KCS v6 Practices Guide

The latest and most complete collection of information about how to do Knowledge-Centered Service (KCS®). This description of the KCS Practices and techniques is based on the collective experience of the Consortium members.

(v6 released 21 April 2016. See changes in Appendix A. This document was created as a pdf on 8 June 2023; the content is subject to change at any time. Please see the Consortium Library for the most current information.)

A good thing to read first is the KCS Principles and Core Concepts document, published 18 April 2016.

Appendix H was added in the spring of 2023: The "Why" Behind Each Practice

- Section 1 Knowledge-Centered Service
  - Why KCS?
  - The Attributes of Knowledge
  - KCS is a Journey, Not a Destination
- Section 2 The KCS Practices
  - The KCS Article
  - The Double Loop Process
  - The Solve Loop
    - Practice 1: Capture
      - Technique 1.1: Capture Knowledge in the Moment
      - Technique 1.2: Capture the Requestor's Context
      - Technique 1.3: Searching is Creating
    - Practice 2: Structure
      - Technique 2.1: Use A Simple Template
      - Technique 2.2: Complete Thoughts Not Complete Sentences
    - Practice 3: Reuse
      - Technique 3.1: Search Early Search Often
      - Technique 3.2: Seek to Understand What We Collectively Know
      - Technique 3.3: Linking
    - Practice 4: Improve
      - Technique 4.1: Reuse is Review
      - Technique 4.2: Flag It or Fix It
      - Technique 4.3: License to Modify
    - Closing Thoughts on the Solve Loop
  - The Evolve Loop
    - Practice 5: Content Health
      - Technique 5.1: KCS Article Structure
      - Technique 5.2: KCS Article State
      - Technique 5.3: Developing a Content Standard

        - Technique 5.4: Creating Evolve Loop Articles
        - Technique 5.5: Archiving Old Articles
        - Technique 5.6: Dealing With Legacy Data
        - Technique 5.7: Priming the Knowledge Base With New Information
        - Technique 5.8: Global Support Considerations
        - Technique 5.9: Knowledge Domain Analysis
        - Technique 5.10: Content Health Indicators
        - Technique 5.11: Self-Service Success
        - Technique 5.12: Self-Service Measures
    - Practice 6: Process Integration
        - Technique 6.1: Structured Problem Solving
        - Technique 6.2: Seamless Technology Integration
        - Technique 6.3: Search Technology for KCS
        - Technique 6.4: Closed Loop Feedback for the Whole System
        - Technique 6.5: KCS Process Adherence Review
    - Practice 7: Performance Assessment
        - Technique 7.1: KCS Roles and the Licensing Model
        - Technique 7.2: Coaching for Success
        - Technique 7.3: Assessing the Creation of Value
    - Practice 8: Leadership and Communication
        - Technique 8.1: Develop and Communicate a Vision
        - Technique 8.2: Create a Strategic Framework
        - Technique 8.3: KCS Benefits and ROI
        - Technique 8.4: Promote Teamwork
        - Technique 8.5: Tap into Internal Motivators
        - Technique 8.6: Recognition Programs
        - Technique 8.7: Communication is the Key
        - Technique 8.8: Leadership Accountability to the Knowledge Worker
        - Technique 8.9: Leadership and Communication Indicators
- Summary
- About the Consortium
- Appendix A: KCS v6 Update Summary
- Appendix B: Glossary of KCS Terms
- Appendix C: Metrics Matrix
- Appendix D: KCS Roles and Competencies
- Appendix E: Article Quality Quick Reference Guide
- Appendix F: KCS and ITIL
- Appendix G: KCS FAQs
- Appendix H: The "Why" Behind Each Practice
- RIGHT TO USE WITH ATTRIBUTION
- Special Thanks

Section 1 Knowledge-Centered Service

KCS v6 Introduction

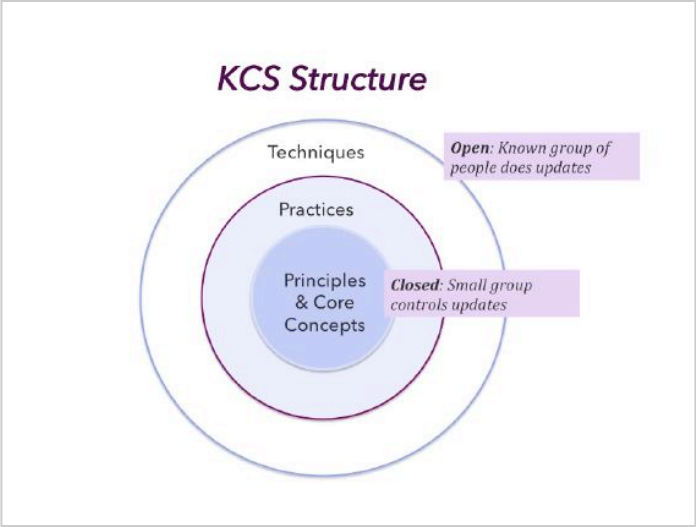
Knowledge-Centered Service (KCS®) is a simple idea that creates profound benefits. If, as part of our workflow, we can reuse, improve, and (if it doesn't exist) create knowledge that is available to the entire organization, we can make gains in our operational efficiency, increase success with self-service, and improve our products and services while enabling organizational learning.

These profound benefits are realized by doing something... profoundly different. For most organizations, KCS represents a shift in thinking. To successfully adopt and sustain KCS requires a new way to think about work, people, measures, and process.

The KCS Principles and Core Concepts precedes the KCS Practices Guide and should be referenced before exploring the KCS Practices. KCS is built on a set of fundamental beliefs (four principles) that explain *why* we are doing what we do. The ten core concepts (which are more detailed) are based on one or more principles. The KCS Practices Guide focuses on the *how*; this document describes the best we know to-date about how to make KCS work.

While KCS v6 uses the same acronym as previous versions of the methodology, we have updated the name from Knowledge-Centered Support to Knowledge-Centered *Service*. This change reflects the broadening application of the methodology outside of the support environment, and stems from the definition of service science, coined by IBM Research as "an interdisciplinary approach to the study, design, and implementation of services systems – complex systems in which specific arrangements of people and technologies take actions that provide value for others." (Wikipedia)

The KCS v6 Adoption & Transformation Guide complements the KCS Practices Guide and the Measurement Matters paper. The Adoption Guide is the accumulation of twenty years of experience by the staff of the Consortium for Service Innovation. This guide provides a step-by-step approach for planning a KCS adoption. Our aim is to share what we have learned about what makes for a successful KCS adoption and help avoid some of the common mistakes.



The KCS Academy is the source for KCS[sm] resources and the certifying body for KCS. The Academy offers KCS certification for people and KCS *Verified* and *Aligned* designations for tools and services. The KCS *Verified* and the KCS *Aligned* programs are useful for assessing technology to enable KCS. The KCS Academy is a network of KCS practitioners and vendors who support the successful adoption of KCS and contribute to the continuous improvement of the KCS methodology across all knowledge-intensive industries.

The KCS Academy is a wholly-owned subsidiary of the Consortium for Service Innovation. The Consortium is the non-profit association that developed and continues to maintain the KCS methodology.

The KCS[sm] methodology is service marked by the Consortium for Service Innovation.

A Note on Usage and Style

In writing this document, we have made two assumptions about its use. First, although we like to read it through repeatedly, we expect that few people will read the entire publication end-to-end in one sitting. We have chosen to repeat a few of the key ideas and concepts where reinforcement will improve understanding and enable each section to be meaningful when read independently.

Secondly, we are expecting that readers will come from different environments. Most of the Consortium member experience is from customer facing support and internal help desks. Different groups use different vocabularies. We have attempted to present the concepts generically so that they will be easily understood, and we hope our word choice will not limit comprehension. We have provided a glossary in Appendix B to help define the terminology we use in KCS.

We are ready to start the KCS journey.

Why KCS?

KCS breaks through the limitations of traditional knowledge management strategies and enables organizations to deliver greater value with more efficiency. The secret? Capitalizing on what they already have: knowledge. This increased value is created and managed by capturing the collective experience of the organization in solving problems and answering questions for customers. Making that knowledge reusable, and evolving it to reflect organizational-level knowledge, creates huge leverage.

There are three categories of benefits derived from KCS. They are realized at different points in time along the KCS journey.

1. Operational Efficiency

The first benefit realized is operational efficiency. Efficiency improves as knowledge workers learn to integrate the use of the knowledge base into their work. This means getting work done includes integrating the reuse of existing knowledge, improving existing knowledge, and capturing new knowledge if it doesn't exist. The degree to which the organization can build both a knowledge-centered culture as well as an infrastructure that facilitates interaction and improves or captures knowledge from those interactions as a natural by-product of getting the work done will influence the degree to which the organization can maximize the benefits.

Improvements in operational efficiency show up as increased capacity and the ability to introduce both new people and new work into the organization with dramatically less effort and time. Integrating use of the knowledge base into the workflow enables people to benefit from the collective experience of the organization. This reduces rework and provides knowledge workers with the most complete and accurate information based on the collective experience of the organization. The collective experience will always be more accurate and complete than any individual's knowledge.

2. Self-Service Success

KCS creates knowledge in the context of demand or use. As a result the knowledge is findable and usable by those who are looking for it. As the name implies, knowledge worker success and productivity is dependent on…. knowledge! Our goal is to provide knowledge that is known (captured and findable) to those who need it, at the moment of need. If we can capture and reuse our experiences, we can enable others to benefit from those experiences through self-service mechanisms. This greatly reduces the overhead of finding information by having to figure out who might know and how to get in touch with them. Effective self-service greatly expands access to what is known and allows knowledge workers to focus their energy on new challenges and opportunities.

3. Organizational Learning and Improvement

The final and perhaps most beneficial benefit we realize with KCS is the identification and prioritization of opportunities to improve our processes, policies, products, and services based on the captured experience. The patterns and trends that emerge from the reuse of knowledge allow us to conduct root cause assessment and take corrective action on the things that have the highest impact on our organizations' effectiveness and more importantly improve the productivity and success of those we serve.

The aggregate impact of these three benefits results in reducing the ratio of our operating costs to revenue (or costs per individual being served) while at the same time improving the success and productivity of those being served.

As organizations embrace a knowledge-centered culture and become proficient at reusing, improving, and capturing knowledge, the organization begins to function more like a network than a hierarchy. It is a network of people and content (captured knowledge). The network connects people to content for known issues and people to people for new issues. A knowledge-enabled network, as an organizational structure, is way more agile and adaptive than the traditional rigid, siloed, hierarchical structures that are so pervasive in business today.

Member Experience

Consortium members who have implemented KCS in either their internal or external support organizations are reporting dramatic improvements in incident resolution and training times, in customer satisfaction, and in employee job satisfaction. As a result, they are realizing substantial savings in operating costs while seeing improvements in service levels.

They find they can:

- Solve Cases and Incidents Faster
  - 50 - 60% improved time to resolution
  - 30 - 50% increase in first contact resolution
- Optimize Use of Resources
  - 70% improved time to proficiency
  - 20 - 35% improved employee retention
  - 20 - 40% improvement in employee satisfaction
- Enable Web Success
  - Improve customer success and use of self-service
  - Up to 50% case deflection
- Build Organizational Learning
  - Provide actionable information to product development about customer issues
  - 10% issue reduction due to root cause removal

People choose to adopt KCS because they need to scale and extend their organization's capabilities in a world of limited staffing and budget resources.

The Attributes of Knowledge

For most organizations the adoption of KCS represents a major transformation.

The transformation starts with an understanding of the attributes of knowledge. We use the word knowledge in business conversations all the time. But when asked to define knowledge, most people pause. A definition is not immediately available off the top of our heads; it requires some thought.

It is helpful to put knowledge in the context of data and information. What distinguishes data from information? Data is just numbers or words, while information is organized numbers or words. The organization of data into information gives it some meaning. What distinguishes knowledge from information? Knowledge is information upon which I can act. Knowledge has action associated with it; we can do something with it.

The definition of knowledge is an ongoing debate in academic and philosophy circles that goes back as far as Plato. We find that for our purposes "information upon which I can act" is a helpful definition. KCS seeks to capture the collective experience of the organization in ways that others can use. "Use" or "act on" being the key point.

If we accept the definition proposed above we can move on to identifying some of the key attributes of knowledge. First we must recognize that information I can act on, or use, is dependent on my having some context or experience with that information. That is, I have to already know some things that complement the information to make it actionable. So information that is actionable to me might not be actionable to you. We all bring something to the knowledge party. This introduces an uncomfortable ambiguity about knowledge. What is knowledge to me might not be knowledge to you. Knowledge is not an absolute!

> For example: the long-range weather forecast for an office worker in San Francisco is interesting information. That same weather forecast for a vineyard manager in Napa is actionable; the vineyard manager will make decisions and take actions to maximize the yield and quality of his harvest. What is knowledge to some is only information to others.

This means that what we have in our "knowledge base" is really only potential knowledge because the usefulness of that information depends on the context, experience and need of the person looking at it. Information becomes knowledge in the moment of use. It is no wonder people pause when asked to define knowledge; it is rather abstract.

We all have some knowledge: the ability to take action on information. It is interesting to consider a few questions about

the knowledge that we have in our heads:

- How did we get the knowledge we have?
- When do we stop learning?
- How confident are we in our knowledge - are we ever one hundred percent confident in what we "know"?
- How do we gain confidence in what we know?

We gain knowledge through interaction and experience. Most of us would agree that we are never absolutely certain about our knowledge because in fact we never stop learning. We are constantly gaining new perspectives and enhancing what we know. And we gain confidence in what we know by trying it, the same way we gain it initially, through experience. We do not systematically get a subject matter expert to review our knowledge and tell us what is good and what isn't.

When considering the attributes of knowledge we could say knowledge is:

- Gained through interaction and experience
- Constantly changing (we never stop learning)
- Never 100% complete or 100% accurate
- Validated through use, experience and interaction (not by subject matter experts)

Is this what people in our organization expect when we say we are implementing a knowledge base or a knowledge management practice? Do they expect it to be created as a result of interaction and experience, constantly changing, never complete, not absolutely accurate, and validated through use? Usually not! Unfortunately, people's expectation of a knowledge base or a knowledge management system is perfect, pristine knowledge approved by experts. We have to change people's expectations if we really want to capitalize on the collective experience of everyone interacting with the knowledge.

For most, the adoption of KCS represents a major shift in thinking. It requires a shift in the culture (values and focus) from:

Individual	to	Team
Activity	to	Value Creation
Completion	to	Evolution
Escalation	to	Collaboration
Content	to	Context
Knowing	to	Learning and Sharing

KCS is a Journey, Not a Destination

KCS is a journey, not a destination. While the work of maintaining a KCS implementation is never done, we hope that the *KCS v6 Practices Guide* will provide some direction. This guide is a compilation of the proven practices of KCS from the problem solving and individual daily workflows, to content quality management, to insights for team leadership and performance assessment. We break out the eight practices of the KCS methodology into two reinforcing loops:

- Solve Loop
  - Capture
  - Structure
  - Reuse
  - Improve
- Evolve Loop
  - Process Integration
  - Content Health
  - Performance Assessment
  - Leadership & Communication

The concept of double loop processes, as opposed to linear processes, is taken from research in the complex adaptive systems area. We will describe more about the double loop concept in the next section.

After an overview of the double loop process, we provide a description of each practice area. We discuss techniques, concepts, and vocabulary and in many cases cover implementation variations and lessons learned. We emphasize the practical experience captured through years of work with support teams around the world. The experience across the membership also reflects a variety of support environments:

- Internal as well as customer facing support organizations
- Low volume, high complexity as well as high volume, low complexity
- Software, hardware, and network environments
- Enterprise, small to medium business and consumer customers

We should note that KCS applies to any information or knowledge intensive business, not just technical support. A number of members are adopting KCS across their entire company. They are implementing the KCS Practices in HR, marketing, sales, product management and development organizations.

The benefits realized in the short term can be assessed using traditional support metrics. The longer-term benefits are in new areas of value creation and, therefore, require new measures.

