San Francisco: Jossey-Bass Publishers.

Fahey, Liam, and Laurence Prusak. 1998. "The Eleven Deadliest Sins of Knowledge Management." *California Management Review* 40 (3): 265–76.

Humphrey, Watts S. 2000. "The Team Software Process (TSP)." Technical Report CMU/SEI-2000-TR-023. ESC-TR-2000-023. Available at http://www.sei.cmu.edu/reports/00tr023.pdf

Lambe, Patrick. 2011. "The Unacknowledged Parentage of Knowledge Management." *Journal of Knowledge Management* 15 (2): 175–97.

Lave, Jean, and Etienne Wenger. 1991. *Situated Learning: Legitimate Peripheral Participation.* New York: Cambridge University Press.

Linder, Jane. 2005. "How Do Things Really Work Around Here?" *Across the Board* 42 (6) (November): 24–29.

Macdonald, John. 1998. "The Quality Revolution - in Retrospect." *The TQM Magazine* 10 (5): 321–333.

Marshak, Robert J. 2009. "Reflections on Wicked Problems in Organizations." *Journal of Management Inquiry* 18 (1): 58–59.

Martínez-Lorente, Angel R., Frank Dewhurst, and Barrie G. Dale. 1998. "Total Quality Management: Origins and Evolution of the Term." *The TQM Magazine* 10, (5): 378–386.

McDermott, Richard. 2002. "Knowing Is a Human Act." *Upgrade* 3 (1): 8–10.

Nonaka, Ikujiro, and Noboru Konno. 1998. "The Concept of 'Ba': Building a Foundation for Knowledge Creation." *California Management Review* Vol 40, 1, Spring (Special Issue on 'Knowledge and the Firm): 40–54.

Nonaka, Ikujiro and Ryoko Toyama. 2003. "The knowledge-creating theory revisited: knowledge creation as a synthesizing process." *Knowledge Management Research & Practice* 1 (1): 2–10.

Orlikowski, Wanda J. 1993. "Learning from Notes: Organizational Issues in Groupware Implementation." *The Information Society* 9, 3: 237–250.

Orr, Julian E. 1996. *Talking About Machines: An Ethnography of a Modern Job*. Ithaca, NY: Cornell University Press.

Prusak, L. 2001. "Where Did Knowledge Management Come From?" *IBM Systems Journal* 40 (4): 1002–6.

Rittel, Horst M.J., and Melvin M. Webber. 1973. "Dilemmas in a General Theory of Planning." *Policy Sciences* 4: 155–169.

Sandow, Dennis, and Ann Murray Allen. 2005. "The Nature of Social Collaboration: How Work Really Gets Done." *Reflections, Society for Organizational Learning* 6, "Innovations in Practice" (4-5): 1–14.

Schön, D. (1983). *The Reflective Practitioner: How Professionals Think in Action*. New York: Basic Books.

Shingo, Shigeo. 1989. *A Study of the Toyota Production System from an Industrial Engineering Viewpoint*. Portland, Or.: Productivity Press.

Snowden, Dave. 2007. "1998 and all that, a return to sin." http://www.readability.com/articles/dxj6sxx6?print=1

Taylor, Fredrick Winslow. 1911. *The Principles of Scientific Management*. New York: W.W. Norton and Company Inc.

Yeo, Roland K. 2005. "Revisiting the roots of learning organization: A synthesis of the learning organization literature." *The Learning Organization* 12 (4): 368–382.

Wenger, Etienne. 2004. "Knowledge Management as a Doughnut: Shaping Your Knowledge Strategy Through Communities of Practice." *Ivey Business Journal* 68 (3): 1–8.

Wenger, Etienne. 1998. *Communities of Practice: Learning, Meaning, and Identity*. New York: Cambridge University Press.

Wenger, Etienne, Richard McDermott, and William M. Snyder. 2002. *Cultivating Communities of Practice*. Boston: Harvard Business School Press.

Wenger, Etienne, Nancy White, and John D Smith. 2009. *Digital Habitats: Stewarding Technology for Communities*. Portland, OR: CPsquare.

Witzel, Morgen. 2012. *A History of Management Thought*. 1st ed. New York, N.Y.: Routledge.

Endnotes

[+] I wish to thank Dennis Goldenson and Taz Dougherty for their advice and guidance, provided in conversations about this paper and in comments on earlier drafts.

[1] There is a collection of 42 definitions of knowledge management at http://www-958.ibm.com/software/data/cognos/manyeyes/datasets/43-definitions-of-km/versions/1

[2] Both sets of ideas are embedded in what has become known as the "Toyota Way" (See Shingo, 1989), which turns Taylorist management on its head.

[3] For examples of the need to share knowledge and the challenges of doing so in highly complex, networked organizational settings, we need look no further than the Major Defense Acquisition Programs (MDAPs), where thousands of people from many organizations, with widely different skills, interests, and affiliations, working in various teams, are contributing, in innumerable ways, to the development and production of a particular weapons system.

[4] As anyone knows who has delved into the distinction between knowledge and information, that it is a highly contentious area. Unfortunately, either because the matter is unsettled, or because people don't pay enough attention to the issues, knowledge and information are often treated as if they are interchangeable. People talk about 'knowledge' when they really mean 'information' and vice versa. Without claiming that my views are definitive or necessarily correct, I hope these ideas help both to reinforce the point that it is important to distinguish between knowledge and information and to stimulate you to think about the differences.

[5] People who, because of their training, or experience, or both, know (understand) differently, surely glean different information in the same circumstances; for example, a master mechanic and layman looking at an engine leaking something.

[6] The fact that much of what people know and need to know to 'understand the problem,' 'get the job done,' or 'find a way out of the mess' comes only from experience explains why it is so important to turn to those who have hands-on experience when drawing up plans, developing capability requirements for new systems, and so on. This fact also highlights a fundamental flaw in high-control management and administrative systems. In high-control organizations, the formal authority to act increases as you go up the chain of command and the greatest expertise is presumed to reside at the top of the organization. This combination often results in a particular type of hubris that leads to problems and breakdowns. Even though they have little or no practical knowledge on which to base plans or requirements, those at the top nevertheless plan and formulate requirements without advice from the people who have experience and they issue directives to subordinates who possibly understand the realities of the situation better than they do.

[7] For a fuller discussion of many of the points that follow, see Addleson (2011).

[8] By the 1980s, two kinds of software tools had appeared that supported collaboration. With one, like Ventana's GroupSystems, designed primarily to facilitate group decision-making, participants (typically aided by a facilitator) sat in the same room responding to common questions. The software aggregated their responses and seeing the results on a screen was a prelude to further conversations, debate, and deliberation. The other, like Lotus Notes, built as a client-server system, allowed virtual knowledge sharing, by participants who were possibly separated by both time and distance. Although this latter category of software, originally known as 'groupware,' has proliferated with the advent of internet-based social networking tools, in many organizations the tools still have not fulfilled their potential to support collaboration. 20 years ago, Wanda Orlikowski (1993), who had studied the roll-out of Lotus Notes in a large management consultancy, pointed out that the way tools are used reflects people's cognitive and technical frames, or perspectives. One reason why tools like SharePoint typically are used for storing data and accessing information, rather than as 'spaces' for sharing knowledge, is that the management mindset, which favors competition, doesn't 'get' the human-social dimensions of collaboration (as opposed to the technical possibilities for enabling it).