To fully address the organizational benefits, measures, and phases of adoption, the Consortium has written the KCS v6 Adoption and Transformation Guide. The phases of adoption are briefly introduced here and further defined in the Adoption Guide. Phases of adoption are referenced from time to time in the Practices Guide where the practice or technique differs based on the organization's adoption phase. *(Note: names of these phases were updates in October 2022.)*

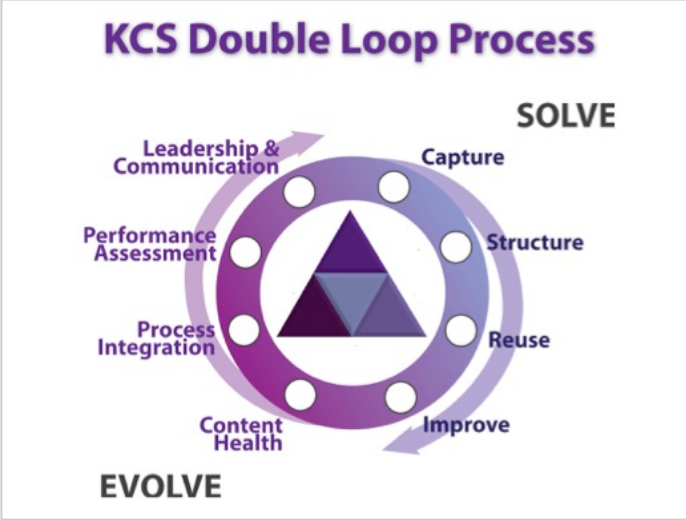
Phase	Focus
Plan and Design	Build tools required for successful adoption
	Gather baseline measurements
	Set realistic internal and external expectations
Adopt in Waves	Create internal understanding and excitement through initial competency
	Establish internal referenceability
Build Proficiency	Create and mature the knowledge base
	Increase process efficiency
	Reduce knowledge worker time to proficiency
	Improve collaboration and knowledge worker satisfaction
Optimize and Innovate	Optimize resource utilization
	Reduce support cost
	Increase customer success
	Improve employee satisfaction
	Improve products and services

Section 2 The KCS Practices

Content is King! Knowledge is at the Heart of KCS

The KCS methodology is organized into three components:

- Knowledge: KCS articles, the output
- The Solve Loop, the reactive process
- The Evolve Loop, the reflective and continuous improvement process



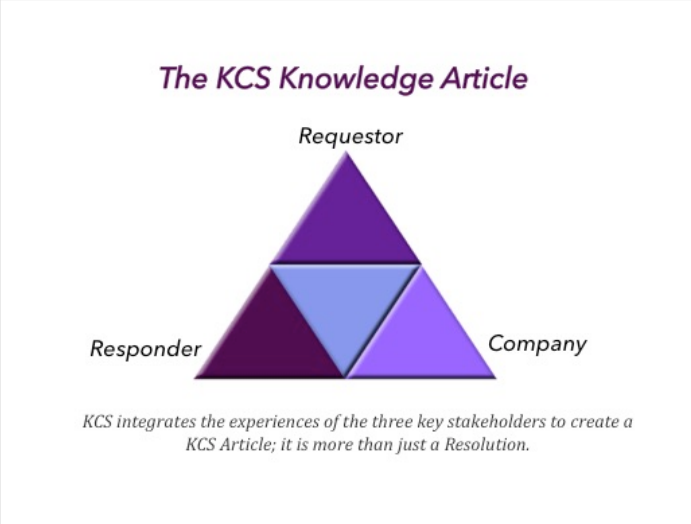
Knowledge is at the heart of KCS. It must be timely, findable, and usable by the audience being served. The Practices and techniques in the Solve Loop focus on reusing, improving, and if it doesn't exist, creating knowledge articles as a by-product of responding to requests: knowledge work. The Practices and techniques in Evolve Loop focus on learning from a collection of Solve Loop events and based on the patterns of knowledge reuse in the Solve Loop.

Knowledge is the by-product of interaction. To make knowledge relevant and findable, it is captured and structured during the request-response interaction and in the context of the requestor. In KCS, we want the knowledge workers to use the knowledge base as the first resource in seeking a resolution to an issue. The act of searching captures the request and the environment in which the issue occurs. A KCS article captures the requestors issue (in their own words) and the environment as well as the resolution from the responder. The article can also capture the responder's experience of resolving the issue. Once captured, KCS articles are improved over time based on demand and usage with the concept "reuse is review." This knowledge maintenance life cycle continues indefinitely as the knowledge evolves.

The KCS Article

The KCS article is the content, or knowledge, created by using the KCS methodology. Articles can be used for many different types of content including a simple question, a complex problem, or a procedure. KCS articles have a little bit of structure, and they integrate the perspective of three groups: requestor, responder, and the organization. The requestor's perspective includes their experience of what is happening as well as information about the environment relevant to their situation.

The issue statement is the requestor's view of what is happening and it is captured in their context (words and phrases). The responder's perspective is captured in the resolution and cause (if appropriate). The organizational perspective is represented in metadata with information such as the article state, the date created, number of times the article has been used, modification history, and the date last modified.



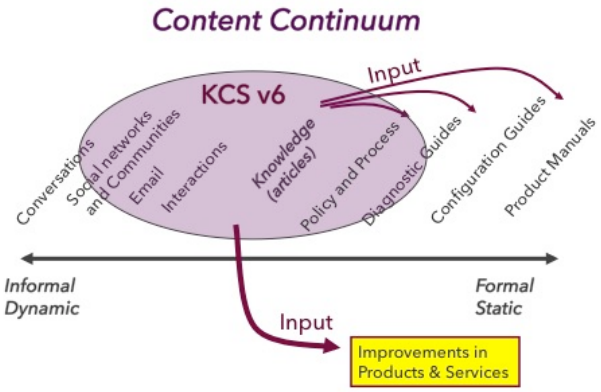
KCS integrates the experiences of the three key stakeholders to create a KCS Article; it is more than just a Resolution.

The KCS article has a structure or format that is defined in the content standard and is intended to cover a wide range of issues including:

- "How-to" or Q&A
- Interoperability issues
- Configuration issues
- Defects
- Diagnostic procedures
- Procedural documentation

The KCS methodology is leveraged by organizations to implement and enhance issue resolution and processes.

Each organization has a broad spectrum of content that contains valuable, reusable information. Historically, technical content like manuals and design documents has been very structured, often following rigid templates, and static—often only altered during product or service updates. However, as collaboration has become more real-time, valuable information is being shared in dynamic forms like instant messaging, email, and telephone conversations.

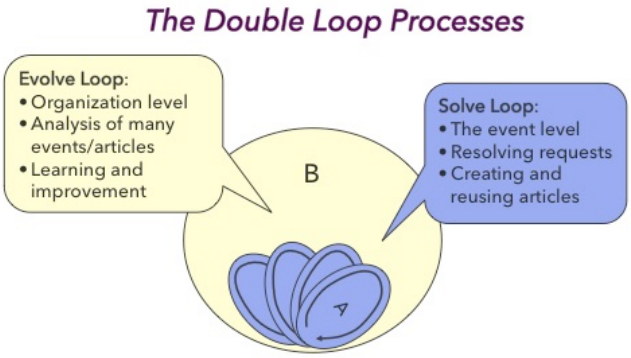


KCS v6 deals with multiple types of content: crowd-sourced, interaction, (JIT) and policy/process (compliance)

KCS articles (just-in-time content) are typically somewhere in the middle of this spectrum. KCS provides a methodology and set of practices for consistently capturing information in a way that is both structured enough to be useful, and dynamic enough to suit the rapidly changing environment. In addition to drawing from many content sources and creating a context-sensitive knowledge base for daily use, KCS processes generate new material that may reference existing content and feed into other business content like product designs, marketing plans, field training, and documentation.

The Double Loop Process

To optimize the health of the knowledge base and the capability of the organization, the KCS methodology uses double loop processes that reinforce each other. These Solve and Evolve Loop processes are the operational activities that make up the system.



KCS health relies on event based (Solve Loop) and organizational level (Evolve Loop) processes.

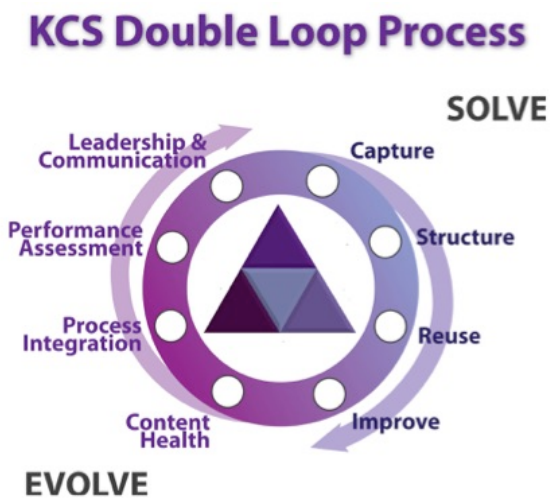
The Solve Loop (A Loop) represents the responsibilities of the responder when they are resolving a requestor's issue. The practices within this loop are reactive and transactional.

The Evolve Loop (B Loop) represents the responsibilities of leadership and the organization-level process. The Evolve Loop defines the Solve Loop processes and standards for content. It is also reflective in that it is a process of continuous improvement based on the analysis of the Solve Loop experiences. The Evolve Loop looks at a collection of Solve Loop events and the articles associated with those events. The Evolve Loop integrates knowledge worker activities in resolving issues with organizational-level processes, including the responsibility to support and continuously improve the overall maturity of the KCS system and improve the Solve Loop.

The Solve Loop

In the Create Value Principle, we say "Work tasks; think big picture." The "work tasks" is the Solve Loop and the "big picture" is the Evolve Loop. We want to focus on completing tasks in the Solve Loop with an understanding of the potential benefits the collection of tasks provides in the Evolve Loop. The Evolve Loop reflects on and learns from a collection of Solve Loop tasks and associated knowledge articles. It is the organizational-level processes which analyze the patterns that emerge for a collection of tasks and knowledge use. It only works if each task is done correctly. The Solve Loop and Evolve Loop are interdependent: each enables the other.

The Solve Loop is the request-response workflow. Responders capture their resolution experiences to create a collective memory. A little bit of structure in the request-response process helps enormously, while too much structure is disruptive. Use of the knowledge base and a structured workflow increases responders speed and accuracy in solving a request. Access to the collective experience through the knowledge base also reduces re-work, i.e. not spending time solving requests that have already been solved.



Knowledge (articles) and the knowledge base are the output of the double loop KCS process.

The Solve Loop and Evolve Loop together create a system that is self-correcting. The loops are co-dependant. The Evolve Loop depends on the people doing the right thing in the Solve Loop. It is the aggregate of lots of events, each handled correctly that enables the Evolve Loop analysis. By analyzing the collection of events and related knowledge articles over time the Evolve Loop identifies areas for improvement in the Solve Loop. And, perhaps most importantly the Evolve Loop identifies opportunities for improvement to the business. The root cause analysis done in the Evolve

Loop can drive improvements to products, services, processes, and policies that are based on the collective experiences of the knowledge worker and of those we serve.

Knowledge

There are four KCS Practices that contribute to the creation and maintenance of knowledge in the Solve Loop:

- **Capture** - capture in the workflow: while responding to the request, we capture the requestor's context (their words and phrases) as well as the responder's knowledge. As tacit knowledge becomes explicit in the context of use, it becomes part of the article. (Tacit knowledge is stuff we know but don't know we know, until someone asks.) Capturing this tacit information makes the article findable and reuseable.

- **Structure** - structure for reuse: consistent structure, simple templates, and a crisp style improves KCS article readability. We can identify existing KCS articles and their relevant elements quickly to reduce the solving cycle and ensure that new KCS articles build on and integrate with existing knowledge.

- **Reuse** - the words and phrases entered to search should be preserved. They are valuable content that can be used and reused to improve existing KCS articles. Or, in the event a KCS article does not exist, the phrases used to search become the beginning of a new KCS article. Searching the knowledge base is part of the request-response process. "Search early, search often" ensures that we are not solving a request that has already been solved and that we benefit from the collective experience of the network. Searching early and often also reduces duplicate knowledge articles.

- **Improve** - reuse is review: as responders, we take responsibility for the articles we interact with. The knowledge base is our collective experience. If we see something that is wrong or that we do not understand, we have the responsibility to "flag it or fix it." If we are confident in the correction and we are authorized with a KCS license, we should fix it. If we are not confident or licensed, we should flag it. If this sense of ownership is developed, reuse becomes review. We constantly review and improve the KCS articles that are being used. Allowing licensed users to edit articles reduces duplicates, keeps articles up to date, and assures high quality and value.

A Shift in Values

For most organizations, the Solve Loop practices are not natural. Our traditional performance measurement practices have not promoted collaborative behaviors—the assessment of contribution has been focused on the individual. In many cases, individual assessment promotes competition, not collaboration. Stack ranking of employees, for example, encourages a competitive attitude that does not promote sharing, improving, or reusing the collective experience captured in the knowledge base.

Many organizations create a hero mentality when they reward the "dragon slayers" or those who time the "diving catch" just right. These organizations communicate a clear value proposition: you are valued for what you know. If the organization values people for what they know, there is no hope for KCS success. KCS success requires the organization to shift the value proposition to the individual to: you are valued for your ability to learn, and your ability to help others learn. If this becomes the value proposition, then we are encouraging the behaviors of collaboration, sharing, and improving. This value shift requires a new performance assessment model outlined in the

_____

section.

The good news: most people do collaborate. Fortunately, they do the right thing in spite of the traditional organizational structures, measures, and linear processes. With KCS, we can begin to evaluate the value of collaboration through the sharing of knowledge. The knowledge base becomes a collaboration space.

Putting KCS to Work

In the next sections we will walk through the eight practices of the Solve and Evolve Loops in detail.

Practice 1: Capture

Overview

Our goal is to capture the experience of resolving issues. We want to capture the requestor's perception of what is happening, the environment in which it is happening, and the resolution to the issue. Depending on the nature of the work, the cause of the issue may also be captured.

Techniques

- Technique 1.1: Capture Knowledge in the Moment
- Technique 1.2: Capture the Requestor's Context
- Technique 1.3: Searching is Creating

Technique 1.1: Capture Knowledge in the Moment

A major step toward KCS adoption embraces the distinctive, real-time way knowledge is created and shared. Rather than being documented after the fact, KCS articles are created as part of the interaction and immediately made visible to others. The capture process is integral to the Solve Loop: from the first description of the request to the final resolution, the article is built along the way. Even if a resolution is not yet known, the KCS article is made visible to others, especially others working in the same product area. In this way, knowledge becomes a by-product of interaction instead of an onerous and time-consuming additional step. If we are creating articles after the request is resolved, we have missed vital information - and we are not doing KCS!

The integration of the knowledge base into the request-response process (search early, search often) greatly reduces the likelihood that we are working on issues that are already resolved or are in the process of being resolved. Searching is a form of capture.

Context is as important as content!

The just-in-time theme in KCS is one of the things that differentiate it from a knowledge engineering approach. A knowledge-engineering process is characterized by subject matter experts (or technical writers) who write articles disconnected from the request-resolve process. This kind of an "after the fact" process is expensive, slow, and lacks the critical element of "context of use." The only way we have found to effectively capture context is in the moment of use; context cannot be recreated after the fact. Once we know the answer, it is hard to ask the question in a way that reflects not knowing the answer. Context is as important as content.

A research project on diagnostic skills was conducted with a medical doctor and an auto mechanic, both of whom were known to be very good at diagnosing issues in their respective fields. They were asked to write down their approach to diagnosing a problem. Then, the research team observed each of them diagnose issues in a real-life environment. Neither of them followed the procedure they wrote down. They both took cues from the situation and tapped into tacit knowledge to improvise a diagnostic process. They could not access that tacit knowledge used in a real situation without the context of that situation.

Problem solving is a creative process. It relies on experience, instinct, context, and the successful processing of multiple variables or inputs at once. This implicit information and our ability to link it to tacit knowledge is difficult to explain and it cannot be accessed or extracted if we try to recreate it in the absence of demand.

Tacit information becomes explicit in conversations, often in response to a question or in the context of the need for information. Therefore, when resolving a request, we need to capture the information in the moment. We need to ensure that we capture the requestor's context - their perception of the issue - as well as tacit knowledge as it becomes explicit in the process of resolving the issue. This links what the responder knows with the requestors experience and creates a relevant and complete (findable and usable) picture of the issue.

Clarifying questions help to draw out and validate details that improve the success of interactions. Searching the knowledge base can reveal similar situations and prompt clarifying questions that can validate or eliminate known issues. Finding similar issues helps us remember what we know.

By capturing context-rich information from the beginning, the whole interaction process and article creation process becomes easier and more effective. When another requestor reports a similar issue, we will be better able to relate to the requestor's experience and find relevant KCS articles quickly.

Technique 1.2: Capture the Requestor's Context

With KCS, we seek to understand the requestor's experience *before* resolving. This understanding requires information about both the perceived issue (in the requestor's words) and the environment in which the issue is occurring. By capturing the requestor's perception of the situation in the first interaction, we dramatically improve the findability and relevance of the KCS article for other requestors who might have the same or similar perception of this issues in the future (context is as important as content).

Capturing the requestor's context - their words and phrases - improves findability for other requestors.

The objective of capturing KCS article elements is to "frame" the situation—to capture need, perception, experience, and relevant aspects of their environment as input to the resolution process. We use this information to either improve an existing article by reflecting another requestor's experience, or to create a new KCS article if one doesn't already exist.

Even if the requestor's perception of the situation proves to be wrong, capturing it will improve the findability of this article for others. The responder's job is to capture with precision and accuracy the relevant environment factors, the resolution, and the cause in the article, *not to correct* the requestor's description of the problem. However, responders may capture additional details to more precisely characterize the issue. Technical accuracy in the problem statements is not required. Of course, technical accuracy is critical in the environment, resolution, and cause statements.

Many of today's knowledge tools enable the capture of the requestor's online search activity before they ask for help. Having the words and phrases the requestor used to search is very useful - this is their context. The requestor's search terms can be used to modify an existing article and improve its findability, or to create the problem statements in a new KCS article, if one doesn't already exist.

Every Search Has Value

In the case of requests submitted via a self-service model, good things can happen if we capture the self-service activities (search strings, articles viewed, and in some situations product and version information). First of all, this guarantees that we are capturing the customer context, and secondly, making this information available to the responder helps the requestor feel like the effort spent pursuing a resolution through self-service was not a waste of time. This information can help expedite the process as we can review what has already been done. We call this "no dead-ends" for the user, and is a critical factor in creating a positive experience with self-service.

Requestor submission of issues via self-service also helps responders improve existing articles or create new ones. If responders need to create a new article for this issue, the search words and phrases the requestor used are valuable

content for the new article. If, as the KCS Practices suggest, we have captured the information the requestor used for their self-service search in a Work in Progress article as we were working the issue, we have already created a new article and simply need to review the environment statements and update the resolution field. Because the problem statements came from the requestor's search activity they are sure to be in the requestor's context.

If, in working on the issue, we eventually find an article in the knowledge base, we should improve that article so others can find it. We have already captured the information in the Work in Progress article, which we can use to update and improve the findability of the existing article. If the existing article was available to the requestor but not findable, we use the requestor experience (and their context) to improve the findability of that article. If the existing article was not available to the requestor, we change the confidence and audience metadata indicators if we are licensed to do so. Reuse drives both improvement of the articles being used as well as increasing the visibility of article being used.

Technique 1.3: Searching is Creating

Capture Relevant Content

While we want the requestor's context in the article, we don't want redundant, long, non-relevant information about the issue. The goal is to capture the information that will make the article findable and usable by others. Relevance of information is one of the many areas of KCS that requires judgment on the part of the responder.

The following are some guidelines for content relevance:

- Words and phrases the requestor uses to describe the issue (even if technically inaccurate)
- Environment statements relevant or unique to the issue
- Environment statements should be true before and after the issue is solved
- Information that distinguishes this article from other articles with similar symptoms but a different resolution (distinguishing characteristics are most often environment statements)
- Diagnostic process used in resolving the issue (details or how to do complex, reusable diagnostic processes are often articles themselves and should be referenced or linked)
- Resolution statements which completely resolve the issue described by the requestor

Capturing and refining content in the article as we work the issue is critical. We start by being very literal and seek to understand before we seek to solve. This may lead to capturing information in the article that, once we have solved the issue, we find is not relevant to the situation. This is typically true of the environment statements (functions, products, version and platforms). Prior to changing the article state from Work in Progress (WIP) to Not Validated or Validated, we should do a quick check for content relevance.

It's okay if the article does not contain every situation in which the issue might appear. The article should address what we know right now and it will evolve as it gets reused and updated by others. It is important to share what we know as we know it. If we spend time covering all situations that might happen, the article will never get published.

Environments where requestors use self-service as the entry point for assistance offer a great opportunity to capture the requestor's context. Capturing the words and phrases they used to search, or their click stream as they browse, as well as articles they opened can be very helpful in the event the requestor moves from self-service to the assisted path. If the requestor does a "click to chat" or "click to submit," the history of their session is passed to the responder. This kind of mechanism ensures that we capture the requestors context. It also makes for a positive experience for the requestor as they move from the self-service path to the assisted path.

See Appendix E for a sample of a quick reference guide for article content and structure.

Search Words are Candidate Knowledge

The words and phrases we use to search are candidate content for improving existing KCS articles or creating new KCS articles. They are especially valuable because they capture the requestor's context. Content used for searching should be saved, updated based on search results, and should become the beginning of a new KCS article in the event a KCS article does not already exist.

Content like the issue and environment statements frame the KCS article. This Work-in-Progress (WIP) KCS article should be saved in the knowledge base even if we don't yet have a resolution. We may continue to work on the issue or submit to the appropriate people for resolution (this process varies based on role and KCS license). A WIP article in the knowledge base lets others know that the issue has been reported. When the resolution is determined, we add it to the WIP article: we finish the article. We consider this KCS article complete and mark it with the appropriate state based on our KCS license (rights and privileges) and our confidence in the resolution. In the event that other open cases have been linked to this WIP article while we were working on it, they can be quickly resolved.

The process of framing and finishing KCS articles draws people into using the knowledge base as the basis for resolving issues. This, in turn, ensures that the collective experience is being captured in the process of resolving issues.

Practice 2: Structure

Overview

KCS proposes a simple structure or format for knowledge articles. This serves the goal of creating articles that are findable and usable. A simple structure improves the readability of the article and provides context for the content. It can contribute to the relevance of search results. And, a simple structure helps immensely in environments dealing with multiple languages for requesters and/or responders.

Techniques

- Technique 2.1: Use A Simple Template
- Technique 2.2: Complete Thoughts Not Complete Sentences

Technique 2.1: Use A Simple Template

KCS prescribes a specific structure or format, which provides context for the content, improves the readability of the KCS article, and promotes consistency.

Any situation or issue can be broken down into the following categories:

- **Issue** (sometimes called symptom, question, or problem)—the situation in the requestor's words. What are they trying to do or what is not working? Everything in the issue statement is resolved when the resolution is applied.
- **Environment**—what function, process, products, platforms, geography, categories, or topics does the requestor have an issue with? How is it configured? Has anything in the environment changed recently? These are the necessary environmental factors for experiencing the issue. The environment statements will be true before and after the issue is resolved. It is the richness of the environment statements that help us find the specific article we need. It enables us to find the correct resolution when the issues may be very similar but the resolution is different. Issue statements can often be vague because they are the customer's perception of what is happening. Environment statements should be precise. Environment statements are one of the key enablers in findability.
- **Resolution** (sometime called the fix or answer)—the answer to the question or the steps required to resolve the issue.
- **Cause**—the underlying source of the issue. (optional, typically only valuable for complex problems or defects)
- **Metadata**—attributes or information about the article. For example the article's audience, confidence, governance, date created, reuse count, modification history, and the date last modified.

By capturing the information in this structure at the start of a request, we are creating as we go. This is also the information we should be using to search the knowledge base for known articles. We reduce issue resolution time and ensure that new KCS articles build on and integrate with existing knowledge.

A little bit of structure helps readablity.

Ideally, as we work on the issue, we are capturing information in the correct area of the KCS article. This should replace the way we take notes today (on paper or electronically). Most of us capture key points while we are talking to the requestor, especially if we have a sense that this is a new issue. We want to take notes in a Work in Progress article for a few reasons. First, if an article about this issue doesn't exist we are creating it as part of the process. Second, we are capturing our notes in a readable, standard structure. And third, if someone else is working on the same or similar issue they are likely to find the Work in Progress (WIP) article; we can avoid redundant work and collaborate on solving the issue.

Once the issue is understood and the resolution is known, we review the content captured and refine the environment statements to be sure they are relevant. Relevant environment statements are critical as this is how we distinguish this

article from another with similar symptoms but a different resolution and cause. If appropriate, we update the cause field.

The most important benefit of this simple structure is it improves readability and usability. Also, some search technologies can take advantage of structured content to improve the relevance of their search results.

For self-service, the value of the KCS article may be improved by including links to specific sections of other documentation relevant to the issue or the resolution. Links to supporting information can help us write articles to the expertise level of the general audience. Those who are novices can follow the links for more information, while those who are knowledgeable in the area will be able to use the article without wading through documentation on things they already know.

We discuss KCS article structure and KCS article quality in more detail in the Content Health section.

For this technique, the key point is that consistent, simple structures help with readability and findability, and simplify the process of creating new articles or modifying existing articles. As we start the KCS journey, we want to keep the structure as simple as possible and let our experience drive improvements to the structure. Similarly we do not want to start with lots of different templates for different types of articles. Start with one, keep it simple, and let our experience drive the need for additional templates.

Technique 2.2: Complete Thoughts Not Complete Sentences

KCS proposes that we capture in the moment, using the requestor's words, but not all of them. We must be able to capture at or near the point of interaction. KCS advocates increasing the "capture-ability" and readability of content by making it short and succinct: complete thoughts or short phrases opposed to complete sentences.

We are not asking resolvers to become technical writers. As previously mentioned, many resolvers take notes as they are working on a request. KCS proposes those notes be captured as a Work in Progress (WIP) article in the knowledge base instead of in a notebook or personal database. For most responders, KCS doesn't propose additional work; it proposes a different way to do what they are already doing. The advantage is that the experience is captured in a structured and sharable way.

A list of single, complete thoughts improves article readability, especially for those users whose native language is not the language in which the article is written. (see the Content Health section for more details).

Practice 3: Reuse

Overview

Eliminating rework is one of the key factors that contributes to the operational efficiency of KCS. If people use the knowledge base as the first resource in seeking a resolution to an issue, they are likely to find that the issue has already been resolved, and thus avoid the work of researching or seeking assistance from others to develop a resolution.

Techniques

- Technique 3.1: Search Early Search Often
- Technique 3.2: Seek to Understand What We Collectively Know
- Technique 3.3: Linking

Technique 3.1: Search Early Search Often

We seek to understand before we seek to solve.

Searching the knowledge base must become an integral part of all knowledge worker processes—search early, search often. The best practice for information capture is to search the knowledge base in real-time. The words and phrases we use to search are potential content to improve an existing article or to create a new article, if one doesn't already exist. Once we have captured a description of the issue and some information about the environment, we have enough context about the issue to search. By listening and searching *early* based on the requestor's description of the issue, we ensure we understand the issue as the requestor sees it and we minimize the risk of investing time in problem analysis and research on an issue that has already been solved.

Searching *often* is important because as we are working on the issue and learning more about the situation, we need to search using the new information to see what we collectively know about this or similar issues.

Searching is not a one-time event but rather something that is done throughout the request-resolution process. The advantages of searching often include the following.

- As new information is collected, a search should be done to see if a knowledge article about this issue or a similar issue exists.
- The articles found in a search, even if they don't directly address the issue at hand, can provide helpful perspectives from similar issues. This can provide ideas on how others have solved similar issues and help us identify clarifying questions.
- It is particularly important to search the knowledge base one more time, before we save a new article, to be sure one doesn't already exist.

Technique 3.2: Seek to Understand What We Collectively Know

Searching during the request-resolution process helps ensure we do not spend time and effort resolving issues that have already been resolved. It also helps us resolve new issues by providing access to the broader experience of work on similar problems.

We may even discover someone else is actively working on the same or a related issue. This is the value of having Work In Progress articles visible to all who work in a given domain. At an organizational level, this technique is a major way we benefit from the collective experience that is captured in the knowledge base.

Search early and often ... but not forever. We should stop searching when the search has been refined a few times, the description of the issues is complete, and we have collected a few characteristics about the environment that are believed to be relevant. If at this point the search response is not providing anything of value, we should move to the diagnosis phase of issue resolution.

Correct searching techniques and making the judgment on when to stop searching is one of the skills Coaches focus on. While it sounds funny that we have to teach people how to search, it is a skill that knowledge workers must develop in the context of the environment in which they are working. Google has ruined us in terms of searching skills. The average number of words entered in a Google search is 1.5. But the Google search algorithms have the advantage of millions of searches a minute. No organization we know of has even a fraction of the traffic that Google has. In KCS, the more context we give the search engine, the better the search results. This is why we have to teach people how to search.

Technique 3.3: Linking

The Importance of Linking

The ability to associate a system of record (email, case, incident, community thread, tweet) with the KCS article that resolves the issue is a critical element of the KCS methodology. The data generated by the association is necessary for many of the Evolve Loop analysis activities. For example, calculating the link rate for individuals and teams, or calculating reuse and enabling the new vs. known analysis are all based on the ability to associate events in the system of record with articles. The association of articles to events is most often done by linking the article to the event in the system of record. Hence the term linking has become part of the KCS vocabulary. Organizations that cannot associate the system of record with articles or link them are at a disadvantage.

Linking to an Existing KCS Article

If a KCS article with a relevant resolution is found, it should be linked to the event in system of record. The existing article should be updated (improved) with any additional symptoms or relevant environment information.

If a relevant article is not found in the knowledge base, a new article should be created. Ideally we already have a WIP article, which has the words and phrases we first used to search (searching is creating), or if the requestor started the process in self-service, the words and phrases *they* initially used to search. The WIP should also contain the notes we took during the problem-solving process. Now all we have to do is update the resolution field (and the cause field, if appropriate), review the environment statements for relevance, and put the WIP into an appropriate state based on our confidence in the article and our license level.

Referencing and Linking to Other Information Sources

No single content resource can hold all knowledge needed to solve all issues. A rule of thumb when creating a KCS article is to keep it to one page and insert hyperlinks to other KCS articles and/or specific sections of online documents (work instructions, policy documents, product manuals, diagnostic guides). Because KCS articles are written in the context of the requestor, the articles can act as a context-sensitive index to the other resources. This approach eliminates redundancy and the need to maintain information in several different places.

Use of links to reference documentation in KCS articles allows more experienced users to move quickly through the content and at the same time enables less experienced users to understand and implement a complex resolution.

Best practice when we have multiple data sources is to provide a unified search capability: a single search across multiple databases. Preferences can be set in most search tools to prioritize data sources, so that the current knowledge base and preferred secondary resources are searched first. It can also be helpful to allow the knowledge worker the option of selecting the sources they wish to search.

Linking to Non-KCS Content

As search engine technology has improved over the past few years, many organizations are now able to index and search multiple repositories or databases with a variety of different types of content. In this environment, linking to the non-knowledge base content that resolves the issue is legitimate if the following criteria are met:

- The information is captured in a maintained repository or database
- The specific resolution or answer (a sentence or a paragraph) is findable by the search engine
- It is accessible by the audience being served (could be internal or external users)
- The content is in the context of the audience that is looking for it (they can find it, using their words and phrases, and they can use and/or understand it

When these criteria are met, the links to other types of content satisfy the fundamental goal of KCS: create findable, usable knowledge for a specific audience. So in this situation, creating a KCS article would add little or no value. Links to other types of content that satisfy the above criteria should be counted in reuse counts, link rates, and the new vs. known analysis.

An important rule of thumb is to avoid duplication of content; whenever possible, link to content that already exists. When linking to non-KCS content from a KCS article the same criteria listed above applies. If all the criteria is not met, the content should be copied and placed in the article.

Managing Article Versions

Many of the knowledge management tools today offer article revision history. The version of the article allows us to see the article's content when it was provided to the requestor as a resolution to their issue. Because articles that are being used should be improved, the current content of the article may be different from what was delivered.

In environments that do have article versioning capability we have to have a way to preserve the content of the article at the time it was delivered. Here are few thoughts on how to do that.

Capture the resolution as provided to the customer as part of the incident

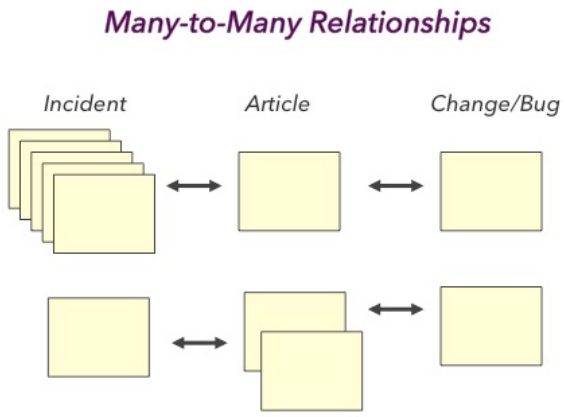
- The responder should be able to establish a link between incidents and articles. Links should be implemented in such a way as to be available to other processes - e.g., reporting, search-ranking algorithms, and search results display. (In other words, a hyperlink embedded in a text field as part of an incident note would not satisfy this requirement.)
- Responders should be able to link and unlink articles

Persistent link and snapshot

- The linking mechanism should allow a responder viewing the incident to see both the current state of a linked article(s) and the content of the article at the time it was delivered to the customer. (For example, by recording a snapshot of the solution at the time of delivery, or by including a link to a specific version of the article if the knowledge base supports version history.)

Many-to-many relationship between incidents and articles

- The linking mechanism should allow for multiple articles to be linked to an incident. Optionally, links may be associated with a link type, for example, "resolved" vs. "referenced."



Next up is the Improve practice, even though we just gave an example of how capture, search, reuse and improve all play together. We are beginning to see that the Solve Loop practices are tightly related. In talking about one practice or technique we cannot avoid discussing how it relates to the others.

Practice 4: Improve

Overview

In the process of finishing a KCS article (once the resolution is known), we apply our best judgment about what information to include in the article. We will often capture information that has no relevance to the issue based on understanding the resolution, and we should be sure to remove information (particularly in the environment statements) that ends up being irrelevant, misleading, inaccurate, or inappropriate to the audience. We want to capture relevant, accurate statements using our preferred vocabulary in the environment field. We want as much consistency in how we describe the environment as possible. And, we want as much diversity in the issue statements as requestors use to describe the issue.

Once again: the issue or symptom statements in the article need to reflect the requestor's experience; technical accuracy is not required. When we improve existing articles, we want to add to the issue statements - not rewrite, edit or delete the ones that are already part of the article. We have to assume that, no matter how incorrect or ridiculous they may appear to us, the statements represent the requestor's part of the article and reflect how they perceived the issue. This is what makes the articles findable by others who are likely to have a similar perception of the issue.

Techniques

- Technique 4.1: Reuse is Review
- Technique 4.2: Flag It or Fix It
- Technique 4.3: License to Modify

Technique 4.1: Reuse is Review

Who better to validate the accuracy of an article than the person who uses it?

We have discussed the capture process and the reuse process. Now we will look at how day-to-day use of the knowledge base *is* review, and how we constantly improve the quality of the KCS articles that have value. The goal is to create a culture of collective ownership. This is *our* knowledge base; it represents the best we know to date. And, in the moment of use, we are responsible for the quality of the articles with which we interact.

In most organizations, we know that up to 80% of KCS articles will rarely or never be reused. Of the remaining 20%, some will be reused a lot. The reuse pattern of articles always fits the 80/20 rule: 80% of the issues are solved by 20% of the knowledge base. The first question that comes to mind is: why create them all, if most are not going to be used? The problem is that we cannot predict the future value of an experience. Here is the logic for this position:

Reuse patterns of articles for self-service are different than they are internally. There are those issues for which requestors will use self-service and are happy to find an answer, but they would not bother to use the assisted model (open an incident) to get an answer. Data from our members on the activity in the assisted model vs the self-service model is a ratio of 1 to 10 (assisted to self-service activity). It seems that requestors will use a good self-service mechanism *ten times more often* than they will use the assisted model. This data is based on customer facing support organizations in high tech. For more information on what makes for a "good" self-service mechanism see Technique 5.10: Self-Service Success.

> Support organizations in high tech see less than 3% of the customer demand for support (see the Consortium's paper "A Demand Based View of Support" on the web site). 97% of support demand is served through self-service or online communities, forums, and social media. While we in support may have the best intentions, we don't have the context to make a judgment about the future value of what we learn from an interaction.

If the issue is worth solving, it is worth saving.

So, we capture all the knowledge worker experiences by linking, improving or, if it doesn't exist, creating articles. If the question is worth answering or the problem is worth solving, it is worth having in the knowledge base. Capture it and let the other KCS processes improve the quality of the articles that turn out to have value. If we implement KCS properly, the reuse, improve, or capture activity does not add to our handle time. And, the future value we get from reuse can dramatically increase our capacity as well as our speed and consistency in providing resolutions.

In the early phases of adoption, we let reuse draw our attention to the articles that have value. By embracing the "reuse is review" practice, we are constantly improving the articles that are being used. For those articles that are never

referenced, we have not wasted time reviewing them. This is an example of the fundamental KCS principle of a demand-driven system, and one of the reasons KCS is scalable and efficient. This demand-driven technique optimizes resource utilization and helps organizations avoid investment in dedicated quality assurance and editorial staff. Review during reuse also helps encourage timely availability of information and avoids costly and slow post-request knowledge engineering.

After we have Built Proficiency in the organization, we have enough KCS Publishers in the organization to do just-in-time publishing for articles we are confident about. To do this successfully, we must have a fast, closed loop mechanism for feedback; the requestors participate in the "reuse is review" process. Requestors are very quick to point out articles that they don't understand or that don't work. With a just-in-time publishing model, we have to be able to respond with corrective actions quickly.

Reviewing every KCS article that is created is a huge waste of time and money. Articles should be written in a way that is sufficient to resolve the issue. In the Solve Loop, this means as responders we are responsible for the quality of the KCS articles we interact with. We improve KCS articles as we reuse them to increase the KCS article quality with each interaction. In this way, we focus only on those articles that are being used.

The best people to create and review knowledge are the people who use it everyday.

Reuse is review reinforces this concept.

Technique 4.2: Flag It or Fix It

Within the culture of KCS, people take responsibility for what they see in the knowledge base; they follow the simple rule of "flag it or fix it." Licensed users can clean up minor problems in the moment, or add information that enriches and evolves the KCS article. KCS articles that are flagged need to trigger a workflow that will get the attention of a subject matter expert. These modifications, based on real usage (demand), lead to continuous, ongoing knowledge base improvement.

As we use the knowledge base, we are responsible for the quality and accuracy of the articles we interact with. If we see something we think is wrong or doesn't make sense, we need to take one of two actions: flag it or fix it. The flag it or fix it concept applies to both content standard violations and technical accuracy or completeness.

- Fix it: if we are confident and we have a license level to make the update we fix it.
- Flag it: if we are not confident or we don't have the license level to make the update we flag it.

Fix It vs Create a New Article?

When is a new KCS article justified? KCS article creation should occur when a unique resolution is required to address an issue within a specific environment and such an article does not exist in the knowledge base or in another searchable, maintained repository (see Technique 3.3: Linking). While the content standard should provide some guiding criteria, as with many things in the KCS methodology, this decision requires judgment.

Generally, there should be one KCS article per resolution and cause. Multiple resolutions for different environments and the same issue or symptom should not be in the same article. However, this is not an absolute rule, and the criteria should be developed based on experience in the environment. KCS articles will evolve through use and sometimes merge or split as additional experience is gained. Decisions should be based on what is best or clearest for the intended audience of the article: what will make the most sense to them.

A single KCS article may include different approaches to resolving an issue with a single cause. For example, the fix or resolution may include a number of options for resolution, such as offering a temporary, manual work-around to the issue or a formal fix or code update. The options offered should include a description of the effort and implications for the user of each option.

As we will see later, we augment Solve Loop "in the workflow" articles with Evolve Loop articles. Evolve Loop articles are high-value articles (based on reuse) or articles that describe diagnostic processes that can guide users through a number steps to get them to the best answer in an optimum way. Each step in the process is an article and the outcome of each step directs the user to the next step. These procedural articles are linked together to get them to the correct

article to solve the issue. This is very helpful in dealing with issues that have very generic symptoms and multiple possible causes. (See Content Health Technique 5.4: Creating Evolve Loop Articles.)

Even though a newly created KCS article, or Work in Progress (WIP), may not contain a resolution, it represents valuable knowledge. Work in Progress articles in the knowledge base enable others in the organization to discover that a problem is being worked. This process helps eliminate duplicate effort—two responders unknowingly solving the same problem in parallel. Awareness also enables collaboration.

Responders should not be expected or try to assess the future value of a KCS article. If the issue is worth solving, it is worth saving. Our goal is to create a knowledge base that reflects the collective experience of everyone interacting with the knowledge. The completeness of that experience then more accurately reflects, through patterns and trends, the requestor's and responder's experiences. If we selectively ignore issues by not capturing them, the patterns over time are less valuable.

When creating new articles we should not attempt to extend the article to cover all possible situations that might occur. Instead, the article should resolve the issue raised by the requestor. Then, if the article is reused, it should be modified or expanded based on demand. Over time, the problem statements in the article will describe the issue in as many ways as requestors have experienced them.

A certain level of redundancy and diversity in a knowledge practice is healthy. Redundancy becomes a problem only when it adversely affects the findability and usability of the content. Some examples of acceptable redundancy include:

- KCS articles for the same issue but for different target audiences. This can avoid confusion. Target audiences can be defined as an environment variable, thus requiring a separate issue with a different resolution.
- KCS articles that capture wholly different experiences but have the same resolution and cause. Initially these articles will not show in a single search. But if these KCS articles are being used and modified over time, their problem statements will eventually have them show up in a single search, at which point they should be merged (updating and keeping the oldest). Having two articles with different issues with the same resolution does not necessarily mean there is redundancy. You must also consider the cause. It is possible to have the same resolution for two complete different issues. If the cause is different, then the issues are most likely unique and therefore no redundancy exists. When you find two articles that have different issues and the same resolution, the advice is to evaluate the articles to see if they are two different descriptions of the same problem. Both may just have different symptoms. In this case there is redundancy and the article should be merged. You may also find two articles with similar descriptions and different resolutions. Upon evaluation the issues and environment are the same, the cause is the same, however the resolutions provided are different. This is also redundancy. In this case the duplicate articles should be merged.

Duplicate Articles

Duplicate articles are inevitable if the organization is truly practicing KCS. To some extent, duplicate articles are a necessary ingredient in a successful knowledge management practice. Duplicate articles become a problem when multiple articles with similar symptoms and the same resolution are showing up in response to a search.

There are two causes of duplicate articles. One is necessary and productive; the other is not.

The first is naturally dealt with in the KCS methodology. A person encountering an issue may describe it in a totally different way or in a different environment than the way in which an existing article in the knowledge base is documented (article A). The responder is not likely to find the existing article and will create a new one reflecting the requestor's described experience (article B). If the issue is one that people encounter often, others will search with a variety of symptoms and may find article A. They should update the symptoms to include the requestors experience if it is not already in the article. Other responders handling this issue may find article B and should update the problem description appropriately. If these articles are being used often, over time they will eventually both show up in a search. The responder who first sees them both should merge articles A and B. If we are following the Reuse is Review technique and constantly updating the articles based on the customer experience, duplicates will evolve over time to the point where they are close enough to both be found in a search. That is the point at which we should merge them.

The second cause of multiple articles with the same symptoms, environment, and resolution showing up in response to a search is a result of not following the KCS practices. Lots of duplicate articles are typically a symptom of one or a combination of the following common violations:

- Responders are given a goal for article creation; this drives the behavior of creating rather than re-using
- Responders are not following the "search early, search often" and/or the "search before you save" techniques and as a result they create articles about issues that have already been solved and captured in the knowledge base
- The culture discourages editing articles that are believed to "belong" to others, so responders create duplicate articles instead (individual ownership of articles is death to successful knowledge management practices)

In any case, we need a way to deal with duplicate articles.

Dealing with Duplicates: The Merge

When duplicate articles are discovered they should be merged. Different KM tools have different ways of dealing with this but the best practice, based on Consortium members' experience, proposes that the newer article (or articles) content and links be merged into the older article, and the newer articles are archived or deleted. Here are some of the key reasons to preserve the oldest article:

- It is important to keep the metadata: information like the date this issue first occurred, its revision history and other important article attributes and history
- We don't want to lose the links to the original incident and subsequent incidents
- We want the reuse count to be based on the complete history of this article
- It is typically less work; the older article is more likely to have a richer set of symptoms and environment statements

Technique 4.3: License to Modify

Not everybody gets to do everything. The KCS licensing model defines a knowledge worker's rights and privileges in the system. The licensing concept is discussed in the Performance Assessment Technique 7.1: Roles and Licensing Model. Knowledge workers who show an understanding of the KCS concepts, content standard, and workflow, and an ability to consistently make good judgments earn the right to do more in the system. While the licensing model varies from organization to organization, generally a knowledge worker with a KCS Publisher license means they can update the audience attribute of an article to make it visible externally.

Technology plays a critical role; the knowledge management tool must have the functionality to manage rights and privileges based on user profiles and license levels defined by the organization. For more information about the technology requirements, please see the KCS *Verified* documentation.

Closing Thoughts on the Solve Loop

Collective Ownership in the Solve Loop

One of the Core Concepts of KCS is collective ownership. The concept, while sometimes tricky, is not unique to KCS. For example, Switzerland is recognized as one of the cleanest countries in the world; it is pristine. How do they do that? Turns out it is a result of collective ownership. In Switzerland if you see trash, you pick it up. It's that simple. The Swiss do not have a bigger budget for clean up crews or more street cleaners than other countries. What they have is a value they have all bought into about a litter-free environment.

This is the same principle we see in the KCS Practices. The collective ownership concept is an Evolve Loop thing; at the systemic level we all own the knowledge base. It is our collective experience. At the Solve Loop, or event level, it means I am responsible for the quality of the articles I interact with. The following Solve Loop techniques all represent the collective ownership value at the behavior level:

- Create an article if one doesn't exist, modify it if it does
- Link
- Update article states as appropriate
- Flag it or fix it
- Reuse is review

Judgment is Required

A common myth about a knowledge management practice is the idea that a good knowledge base will allow the organization to deliver issue resolution using lower skilled, lower cost resources. In KCS this is not the case. We can see from the Solve Loop practices of reuse, improve, and create that we need people who consistently exercise good judgment. If, in the process of solving an issue, we find a KCS article we feel resolves the issue, we should review the article to ensure that it is relevant and accurate. We must exercise judgment, and we are responsible for the advice we give. If we are uncertain about a situation, we need to get a second opinion, just as we would before KCS. The knowledge base does not reduce the need for good judgment. In fact, the need for good judgment increases.

The Evolve Loop

Evolve Loop Overview

The Evolve Loop reflects on and learns from a collection of Solve Loop tasks and associated knowledge articles. It is the organizational-level processes which analyze the patterns that emerge for a collection of tasks and knowledge use. It only works if each task is done correctly. The Solve Loop and Evolve Loop are interdependent: each enables the other.

The Evolve Loop is the systemic view that provides:

- Definition of the higher level organizational practices and techniques
- Definition of the measures and indicators of health for each practice
- Continuous improvement processes for the:
  - Solve Loop and the infrastructure that supports the Solve Loop behaviors (functionality and tool integration)
  - Knowledge base through creation or refinement of high-value articles based on patterns of article reuse and self-service activity
  - Self-service experience and success
- Identification of improvement opportunities based on knowledge worker activity and knowledge reuse.
- Root cause analysis to identify high impact improvements to the products, documentation, business processes, and policies

The Solve Loop and Evolve Loop together create a system that is self-correcting. The Evolve Loop depends on people doing the right things in the Solve Loop. It is the aggregate of lots of events, each handled correctly, that enables the Evolve Loop analysis. By analyzing the collection of events and related knowledge articles over time, the Evolve Loop identifies areas for improvement in the Solve Loop. Perhaps most importantly, the Evolve Loop identifies opportunities for improvement to the business. The root cause analysis done in the Evolve Loop can drive improvements to products, services, processes and policies. These improvements are based on the collective experiences of the knowledge worker and of those we serve.

KCS Value and Health Indicators

Each of the Evolve Loop practices includes a set of indicators that together can be used to assess the value KCS is creating for the organization. These indicators can also be used to assess the health of the KCS Practices. The definition and use of these indicators will be discussed in more detail under each Practice section. Here is a summary of the key indicators by Practice:

- Content Health
  - Content Standard Checklist (formerly Article Quality Index or AQI) - adherence to the content standard
  - Assessing the value of articles - indicators of value: reuse, reference and value of the collection of articles
  - Search effectiveness: a combination of search technology, structure, context, and environment statements
- Process Integration
  - Process Adherence Review (PAR) - adherence to the Solve Loop workflow
  - Technology - Technology or Infrastructure functionality, integration, and response time
- Performance Assessment
  - Value Creation Indicators (VCI) - performance assessment for individuals and teams
- Leadership & Communication
  - Communication Effectiveness Indicators (CEI) - understanding and buy-in to KCS across the organization
  - Organizational Value Indicators - realization of the KCS benefits for the organization and delivering on the WIIFM (What's In It For Me) for each of the stakeholders

Content is King... and Leadership is Required

In KCS, content is king. We will start the Evolve Loop discussion with Content Health: what are we trying to create? Then we will discuss the Process Integration practices: how are we going to create good content? From there we move to a discussion on Performance Assessment: how do we measure how we are doing? Performance Assessment is about individual and team value creation. Last, but certainly not least, we will discuss the critical elements of Leadership & Communication: how do we get people, at all levels of the organization, to understand and buy in to KCS? It is leadership's responsibility to create an environment where people are willing and eager to change their behavior; it is the behavior change that KCS requires of both the knowledge worker and the leadership of the organization that creates the profound benefits.

The Evolve Loop Practices

- Content Health
- Process Integration
- Performance Assessment

- Leadership and Communication

Practice 5: Content Health

Overview

Capturing Value in the Interaction Process

Knowledge has value - if we capitalize on it! Our goal is to maximize the value of what we learn in the process of interaction. Following are some of the key content best practices that drive the value of what we learn to the whole interaction network.

- **Capture all the KCS article information during the interaction**. Rather than documenting articles at the end of an interaction, KCS recommends capturing the KCS article information during the discovery process. The situational elements should also be collected so that the KCS article represents the entire experience. This practice of considering all the elements such that they *can* be captured provides focus on the resolution process and creates operational efficiencies, even when resolving a new request.
- **Respect the contributions of all people interacting with the knowledge**. Responders are the primary creators of KCS articles through their interactions with requestors in resolving issues. However, KCS articles can be captured or modified whenever and wherever they occur. Requestors and responders are both in positions to capture new KCS articles, and their different perspectives on the environment and context of the situation can enrich the knowledge base. The combination of the KCS licensing model and the KCS article metadata attributes of audience, confidence, and governance provide a means to manage who can see an article and who has the privileges to modify or create vs simply flag and comment on an article.
- **Structure KCS articles for rapid reuse**. Consortium members have observed that the redundancy rate for issues is anywhere from 65% to 90% of the total incident volume. Reusing existing KCS articles improves the speed, accuracy, and consistency of the response in the assisted path and enables requestors to find resolutions on their own through self-service. It is always faster to use what is known than to research and create a new resolution.
- **Structure information for findability and readability**. A simple, consistent structure is one of the key things that improves findability. The same structure provides context for the content of the article and improves readability.
- **Collective ownership.** This is one of the Core Concepts of KCS**.** Article quality is the responsibility of everyone who interacts with the knowledge base**.** If knowledge workers feel a sense of ownership for the knowledge base and the article quality, it encourages the techniques of "reuse is review" and "flag it or fix it" in the Solve Loop. These techniques drive continuous improvement of the articles being used.
- **Evolve Loop articles**. Evolve Loop articles are created in the Knowledge Domain Analysis process, usually done by Knowledge Domain Experts (KDEs). Patterns and trends that emerge from the reuse of Solve Loop articles, those that are created as issues are being solved, are the basis for high-value Evolve Loop articles.
  These include articles that are improved by the KDEs because of a high rate of reuse. Based on the analysis of a collection of articles (usually thousands), a small number of KCS articles may be updated, merged, or improved. Analysis of the self-service experience may identify issues for which an article does not exist, in which case the

KDE creates one (or facilitates its creation) to satisfy that demand. And finally, we may identify a need for diagnostic or procedural articles. These are reference articles; they do not contain answers or fixes, but they provide direction on how to perform tasks that are helpful, or in some cases, necessary to analyze a situation. We call this type of high-value content Evolve Loop articles.

Implemented in everyday actions, these core ideas enable the organization to realize the full value of the knowledge base. The next sections provide specific techniques to the numerous aspects of creating a healthy KCS knowledge base.

Techniques

- Technique 5.1: KCS Article Structure
- Technique 5.2: KCS Article State
- Technique 5.3: Developing a Content Standard
- Technique 5.4: Creating Evolve Loop Articles
- Technique 5.5: Archiving Old Articles
- Technique 5.6: Dealing With Legacy Data
- Technique 5.7: Priming the Knowledge Base With New Information
- Technique 5.8: Global Support Considerations
- Technique 5.9: Knowledge Domain Analysis
- Technique 5.10: Content Health Indicators
- Technique 5.11: Self-Service Success
- Technique 5.12: Self-Service Measures

Technique 5.1: KCS Article Structure

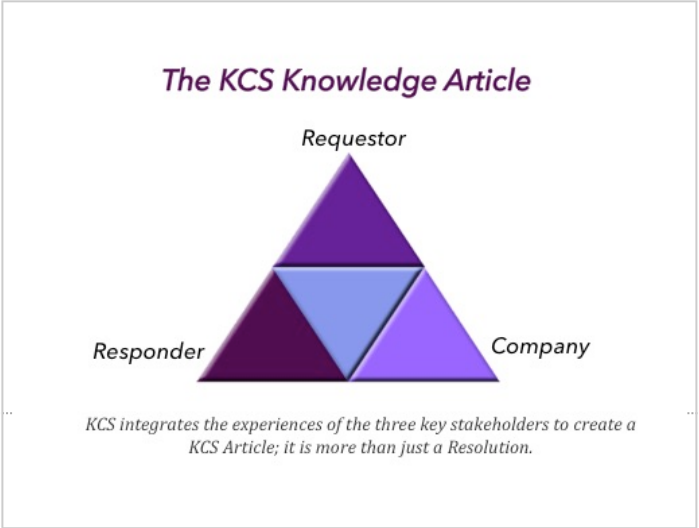
Overview

Content Health begins with the article structure. A well-defined, simple structure is a fundamental element of KCS. A consistent structure contributes to both findability and readability of articles. The goal of KCS is to capture the organization's collective experience, or knowledge, in the form of *articles*.

Articles capture what we have learned in responding to a request. The article content is the reusable part of the experience and should not include information that is specific to the requestor such as company names or contact information for people, entitlement, or specific locations. That information should be kept in the system of record for interaction. For support organizations, this event specific information is kept in the case or incident management system.

KCS articles are more than just the question and the answer. The article connects the requestor's context with the responders experience and resolution and information that is valuable to the organization.

KCS is a modular approach to knowledge. Ideally, KCS articles are a page or less in length. A given situation may use multiple articles to get to the resolution. KCS articles often contain links to other other articles or more reference information that already exists in other databases.



KCS integrates the experiences of the three key stakeholders to create a KCS Article; it is more than just a Resolution.

Establishing a Good Format or Template

The right structure ensures that KCS articles in the knowledge base are findable and usable *by the intended audience*. Identifying the intended audience is important because the audience defines the context for the KCS article. Ideally, the audience we are serving should be involved in creating and giving feedback on the articles. Unfortunately, not many organizations actually enable this.

KCS seeks to create content that is good enough to be findable and usable by a specific audience.

One of the key goals of KCS is to capture the *context* of the issue: the description of the needs and perspective of the requestor, in their own terms. To achieve both broad reuse and relevance, the reusable context for a given situation is contained within the KCS article, in its own section.

We have found that a simple, single structure works best. And, this same structure can serve many different needs including:

- simple Q&A
- technical issues (both simple and complex)
- how-to questions
- process instruction
- diagnostic procedures (both simple and complex).

These are the four basic, common elements or fields of a KCS article:

- **Issue** (symptom, problem, or question)—the issue is described in the requestor's words and phrases—what are they trying to do, what is not working, or what are they looking for? It is helpful to view this field as belonging to the requestor (even though it may be captured by the responder). It must represent the requestor's perspective and context.
- **Environment**—what product(s), category, or business process does the requestor have? Has anything been changed recently, such as upgrades, additions, deletions? The environment description should be as precise as possible, with standard ways to document product names, versions, or processes. The environment will remain the same after the issue is resolved.
- **Resolution**—the answer or the steps taken to resolve the issue.
- **Cause**—the underlying cause of the issue. Cause is an optional field as it is not appropriate or necessary for some types of articles. A simple Q&A, for example, doesn't need a cause. However, for complex technical issues, a cause can be very helpful in assisting the user in determining if an article is relevant to them.

The article also contains a collection of attributes that describe a variety of things about the article called **metadata**. Some of these attributes are added by the knowledge management system automatically, like dates, time stamps, versions, reuse counts, and the identity of the knowledge worker(s) that created or have modified the article. Other attributes are explicitly set by the knowledge worker as the article is created and used. This includes the audience, quality, and governance attributes.

Here again we must reiterate the "keep it simple" idea. Resist the temptation to over-engineer the the structure or the number of templates or the metadata fields. Make it as simple as possible and then try it. Evolve the structure, templates, and metadata based on the organization's experience. We have a tendency to want to make these things everything they *could* be in anticipation of the many ways we *might* use them. Don't do it! Use the principle of Demand Driven or, if you like, an Agile approach. Design it to be good enough to start and then plan to iterate on it: improve it based on experience.

Leaders should note that structuring KCS article content requires a change in behavior for the knowledge worker. There is a learning curve as the knowledge workers learn to capture and structure in their workflow. They have to learn to distinguish the event-specific content (the event itself) from the reusable content (what we learned from the event). Coaching is crucial at this stage as that is how we promote and create new habits. This represents an investment. However, as the Solve Loop practices become second nature and we capture our collective experience as

articles in the knowledge base, reuse quickly increases and create activity decreases. The time invested in coaching to get "over the learning curve" will be more than compensated for by the time saved in the improved request-resolution process.

The KCS article contains the reusable parts of the experience, not customer-specific or proprietary information.

Details on the Resolution Field

The resolution contains the answer to the question, a workaround, or a fix to the problem. If the resolution contains a multi-step procedure, it improves article readability if we number the steps.

Sometimes the resolution requires authorized access, special tools or skills that the user or audience may not have. If the audience for the article does not have the access or resources to complete the resolution, the resolution should provide instructions like "contact your support center for assistance in resolving this issue." The support center should have access to a restricted field in the article (which the user cannot see) that provides the steps to resolve the issue. It is a good idea to have an article that the user can find to indicate the issues known. Including guidance for obtaining service in the resolution field of the externally visible article can help the requestor contact the support center and provide relevant information to the responder to minimize diagnosis.

Adding an "Internal Resolution" field in the knowledge article provides a place to capture a resolution that requires assistance. The "Internal Resolution" field is not visible to requestors even on externally visible articles. This is a technology requirement for your knowledge management tool. If we don't have that capability, we can create a separate article that is flagged as "internal use only" and linked to the externally visible article.

Details on the Cause Field

As we mentioned above, the Cause field is optional as not all issues have a cause (or the cause may not be known). For example, "how-to" articles never have a cause, unless you'd like to point out that the requestor didn't read the instructions or manual.

If the cause of an issue is known, it should be added to the knowledge article. This can be used to distinguish between two knowledge articles with the same issue description which are actually two different problems. For example, an issue of "I can't print" may be due to the printer being out of paper, out of ink, paper jam, or a number of other potential causes, each requiring a different resolution. When searching the knowledge base and multiple articles are found with similar issues, the cause within each article can be used to verify which problem exists for this reported issue.

An additional strategy of value to consider is to add an additional field related to the Cause field, called "Cause Test." In the "Cause Test" field will be the procedure or description of how to validate the cause. The requestor or responder can then use this test to confirm that the issue they have matches the knowledge article and will then have confidence that the resolution will address the issue. For example, a cause of "out of paper" may include a cause test describing how to check the paper level in the printer.

Multimedia as Content

Throughout this document, we have talked mostly about KCS articles presented as text. However, for certain audiences and for certain types of knowledge, multimedia proves to be far more effective than text. Many of the Consortium members are including pictures or screen shots, animation, voice and short videos as knowledge articles or as resolution to an article. Visual images can bridge language gaps and overcome translation issues. Voice and audio clips are also increasingly common, both for ease of comprehension and for compliance with increasing regulatory requirements for accessibility. As more organizations pursue self-service, multimedia formats can be very beneficial in speeding resolution and improving the user's experience. But again, it depends on your audience. The nature of the knowledge should dictate when and where multimedia makes sense.

The KCS methodology and processes remain the same, but the knowledge base and delivery tools may need to be adjusted to accommodate multimedia content.

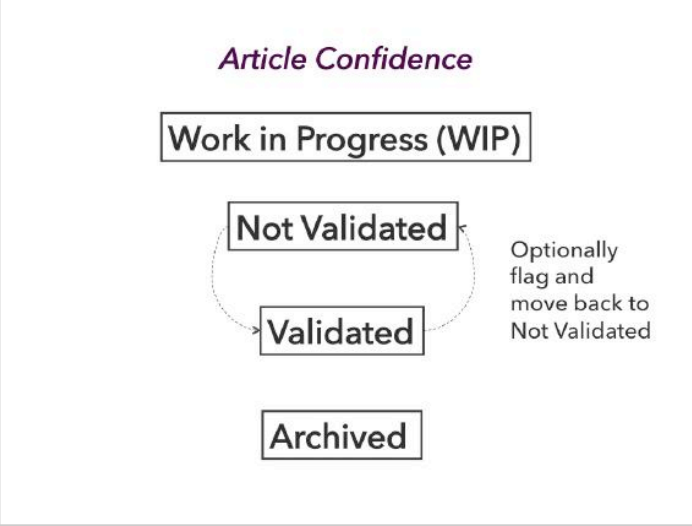
Technique 5.2: KCS Article State

Article state is comprised of three metadata fields: article confidence, article audience (formerly known as article visibility), and article governance. The *confidence* of an article tells us about the level of certainty we have in the article's structure and content. The article *audience* adds an additional layer of control that allows you to create business rules allowing different access to different audiences. The article *governance* is another level of segmentation by subject matter. The KCS Roles or licensing model indicates which users have demonstrated the competence to create quality articles and the judgment to make them visible to the appropriate audience.

Article Confidence

To manage the readiness of a KCS article, we use article confidence to indicate where it is in its life cycle. All articles, regardless of the confidence we have in them, are worth sharing in case someone else is dealing with the same new issue. As the article is resolved and then reused, we gain confidence in its quality. Knowledge is never complete; it continues to evolve as it is used. Within this evolution, the KCS article life cycle is non-linear—KCS articles may move through the various qualities in many different ways.

- WIP - Work in Progress
- Not Validated
- Validated
- Archived



WORK IN PROGRESS (WIP)— the article does not include a resolution; the problem or question and some information about the environment has been captured but the resolution is not yet known. This is sometimes referred to as a "framed" article.

- Having WIP articles in the knowledge base helps us avoid duplicate work. This is especially true in moderate to highly complex environments where the resolution of issues often takes days or weeks. WIPs inform other knowledge workers that the issue is being worked on if they encounter a similar request. This visibility to work in progress promotes collaboration across the organization.
- WIPs are temporary. Generally they should become either Not Validated, Validated, or they should be deleted. If the issue is never resolved and the request is closed because it was abandoned by the requestor, the

responder needs to decide if anything of value has been captured in the article. If the request is no longer being worked and there is nothing of value in the article, then it should be deleted.

- WIP helps responders manage pending work (as the name implies). WIPs give us an easy way to identify and manage incomplete articles in the KB.

NOT VALIDATED — the article is complete in that the article has a resolution, but we are not confident in the structure or content due to lack of feedback, others' use of the article, or because the article may not be structured or written in a way that complies with the content standard. For example, the article may have been created by a KCS Candidate who does not yet have the skills to validate an article.

- Not Validated gives us a way to capture the collective experience of the organization and distinguish between those articles we have confidence in (Validated) and those we don't (Not Validated). This allows knowledge workers to capture and share all their experiences.

- Confidence - The Not Validated designation supports the concept of "capture everything." If the issue is worth answering or solving for the requestor, it is worth having in the KB. It also supports the concept of "sufficient to solve." People are often hesitant to capture all of their experiences as articles in the KB if they are not confident in the resolution. Not Validated provides a way to indicate, "here is what I did in this situation, but I have not been able to validate the resolution or answer."

- Efficiency - Let reuse of Not Validated articles be the review and drive the article life cycle. Reviewing all Not Validated articles creates overhead and expense that produces very little value.

- The users of the KB who have visibility to Not Validated articles need to understand that Not Validated articles should not be used unless the article is relevant to the situation they are working on, and they have confidence in the resolution. Judgment is required.

- KCS Candidates (people learning the KCS Practices) can only create and modify Not Validated articles, which are then reviewed by a KCS Coach.

VALIDATED—The article is considered complete and reusable, and we are confident in it. The article is being worked by a licensed KCS user (KCS Contributor, Publisher, KDE or Coach) or by a KCS Candidate and it has been reviewed by their Coach. The criteria to designate an article as validated are:

1. Confidence - Responder confidence can be the result of confirmation from the user that the resolution worked, or the problem was recreated and the responder was able to validate the resolution, or, simply based on their experience, the responder is confident in the resolution.

2. Compliance with the content standard - The article is compliant with the content standard, written in the context of the requestor.

ARCHIVED—Archiving an article from the knowledge base is normally done only when the article is defined as having no value. Archiving an article is better than deleting it. If an article has been linked to a request, you will not want to delete the actual article from the knowledge base as it will result in a broken link between the knowledge base and the

system of record, incident, or ticketing. A common method to remove the article from a searchable knowledge base is to set it to Archived. The article is logically deleted from the knowledge base so that it will no longer be presented in a search results or browse function. It can still be viewed from a previously linked incident or by an advanced search function that includes archived knowledge articles.

The confidence, audience, and governance fields are stored as part of the KCS article metadata. As an article evolves and is used, reviewed, and enhanced, the article state is updated. The article confidence affects the trust users place in its accuracy. Article confidence is extremely important and frequently referenced by the users of the knowledge; therefore technology should make the article confidence visible to the users.

An article can move through from the Not Validated to Validated qualities several times throughout its life cycle. The process will vary depending on:

- The KCS maturity of the organization
- The usage of the KCS article
- The license level of the person interacting with the article

Due to this dynamic, we may want to keep a copy of an article as it existed when we delivered it to a specific requestor. If we are supporting a mission-critical product or process, there are often regulatory compliance requirements that mandate capturing the situation exactly, noting the KCS article recommended, and explaining why. The exact KCS article can be preserved by copying a snapshot into the incident or by having version control for articles in the knowledge base.

Confidence Transitions

Following are a few examples of quality transitions.

Work in Progress (WIP)

- A WIP is created at or near the point in time the first search is done on the issue
- A WIP becomes a Not Validated if:
  - The article is complete but created by a KCS Candidate (KCS I)
  - The article is complete, created by a KCS Contributor or Publisher but confidence in the resolution is low
- A WIP becomes Validated if:
  - The article is being worked by KCS Contributor, KCS Publisher, or Coach, they have confidence that the article resolved their issue, and it complies to the content standard.
  - It is important to note that WIP articles should exist only while the issue is being worked on, or when the associated incident is open. When the incident is closed, the WIP should either be deleted because an existing article was found that resolved the issue, or the article should be updated with the resolution and moved to Not Validated or Validated based on the KCS license level and confidence in the resolution. If the problem is not resolved and the incident is closed, as a general rule, the WIP should be deleted or set to Archive. An exception to consider in dealing with unresolved issues and WIP articles is the situation where there is valuable, reusable information that has been captured in the WIP. For example, this can occur if a diagnostic process was documented in the article. In this case, set the article into the appropriate confidence setting based on license level and confidence in the content in the article.

Not Validated, Validated

- Not Validated becomes Validated if the article is being worked/reused by a KCS Contributor, KCS Publisher, or Coach, they have confidence and validation from the requestor that the article resolved their issue, and the article complies to the content standard.

Archived

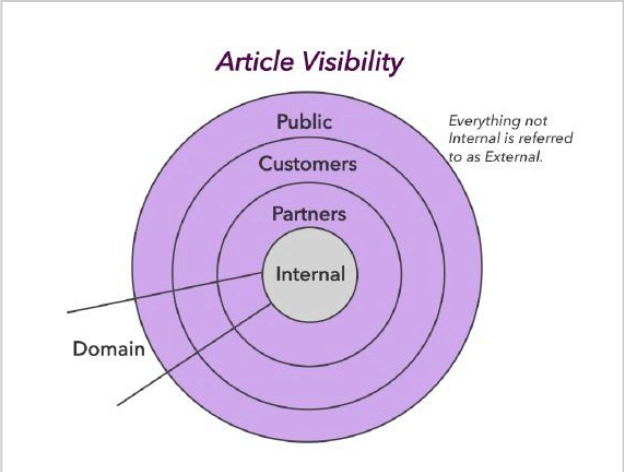
- Any article can become Archived when the objective is to remove the article from the searchable knowledge base without physically deleting it.
- An archived article can be restored to any other confidence setting should it be discovered the article was prematurely archived.

All of the above approaches align with the fundamental KCS Principle of a demand-driven process. Demand drives our attention to the articles that have value. Do not review articles for the sake of review, as they may never be reused. If we are reviewing articles in absence of demand, we are not doing KCS. The only exceptions to this rule are when a Coach and a KCS Candidate are working together, or when the KDEs do Evolve Loop assessments of clusters, patterns, trends, and the new vs. known analysis.

Article Audience: Who Gets to See What

To manage who can see which KCS articles, we use article audience. As an organization, you may decide that you want to control what an external customer or an unidentified requestor has access to. Business rules can then be established based on both article audience and article confidence. You may decide that only Validated articles that you have confidence in can be visible to Partners and Customers. As the article is designated as Validated and the reuse of that article has reached a certain threshold, it may become visible to a given audience. Or, as the KCS implementation matures, the audience may expand.

- Internal
- Within a Domain
- Partners
- Customers
- Public



INTERNAL - Only an audience internal to an organization will be able to see the article. Note: anything with wider visibility than Internal is referred to as External.

WITHIN A DOMAIN - A group associated with a particular product domain, topic, job function, department etc.

PARTNERS - Someone who is not an employee but acts as a trusted extension of the organization.

CUSTOMERS - Customers or users of our products or services . These articles typically are made accessible via a

web-based self-service portal for registered users.

PUBLIC -The KCS article is intended for anyone unidentified in the public domain. A common practice is to have this article optimized and indexed for a publicly available search appliance like Google.

Demand-based audience and confidence can be achieved by using these attributes in conjunction with reuse.

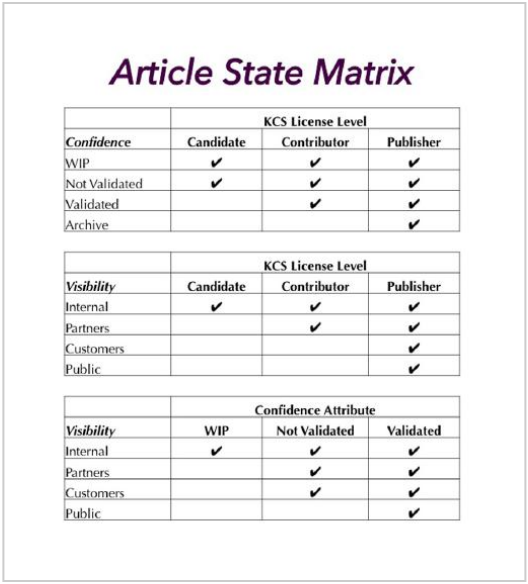
- Early in the KCS adoption, External, Validated articles become visible based on reuse. Demand drives our attention to those articles that have value and therefore should be made visible to external users (customers, end-users, or partners). This demand-driven migration of articles **should be temporary**, replaced by the just-in-time publishing model of a mature KCS environment. Unfortunately, organizations often get stuck in the "make External only after internal reuse" phase and as a result sub-optimize the benefits of KCS. Patterns of reuse external to the organization are different than patterns of reuse internally. While we are learning to do KCS, demand or reuse is a reasonable way to know what should be published. However, to fully capitalize on what we know internally, we must make a high percentage of that available externally, and quickly. This is why the "make External after reuse" model should be a temporary practice on the KCS journey.
- In a mature KCS environment (Optimize and Innovate), we use the 90/0 goal: we should share most of what we know externally as quickly as we can. 90% of the articles in the KB should be available at or before the time of case closure. This will increase use of and success with self-service.

In the Optimize and Innovate phase of KCS transformation, we should have lots of KCS Publishers and we should be making judgments on moving Not Validated or Validated articles to become visible outside the organization as we create or reuse articles. The judgment to set the article audience is based on our confidence in the resolution and adherence to the the content standard. In Optimize and Innovate, most articles created would be in Validated state with audience set to External.

A KCS Publisher can modify a Validated article. Those who are not yet KCS Publishers, but who notice that the article does not comply with the KCS content standard, or otherwise requires enhancement, correction, updating, or improvement, should flag the article and mark it as Not Validated for review by a KCS Publisher (who may also be a Coach or a Knowledge Domain Expert). Some KM technologies allow those who are not yet KCS Publishers to edit an unpublished version of the document while the current version stays on the web; this is desirable as it removes rework by the KCS Publisher, who need only approve the changes. In general, articles that have been published to the web should not be removed from the website if they have been flagged, although if the knowledge developer feels a Published article poses an active risk of harming customers, he or she should escalate the issue to a Coach or Knowledge Domain Expert immediately.

As we find and use KCS articles, we should improve them; reuse is review. As "flag it or fix it" becomes part of the culture, we are taking responsibility for the content that we interact with. This dynamic ensures the content being used is constantly being reviewed and improved. As the KCS articles improve and are validated through use, they should become visible to a broader audience and eventually be made visible externally. Early on in the KCS adoption process, organizations implement the concept of migrating content to be visible to customers through a variety of demand-driven techniques (mentioned above). In a mature KCS environment, a high percentage of content publishing is happening in real-time or just-in-time. (See case studies for examples of just-in-time publishing in action.)

To manage article audience, and thus appropriate levels of security for the knowledge, we recommend creating an Article State Matrix (example shown). This reference document maps the audiences (internal, partners, external) against the KCS article attributes that affect visibility (state, knowledge domain, special considerations) and defines access rights.



Article State Matrix

	KCS License Level		
Confidence	Candidate	Contributor	Publisher
WIP	✔	✔	✔
Not Validated	✔	✔	✔
Validated		✔	✔
Archive			✔

	KCS License Level		
Visibility	Candidate	Contributor	Publisher
Internal	✔	✔	✔
Partners		✔	✔
Customers			✔
Public			✔

	Confidence Attribute		
Visibility	WIP	Not Validated	Validated
Internal	✔	✔	✔
Partners		✔	✔
Customers		✔	✔
Public			✔

As the KCS practices mature in the organization, a just-in-time model for external visibility should be adopted, so that a high percentage of knowledge workers are licensed to make articles externally visible without review. This level of maturity takes time to develop and is most appropriate when the KCS workflow and content standard (discussed next) are well understood and have become second nature for the responders. Just-in-time external publishing requires people to make good judgments about technical and content accuracy. If they are licensed and confident in the article's accuracy, they should make it visible to the largest audience externally. If they are not confident in the article's accuracy, they should request a technical review. Judgment is required.

Options for Transition

Again: to fully capitalize on what we know internally, we must make a high percentage of that available externally, and quickly. In some environments, while we are learning to do KCS, demand or reuse is a reasonable way to know what should be published. However, making articles externally visible based on reuse should be temporary, replaced by the just-in-time publishing model of a mature KCS environment. If you must do it:

- In some organizations, a Validated article becomes External automatically when the reuse count is hit. The philosophy here is the article has been used and therefore reviewed a couple of times, so it is sufficient to be External.
- The "conveyor belt model." Some members have had success with a conveyor belt model of automatically making articles External. Once an article has been reused three times internally, a timer starts for that article and it will be External in five days. People in the organization can opt-in to review the article or pull the article off the conveyor belt by changing the state or audience at any point during the five-day period. The article will be External when the timer expires, whether or not it is Validated.

Remember that patterns of reuse external to the organization are different than patterns of reuse internally. Requestors will persue resolutions for many issues through a self-service mechanism for which they would never open an incident. This is why the "make External after reuse" model should be a short-term practice on the KCS journey.

Article Governance: Who Can Create or Modify

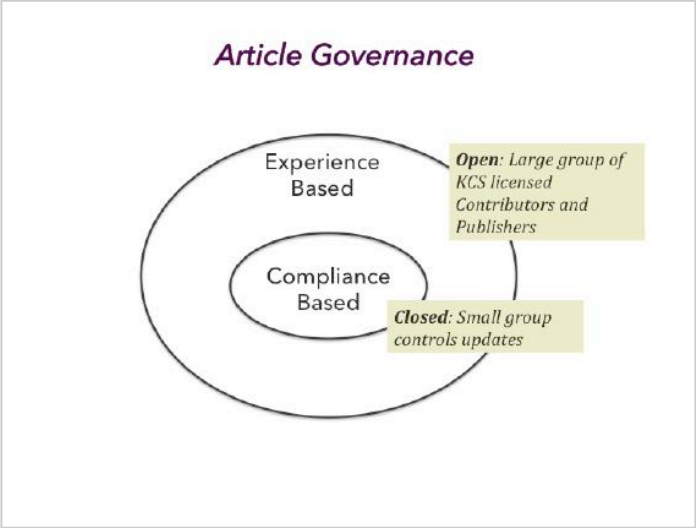
Article governance is an attribute of an article that allows you to control sensitive, critical, or regulated information. Not all articles have the same requirement for compliance reviews. Some articles are based on the collective experience of those who use the articles. Other articles have policy or legal information that require tight control.

The governance attribute used in conjunction with KCS Roles enables us to manage articles and their state specific to

the compliance requirement.

The two governance attributes that allow us to distinguish the collective experience articles from those that have compliance requirements are:

- Experience Based
- Compliance Based



Experience Based - The Experience Based attribute is the most open level of governance, and control is a function of being a member of the community and having an identity. Sign-in is required. The number of people with this level of rights and privileges will be very large; they can create and modify articles with the Collective attribute. The individual's KCS license level defines their rights and privileges in setting the audience attribute (Internal, Partner, External) and the confidence attribute (Not Validated, Validated)

Compliance Based - The Compliance Based attribute is restrictive, in that only designated individuals or specific groups of individuals can create and modify articles with the Compliance attribute. These articles contain information that describes policy, regulatory, or legal information. While everyone should be able to comment on all types of articles, not everyone can create or modify articles with the Compliance attribute.

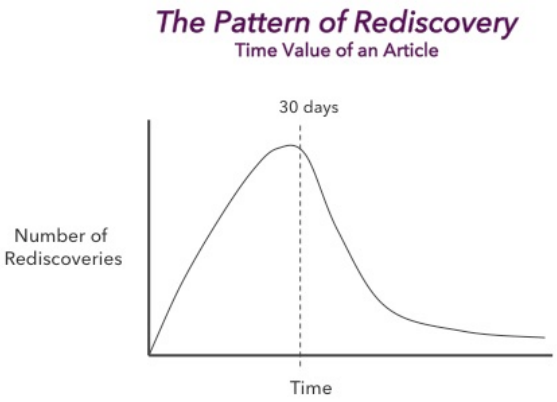
KCS Stagnation

The flow or movement of articles through the life cycle states is an important indicator of the health of the KCS system. This is not to say all articles should necessarily move through the life cycle states, as reuse should be the driver of what moves and what doesn't. Not Validated articles that have never been reused, or that we don't have confidence in, should stay Not Validated; Not Validated articles are okay. However, articles that are being reused or that we have confidence in must eventually make it to Validated and visible outside of the organization (if appropriate).

Many organizations who do a great job at Adopting in Waves and Building Proficiency will realize significant operational improvement, and then the system slowly dies. People lose interest, participation rates drop off, and the benefits decline. The common underlying theme in these scenarios is KCS stagnation: the flow of articles stops. By this we mean the organization has not created a self-service mechanism, or the rate at which articles are getting made available outside the organization is not sufficient to support success with self-service. The primary motivation for people to create and maintain the knowledge base is the promise of reducing redundant work - not solving the same problem over and over. If articles are not visible externally, or there is no effective self-service model, the responders will not see a change in the ratio of known vs. new issues they are working on. They will lose interest in the KCS practices. The flow of articles through the states is critical for the sustainability of the KCS practices.

The Time Value of a KCS Article

KCS proposes that knowledge base content is different and should be managed differently from other types of technical content, such as documentation, white papers, or manuals. Knowledge is dynamic and needs to be created, managed, and delivered for just-in-time accuracy and freshness. One justification for this is that the value of support knowledge begins to diminish 30 days after the issue is first discovered. Unfortunately, many non-KCS organizations take 60-90 days or more to document and release new articles. This is an expensive proposition that misses a major portion of the content's window of opportunity.



HP and Dell both conducted studies that produced a rediscovery curve of this shape.

Technique 5.3: Developing a Content Standard

We have discussed the many benefits of having a simple, single structure for articles rather than having free form text, but how much structure is enough? How do we communicate across the organization so that everyone captures the appropriate information in a predictable format? This is the purpose of the *content standard*. This formal document or template built into the technology help fields describes decisions made about KCS article structure and content.

Through years of KCS deployments, the collective Consortium experience indicates that about 70-80% of the content standard is the same across large or diverse interaction networks (companies or organizations), while 20-30% of the content standard is tailored to a specific knowledge domain or division or an organization.

A content standard, as the name implies, defines standards for content that promote consistency. The content standard needs to cover a broad set of elements. Here is a list of elements that need to be included in the content standard:

- **KCS Article Structure, Definition for Each Field** —a list of article fields with definitions for each. Including issue, environment, resolution, cause, and metadata
- **Good and Bad Article Examples**—the contrast between bad articles and good articles reinforce the concepts and intent behind the field definitions
- **Metadata Definitions**—a list of the article attributes and the meaning and implications of each as well as how each metadata element is set (automated or manual)
- **Article Confidence**—as defined in KCS Article State
- **Article Audience**—who gets to see what as defined in KCS Article State
- **Article Governance**—mechanism for compliance based articles, defined in KCS Article State
- **Templates**—if we are using more than one template, a list of templates available and criteria for the use of each as well as directions for filling the fields out in each template
- **Style Guide**—describes the preferred writing style for articles
- **Supporting Material**—format and criteria for references and links from articles
- **Vocabulary**—preferred terms aligned to the audiences' context and level of expertise, voice, standard for environment statements; platforms, product names, releases and versions; supports trademark protection
- **Multi-language Considerations**—writing guidelines that ease translation effort, promoting International English
- **Multimedia Considerations**—criteria for deciding what type of content and for what audience is multimedia appropriate
- **Quick Reference Guide**—one page reference guide with hints and tips on how to write good articles (aligns with the content standard as a whole and especially the Content Standard Checklist). See Appendix E for an example.

The content standard should be developed and owned by the people who use the content everyday: the knowledge workers. The content standard design should be done by a cross-functional team made up of people who will be using it

to create KCS articles.

Different groups may use different content standards, but they must be careful to keep enterprise-wide considerations in mind.

Technique 5.4: Creating Evolve Loop Articles

Two Types of KCS Articles

Earlier we discussed the interdependent nature of the Solve Loop and Evolve Loop. Each loop generates knowledge by considering articles at different levels. To recap, Solve Loop articles are created and improved by knowledge workers in the role of responder, while they are resolving issues. At the time Solve Loop content is created, it is difficult to judge how important or valuable it may be, but if an issue is worth a response, it is worth capturing in the knowledge base for others to reuse and thereby contribute to the patterns that emerge in the Evolve Loop analysis.

Ideally, Solve Loop articles are developed just-in-time based on requestor demand. Solve Loop articles must adhere to the content standard so that the articles have a consistent structure and are findable and usable by the intended audience.

Evolve Loop articles are high-value articles. These articles are usually created by Knowledge Domain Experts based on patterns and trends in article reuse or the analysis of self-service activity. Evolve Loop content is high-value because it is derived from the patterns of reuse or the clustering of articles around a common theme or issue, or critical processes and procedures. Evolve Loop content generally represents a very small percentage of the total knowledge base.

For a thorough look at creating Evolve Loop articles and other Knowledge Domain Analysis activities, please visit the KCS v6 Knowledge Domain Analysis Reference Guide.

Technique 5.5: Archiving Old Articles

Age and Size Shouldn't Really Matter

One of the by-products of the KCS Practices is improved success with search. If we are only finding relevant articles when we search, we don't have to worry about how big the knowledge base is or if it contains "old stuff." If it is not relevant, old stuff should not show up in our search results. In fact, we could make the case that on those rare occasions when we need the old stuff, the value of the seldom-used, old articles is higher than the set of frequently used articles, since the knowledge about the frequently used articles also exists in the knowledge workers' heads. Imagine a situation arises about an old issue, requiring knowledge that people have long forgotten or those who knew it have left the organization. Having access to the older, seldom-referenced knowledge can be of tremendous value. But only if it shows up when it is relevant.

However, findability is a common problem as organizations grow their knowledge. Archiving old articles treats the symptoms of findability, not the cause. Relevance is the key. Relevant search results are enabled by a combination of: context, structure, rich environment statements, and search technology. KCS addresses the first three - the content factors- but it does not address search technology. While search technology can help, it can not overcome deficiencies in our content. If we are having findability problems, the first place to look for opportunities to improve our search success is to review our context, structure, and the richness of our environment statements. More information about the role technology plays in KCS is covered in the Process Integration section.

> Addressing findability issues by reducing the size of the knowlege base is treating the symptom, not the cause.

Some have tried to improve relevance by reducing the number of KCS articles in the knowledge base. This reduction will compromise the completeness of the knowledge. The greatest value from the knowledge base comes from it being a complete collection of the organization's experience and our ability to quickly find what we need when we need it.

This is not to say that knowledge base cleanup and maintenance should never be done. There is definitely a need for ongoing knowledge base maintenance, but it should be done in a way that improves the findability of what we collectively know, not by reducing what we collectively know. Maintaining a knowledge base is like tending to a garden: it requires constant weeding. We have to be sure we can distinguish the weeds from the flowering plants, some of which may only occasionally produce beautiful flowers. The "reuse is review" and the "flag it or fix it" Solve Loop activities play an important role keeping our knowledge up to date as we interact with the knowledge base. We have to compliment that with a knowledge base maintenance strategy that looks at the collection of knowledge in a given

domain. This is an important part of the Knowledge Domain Analysis process and is typically done by the Knowledge Domain Experts (KDEs).

Technique 5.6: Dealing With Legacy Data

Migrating and Integrating Legacy Data

Most organizations face the question of what to do with non-KCS knowledge content that exists in a legacy knowledge base being replaced by a new KCS knowledge base. We have never seen a mass migration of legacy content work in a KCS implementation. The legacy content is not in the KCS structure, nor is it expressed in the context of the requestor. Those who have done a mass migration have ended up removing the legacy content because it disrupts findability. The investment of time and money to clean, write scripts, and move legacy knowledge is not worth it, and turns out to be counter productive.

A better strategy to deal with legacy content is to create a demand-based process that will help us identify the legacy content that has value. Keep the legacy content in a separate repository and make it available to knowledge workers to search. Let requestor demand focus our attention on the legacy content that has value. Create KCS articles in the new knowledge base for the content that is being used from the old knowledge base.

Following are some considerations that support a demand-based migration strategy:

- Make the old knowledge repositories read-only
- Search the new knowledge base first
- If a KCS article is not found in the KCS knowledge base, search the old knowledge repositories
- Repurpose the old content that is useful (based on demand) to create articles in the KCS knowledge in the requestor's context and in the KCS structure.

Consortium member experience shows that 90%-95% of what is in the old knowledge base will never be referenced. With the demand-based process outlined above it doesn't take long before the knowledge workers stop searching the legacy content. And the migration effort is only spent on the content that has value.

A variation on this demand-driven theme: if our legacy system allows us to create a list of the most used items in the legacy database, use the items on that list in the KCS training as exercises for knowledge workers to rewrite the most frequently used items as KCS articles. This is not only a great training technique, it helps seed the new knowledge base with valuable legacy content - without disrupting findability.

Technique 5.7: Priming the Knowledge Base With New Information

KCS is a demand-driven system; this means we should not add content in the absence of demand. Just as we should not try to anticipate the future value of an issue (if it is worth resolving, it's worth capturing), we should not create articles in anticipation of demand. This often causes concern and raises the question... What if we know a situation will occur with the release of a new application or process? If we know something will happen it is probably based on past experience, as in, the last time we did *this*, *that* happened. Or, we know from the alpha and beta testing that users will experience *these* issues. That is demand-driven.

The general rule of "don't add articles until someone asks" raises a problem when introducing new products or processes. How do we prime the knowledge base for them?

Perhaps the worst thing we can do is have development or engineering write articles about the new product or process: those articles will be in the context of how the product was designed and built, not how customers will use it, and not how it will break. We can, in fact, capture information about new products in a useful context. As a new product or process is going through alpha and beta testing or user acceptance testing, we should capture those experiences in the context of use. Creating articles that address the issues users are likely to encounter because of what we learned in the pre-release testing is the best way to seed the knowledge base.

During product beta cycles, we pay special attention to creating content in context of the beta testers experience. Generally these pre-release articles should be in a Draft or Not Validated state (not visible to customers) until they have been reused to solve a customer issue, and, as a result, updated with the customer context and then made available externally for direct customer access through self-service.

KCS articles can also be pre-populated in the new knowledge base during the KCS training and pilot phase. Students bring their top ten current issues to training and use these issues to practice creating KCS articles. We structure and enter the knowledge according to the KCS content standard. As these KCS articles are reused in the request-resolution process, they should be modified to include the requestor's context.

Technique 5.8: Global Support Considerations

The majority of Consortium members operate in global, multi-lingual, multi-cultural environments. Both the markets for revenue growth and available talent and resources are in emerging markets where the language and culture are different from those of the home office. Many companies in the high tech sector have standardized on English as the language for business, even though they are based in non-English speaking countries, serve markets, and have employees in non-English speaking parts of the world. This presents some challenges when it comes to sharing knowledge on a global basis. As best we know, there is no easy answer. Cultural sensitivity and language translation are both difficult and expensive to maintain.

KCS as a methodology does not address cultural sensitivity but KCS does offer some relief in the area of multi-language support. If an organization adopts the content structure and style recommended in the KCS methodology of "complete thoughts, not complete sentences" then this creates the following benefits in a multi-language environment:

- Complete thoughts are often easier to comprehend than complete sentences
- The KCS structure gives meaning and context to the words and phrases in the article

The use of machine translation has increased dramatically over the past few years. It is not perfect but it is gaining acceptance as sufficient for support content. Following are some examples of how companies are leveraging machine translation:

- "Just do it" - Use machine translation for all support content in the knowledge base and translate it into selected languages. Intel uses machine translation to offer their support web site in five different languages
- "Demand driven" - Limited machine translation; only articles that have reuse get translated
- "The hybrid" - A hybrid approach of machine translation with a manual post edit for reused articles
- "Side by side" - Microsoft has found that offering the original article along side the machine translated article greatly increases user's confidence and therefore use of machine translated articles

For more information on machine translation visit the Translation Automation Users Society at www.taus.net.

Technique 5.9: Knowledge Domain Analysis

Evolve Loop content processes are critical for continuous learning, innovation, and improvement. They leverage the Solve Loop content, create incremental value for the organization, and help to elevate awareness and sensitivity to the requestor or customer experience in the organization. As the organization matures in its use of KCS, Evolve Loop content is managed by an important function: Knowledge Domain Analysis. This critical function assures that issues are resolved effectively and efficiently, while also gathering and communicating data about the impact KCS in having in the organization.

To help maximize the benefits of KCS, Knowledge Domain Analysis focuses on the knowledge base and pays attention to the quality of the articles, the effectiveness of the workflow that produces and improves the articles and, perhaps most importantly, the use of the articles.

The knowledge workers doing this function, Knowledge Domain Experts (KDEs) must have both deep subject matter expertise as well as a profound understanding of KCS. KDEs look after the health of a collection or domain of knowledge, usually a subset of the knowledge base that aligns with their expertise.

The KDE seeks to optimize the creation, improvement, and use of articles as well as identify patterns and trends of reuse to identify potential product, process, or policy changes that could eliminate the root cause of the most frequent issues. Based on the analysis, the KDEs work with Coaches and the KCS Council to improve the content standard and the KCS workflow. Success of the Knowledge Domain Analysis function is measured through improvements in findability, self-service use, and success rates and incident volume reduction that is a result of corrective actions taken to eliminate the cause of pervasive issues.

For a detailed look at what's involved in Knowledge Domain Analysis, including creating Evolve Loop articles and conducting a New vs. Known analysis, please visit the KCS v6 Knowledge Domain Analysis Reference Guide.

Knowledge Domain Analysis outputs include the identification of :

- Improvements to the content standard and process integration (workflow)
- Findability issues: knowledge exists but is not being found - search performance and optimization
- Content gaps: knowledge people are looking for that does not exist
- Content overlaps: consolidating duplicate articles, identifying the best or preferred resolution among many proposed resolutions
- Improvements in how we leverage known issues, eliminating re-work, improving access and findability
- Improvements in how we solve new issues, suggestions for problem solving and collaboration to solve new issues quickly
- Pervasive issues: facilitating root cause analysis and working with business owners on high impact improvements

- Value of the knowledge base, such as article reuse rates, self-service success, and contribution in improving time to resolve
- Archiving strategy for the knowledge base

Technique 5.10: Content Health Indicators

As the organization gets started with KCS adoption, the KCS Coach plays a major role in the quality of the knowledge base content by reviewing the articles created by the KCS Candidates who do not yet have the permissions to put articles in a Validated state. The Coach's goal is to support the KCS Candidates in learning to do the Solve Loop, adhering to the content standard, and using the structured problem solving process.

KCS proposes a competency or licensing program that uses the Content Standard Checklist (previously Article Quality Index or AQI) and the Process Adherence Review or PAR (previously known as Process Integration Indicators or PII) to assess the knowledge worker's ability to consistently create articles that align to the content standard and follow the KCS workflow. By earning a license, knowledge workers are recognized for their KCS understanding and capability, thereby earning rights and privileges in the system.

The licensing program ensures that people understand the KCS workflow and the content standard. This program contributes to the level of quality and consistency of the articles in the knowledge base.

While there are many checks and balances in the KCS methodology to ensure quality articles, there are five key elements that contribute to article quality:

1. A content standard that defines the organization's requirements for good articles
2. Content Standard Checklist (below), which determines an article's alignment with the content standard
3. Process Adherence Review (PAR) - are we following the KCS workflow?
4. The licensing and coaching model
5. A broad and balanced performance assessment model.

The Content Standard Checklist

In previous iterations of the KCS methodology, we have referred to this as the Article Quality Index. The idea was that especially for large and distributed teams, organizations must have consistent quality metrics for rating the article quality and performance of those contributing. In practice, this meant that organizations put all of their focus on the number generated as an indication of the quality of their knowledge base.

In reality, this is meant to be a coaching tool to help knowledge workers understand and remember how we are aligning our articles with the content standard, and perhaps to help the organization have a broad picture of how well knowledge workers are understanding and applying that content standard. Because we are addressing technical accuracy with the practice of "reuse is review," this is not meant to serve as a technical review or to look at other aspects of article value or

quality.

The Content Standard Checklist can be customized and evolve over time, and should be consistent with the content standard for a "good article," quantifiable to facilitate reporting, and shared with both management and the individual (as part of a conversation about understanding and behavior - not about the number). To begin, we suggest these basic checks in the form of yes/no questions:

- Is the article **Unique**? - not a duplicate article, no other article with same content whose create date preceded this articles created date (this is a critical part of the Content Standard Checklist)
- Is the article **Complete**? - complete problem/environment/cause/resolution description and types
- Is the **Content Clear**? - statements are complete thoughts, not sentences (as appropriate)
- Does the **Title Relate to Article**? - title contains description of main environment, and main issue (cause if available)
- Are the **Links Valid**? - hyperlinks are persistently available to the intended future audience
- Is the **Metadata Correct**? - metadata set appropriately: article state, audience, type or other key metadata defined in the content standard

We capture the answers to these questions in an *Content Standard Checklist* spreadsheet. In the example below, we review a handful of articles that each knowledge worker has linked or created in a specified period of time, and tally the number of "yes" answers in each column. The resulting percentage reflects the frequency with which a knowledge worker (or group of knowledge workers) followed the content standard while interacting with those articles.

While we can use this as an indicator of the general health of the articles in the knowledge base, it is best used as an indicator of the understanding of the KCS content standard. Tangible, quantified information like this improves the quality of feedback we can provide to individual knowledge workers to enhance skills development and drive article health.

Start simple. Here is an example of an Content Standard Checklist spreadsheet focused on the big six items:

Simple - Content Standard Checklist

			Article			Content				
	# of articles with issues		15	4	4	14	4	0		
Last Linker	Content Standard Met	Articles Reviewed	Unique	Complete	Content Clear	Title Relates to Article	Hyperlinks Valid	Metadata Correct	Total Points	Total Possible Points
Al	90.48%	14	14	14	12	12	14	14	80	84
Beth	90.00%	10	10	10	9	9	9	10	57	60
Chuck	28.57%	7	1	4	7	3	5	7	27	42
Hector	94.44%	12	11	12	12	11	12	12	70	72
Grace	86.67%	10	7	10	10	9	10	10	56	60
Joe	95.83%	8	8	8	8	8	7	8	47	48
Kim	60.00%	10	5	9	9	5	10	10	48	60
Group Content Standard Met										
	90.4%	71							385	426

Note: 90% is the goal. Percentages are calculated using issues found times 2 to create more differentiation between those doing well and those who aren't

Problems	Too many duplicates being created Titles not well written These might be related!

Percentages are calculated using (total possible points minus total points times 2) divided by total possible points. In Excel, the formula for Al's Content Standard Checklist % = 1-(2(L5-K5))/L5*

In both example spreadsheets here, the errors carry a weight of 2. This is done to enable the percentages to better reflect differentiation between those doing well and those who need some help. In these examples, anyone with an Content Standard Checklist percentage below 90 should get some attention from a Coach. If they are consistently below 80, they are at risk of losing their KCS license. It is important to monitor trends over time on the Content Standard Checklist percentages for both teams and individuals.

This matrix can be customized to suit an organization's requirements. A consumer product may need more emphasis on usability and formatting compared to a highly technical audience. But: don't over-engineer it! Start simple and evolve it based on experience.

Over time, as our KCS adoption matures and the organization gets good at the basics, we might add some additional or more granular areas of focus. We find that the content standard is 70-80% common across organizations and 20-30% tailored to a specific organization. Some of the factors in the criteria will be influenced by the knowledge management technology being used in the environment. The content standard sets the criteria for the checklist, and must be tailored to the environment and the tools being used.

Following is a sample Content Standard Checklist spreadsheet in a mature KCS environment, for example purposes only.

Mature - Content Standard Checklist

# articles with issues in this area			4	12	2	1	10	2	3	0	0	1	2	2	1	7		
			Article			Content								Attributes				
Last Linker	Content Standard Met	Articles Reviewed	Variation of Symptoms	Unique	Complete	Single Statements	Environment Reflects Issue	Environment to Standard	Fix Complete	Complete Thoughts	Appropriate Requestor Info	Links Accessible	Hyperlink Correct	Audience Correct	Status Correct	Type Correct	Total Points	Total Possible Points
Al	95.2%	15	14	15	14	15	14	15	15	15	15	14	14	15	15	15	205	210
Beth	91.7%	12	12	11	12	12	12	12	12	12	12	12	11	10	12	9	161	168
Chuck	87.8%	7	7	3	7	7	5	7	7	7	7	7	7	7	7	7	92	98
Hector	98.0%	14	14	14	14	14	14	14	14	14	14	14	14	14	13	13	194	196
Grace	94.3%	10	10	10	9	9	9	10	9	10	10	10	10	10	10	10	136	140
Joe	92.9%	8	8	8	8	8	8	8	7	8	8	8	8	8	8	5	108	112
Kim	75.3%	11	8	4	11	11	5	9	10	11	11	11	11	11	11	11	135	154
Group Content Standard Met																		
	91.3%	77															1031	1078

Note: 90% is the goal. Percentages are calculated using issues found times 2 to create more differentiation between those doing well and those who aren't

Problems
Environment statements not well understood
Too many dupllicates being creation
These two issues might be related!

Some key ideas to note in the mature example (for more details, see Practice 7: Performance Assessment):

- Compare the number of articles reviewed for each creator. A legitimate sample size is important. Creators Chuck and Ed may have too few articles to be fairly weighed.
- In the first row (# of articles with issues in this area), organizational performance is visible. This is a great area to focus on with groups of KCS Coaches. In this example, many articles are duplicates, incomplete, or unusable. This result could mean knowledge workers need more training on searching and documenting content.
- One contributor, Kim, is a prolific contributor, but also leads in the top three categories of problems. Attention from a KCS Coach is merited.

Evolve the Content Standard Checklist items based on experience. As an organization matures in its use and confidence in KCS, it becomes easier to pay attention to more granular or refined content considerations like versioning, global distribution, use of multimedia, and measuring team-based contribution in addition to individual contribution.

Many mature organizations develop a weighting system for this more complex criterion as the violations do not all have the same impact. For example, a duplicate article is a more serious error than an article that is too wordy. (See the

Progress Software case study on the KCS Academy Resources page for an example in practice.) Again, the criteria and weighting should be done based on the needs of the organization and should be considered only after the organization has had some experience with the Content Standard Checklist process.

Knowledge Sampling for the Content Standard Checklist

To complete a Content Standard Checklist spreadsheet, a group of qualified reviewers (usually the KCS Coaches) participate in regular *knowledge sampling* of articles from the knowledge base. While the articles are selected randomly (and assigned randomly to reviewers, so coaches are not often evaluating their coachee's articles), it is important to be sure to sample articles from each individual.

Here is a typical process:

- Develop a Content Standard Checklist and criteria
- Evaluate a sample of articles
- Calculate the content standard percentage met and develop summary reports
- Provide regular feedback to the knowledge workers on possible content standard areas of focus with comments from the Coach who did the evaluation
- Provide periodic feedback to leadership

The Content Standard Checklist should be used for learning and growth of the knowledge workers. During rollout and training, the frequency of this monitoring should be weekly. It will take more time due to the high number of KCS Candidates (people learning KCS). Once the organization has matured, the frequency is typically monthly and should not consume more than a few hours of time per month per reviewer. Note that what the organization focuses on around the content standard should change over time. The elements for assessment at the beginning of a KCS adoption will be more basic than those things the organization will focus on two years into the knowledge journey.

There are a number of considerations for monitoring knowledge health in organizations. With the Content Standard Checklist, we are concerned with an article's compliance with the content standard. Other areas to consider include the Process Adherence Review (PAR), case documentation and handling, customer interaction, and technical accuracy. Organizations have various ways to monitor the health and quality of these important elements of the process.

As organizations reflect on their processes in the Evolve Loop, they are identifying key monitoring elements and ways to integrate monitoring across the processes. One element that is emerging as critical to monitor on a regular basis is link accuracy. This is part of the PAR, which is also done using a sampling technique and can be integrated into the Content Standard Checklist process. Assessing link accuracy also shows up in the New vs. Known Analysis.

Link rates (percentage of cases closed with an article linked) and link accuracy (the article resolves the issue raised in the case) are the key enabling elements for identifying the top issues that are driving support cost (case volume) and user disruption. In order to provide credible and actionable input to product management and development about the highest impact issues, we need to have link rates that exceed 60% and link accuracy that exceeds 90%. Link accuracy is more important than link rate.

Reviewing Articles Through Use

The KCS Principle of demand driven and the Core Concept of collective ownership combine to create efficiency. The idea that people feel a sense of responsibility for the quality of the articles they interact with is critical. The cost and delay of the alternative: that someone else owns article quality and that it is someone else's responsibility to review it, is prohibitive. This sense of collective responsibility is reinforced through coaching, the competency/licensing program, communications from the leaders, the performance assessment program, and the recognition programs. The new hero in the organization is the person who creates value through their contribution to the knowledge base, not the person who knows the most and has the longest line outside their cube.

Feedback to the Knowledge Worker

Knowledge workers must have visibility to their Content Standard Checklist evaluations so they understand where to self-correct. Again, this is as part of a conversation about understanding and behavior - not about a percentage or number. Content Standard Checklist results are also a key tool for the Coaches as it helps them identify opportunities for learning and growth.

Assessing the Value of Articles

As we move through the KCS adoption phases, the knowledge base will grow. We will want a way to assess the value of the articles in the knowledge base. There are three perspectives to keep in mind when assessing the value of articles: frequency of reuse, frequency of reference and the value of the collection of articles. The reuse frequency is a strong indicator of the value of an individual article and is fairly easy to assess. The frequency of reference is equally important and is much harder to assess. The value of the collection of articles has to be looked at from a systemic point of view.

Article Value Based on Reuse

The value of any particular article can be measured by the number of times it is used to resolve an issue. If we are linking articles to incidents, we can easily calculate the reuse count. As we expand into the Build Proficiency phase of the KCS transformation, measuring the reuse of articles becomes much more difficult because customers using the article through self-service do not link articles to incidents nor do they show much interest in answering the oft-asked question, "Was this article helpful?" To assess the value of individual articles in a self-service model, we have to infer value based on a number of factors.

A few Consortium Members have developed article value calculators that take into account the following:

- Page views
- Internal links
- Customer feedback (Member experience indicates that customers provide feedback on a tiny percent of articles viewed: 1-2%)

Article Value Based on Reference

The second perspective is the value of the collection of articles. Even though a specific article may not be the resolution to the issue, an article about a similar issue may provide some insight or remind us of an approach or diagnostic technique that we know but had not thought about. The frequency of reference is extremely valuable and hard to measure.

The Value of the Collection of Articles

The indicators for the value of the collection of content can be calculated based on the rate of customers' use and success with self-service. More specifically, support organizations often look at the subset of the self-service success rate that represents issues for which the customer would have opened an incident had they not found an answer through self-service. This is often referred to with the unfortunate vocabulary of "call avoidance" or "case deflection." This avoidance or deflection view represents a vendor-centric view of support, not a customer-centric view. A customer-centric view does not avoid or deflect customers; it promotes customer success through the path of least resistance and greatest success - for the customer!

How Good is Good Enough?

One of the things we learned from W. Edwards Deming, the father of the quality revolution, is that quality is assessed against a standard or criteria. Quality is not a standalone, universal thing; it is specific to a purpose. In order to manage the quality of our output, we have to know the criteria for what is acceptable and what is not. In the KCS methodology, the quality criteria for knowledge articles is defined in the content standard. However, not all knowledge articles are equal in their importance or purpose. Most organizations deal with different types of knowledge, and not all types of articles have the same criteria for quality. For example, some knowledge articles capture the experience of people getting their work done (where we have a high tolerance for variability and interpretation), while other types of articles describe company policies or regulatory requirements imposed by law for which we have no tolerance for variability or interpretation.

How good is good enough? Well, it depends on the type of information we are dealing with. By identifying in very broad categories the different types of information and their related compliance requirements, we can define both the criteria for a quality article and the governance we need for each type of article. A word of caution here: we do not want to over-engineer the number of article types. We want to start with the minimum, which is often just two: experience-based and compliance-based content. Then adjust the article types and criteria based on our experience. Each organization that adopts KCS must define what is good enough for their various audiences and the types of knowledge they deal with.

Not all knowledge articles will have the same quality requirements.

To understand article quality issues better, the Consortium conducted a survey of its Members' customers. The survey participants were approximately 67% large enterprises (highly complex business production environments of over 300 users) and 27% small to medium businesses (business production environments less than 300 users) from the Americas, Europe, Middle East, and Africa. The remaining 6% were consumers.

This survey assessed customer needs and quality criteria with respect to web-delivered KCS articles providing technical knowledge. This KCS article content could be in the form of known problems, technical updates, or other knowledge base articles. Almost all of the respondents were already comfortable using web self-help, so they may be considered advanced users. Based on experience, however, we believe the results can be extrapolated to reflect knowledge base content as a whole.

Customer response to the survey indicates articles need to be good enough be findable and usable, or what we call "sufficient to solve."

Getting the Basics Right

To begin with, we examined the basic content requirement—the material that must be included in the KCS article. Respondents chose the following, mostly in the category of "accuracy," as **"very important."** Responses are listed in priority order:

- Technically accurate and relevant
- Problem and solution description
- Cause of problem
- Complete information
- Quickly found
- Clarity of content
- Valid hyperlinks
- Configuration information
- Vendor's sense of confidence in the answer

Considered **"somewhat important,"** mostly in the category of "editing and format," were:

- Compound vs. single thoughts
- Complete sentences vs. short statements
- Date created
- Correct spelling
- Grammar
- Last modified
- No duplication of information
- Frequency of usage
- Punctuation

Of **"least importance,"** perhaps not surprising in a technical audience, were the attributes:

- Legal disclosures
- Correct trademarks
- Date last used

Impact on Company Image

Most respondents considered editorial format somewhat important. Since the process involved in achieving editorial perfection can be time-consuming and delay access to information, we decided to assess the impact on corporate image of publishing KCS article information at various levels of editorial quality. The results were revealing. The majority of respondents:

- Disagreed with or were neutral to the statement: "I have a lesser image of a company that withholds support information access in order to technically validate it." (In other words, the majority of respondents did not fault a company for withholding information that was not technically validated.)
- Agreed with the statement: " I have a lesser image of a company that withholds support information access in order to achieve editorial perfection."
- Agreed with the statement: "To gain knowledge faster, I would like an option to select to see support information that has not been fully validated."
- Agreed with the statement: "To gain knowledge faster, I would be willing to take responsibility for using any of the incomplete information should there be mistakes." Note: To mitigate risk from sharing this knowledge, many support organizations require customers to accept a disclosure agreement before seeing the KCS article.
- Would have a higher or at least the same opinion when asked: "If the support information were marked as being in draft format, what opinion would you have of a company that shared everything they know, even if it had editorial mistakes?"

Time/Value Tradeoff: KCS Recommendations

From this survey feedback coupled with other experience implementing KCS, the Consortium feels confident recommending that organizations invest in content speed and accuracy over presentation and format. We should strive for timely and accurate knowledge, ensure we are investing appropriately in training, have a good balance of competencies, develop a licensing model (see the roles section in Practice 7, Performance Assessment), and follow the recommendations for maintaining just-in-time KCS article quality through a sampling process and the creation of the Content Standard Checklist.

When it comes to information completeness and degree of validation, organizations must individually assess the risk-benefit tradeoff of sharing information early. The Consortium's findings should not be used as a substitute for asking customers about their needs in this area. In our experience, the just-in-time information model has become increasingly accepted as the business community has embraced open source, monthly and quarterly software releases, and extended and open beta-testing programs. Appropriate disclaimers, click-to-acknowledge interfaces, and a clear indication of KCS article status (confidence) are all ways to make the KCS article visible earlier and let the customer determine their own risk profile for the situation.

Technique 5.11: Self-Service Success

The benefits of self-service are profound. When done correctly, it becomes the path of least resistance and best results for the audience you serve. The goal in designing the self-service mechanisms is to provide access to knowledge with the least amount of effort on the part of those who are seeking knowledge. We have learned that designing such a system is not trivial. To deliver self-service in a way that minimizes the requestor's effort and maximizes their success requires careful thought and planning. If done well it is mutually beneficial for both requestors and responders.

First, let's define what we mean by self-service; we use a broad definition. Self-service is the mechanism by which anyone can solve an issue (an exception) without an interaction with others. Historically, the most common form of self-service has been a web portal offering information such as frequently asked questions and a searchable knowledge base. However, many of the Consortium members are investing in integrating self-service into the user interface for the product, application, or service. This moves the self-service experience from a distinct and separate event to an integrated experience within the product. Additionally, technology vendors are investing in automation that will detect and repair issues or programmatically assist with the resolution of issues. Regardless of the approach, self-service success is dependent on knowledge.

Have a Strategy and a Process of Continuous Improvement

It is important to develop a strategy for self-service. We should be very clear on why we are developing a self-service capability. Some of the items that should be included are:

- Vision statement
- Who is (are) the intended audience(s)?
- Goals: what does success look like?
- Measures
- Assessment and continuous improvement process

The greatest value from self-service comes from viewing it as a customer engagement strategy. It enables us to leverage what we know to support greater user success. While a side benefit of self-service is cost reduction for the organization, it is not a deflection or avoidance strategy.

Self-Service Design Criteria

In designing the self-service capability there are five key enablers we have observed about successful self-service models:

1. **Findability** - Findability is driven by three things (all of which are enabled by KCS): context, structure, and rich environment statements.

2. **Completeness** - Most of what we know, that is self-serviceable, needs to be available quickly (also a focus of KCS). While the 90/0 rule (90% of what we know is made available 0 minutes - at or before the time of case closure) is a bit provocative, we continue to learn how time sensitive knowledge is.

3. **Access** - Is it easy to find the self-service access point? Ideally access to self-service is integrated into the user interface and is context sensitive for where we are in the product. If self-service is a portal or web page, is it obvious and easy to find?

4. **Navigation** - Is the navigation in the self-service mechanism intuitive for the users and does it align with the requestor's intent? "No dead ends" ensures there is a smooth way to transition from self-service to assisted support: click to chat or click to submit. Navigation should provide for both browsing and searching.

5. **Marketing** - Self-service requires a marketing plan. The "build it and they will come" model doesn't work for self-service mechanisms. We have to take overt, intentional actions to get requestors to use it. If they have a positive experience (see items 1- 4 above) they will use it...a lot.

A positive experience with self-service means that the requestor will use it again. Not only will they come back, but they will use it a lot; in fact, they will use it more often than they ever requested assistance in the past. As a general rule of thumb, if requestors find helpful information 40-50% of the time, they are likely to use self-service again. This is the industry average for self-service success (see Service XRG for the research). In a mature KCS environment where 90% of what we know is available on self-service immediately, the success rates reported are in the 80-85% range!

Findability

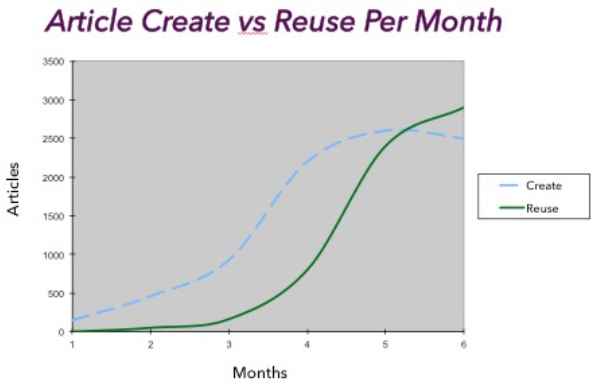
It doesn't matter how much content we have available to requestors on the web - if it is not in their context, they are not likely to be able to find it. This re-enforces the need for capture in the workflow. As we discussed in the Capture practice, it is very difficult to re-create the user's perception of the issue if we are not users, and if we know the answer. Creating articles in the requestor's context requires that we capture their context or experience when they first express it. The second factor for both findability and readability is structure. KCS proposes a simple structure for knowledge articles which helps the search engines be more effective and improves the users experience. The last key element in findability is rich environment statements. The environment statements in the article helps the precision and confidence that we have the correct article for our issue.

Completeness

How many articles are visible externally, and how fast? The primary enabler of success with self-service is volume and speed. The goal is to get as much as we know into the self-service channel as fast as we can. For requestors to be successful with self-service, articles have to be making it to a Validated state (depending on the business rules) with audience set to External. In the early phases of the KCS adoption, this is driven by reuse. Articles that are reused internally are moved externally quickly. In the Optimize and Innovate phase, we want to be validating and moving externally as much as we can in the moment. The model of reuse driving validation and external visibility should be a

temporary state, because we know from member experience that reuse rates on the web are different from reuse rates in an assisted model. It turns out that requestors will use a good self-service model 10 times more often than they will call us. An issue that one requestor raised but has never been reused internally might be used externally a lot; other requestors would use the information, but would not bother to contact us for an answer. Our goal is to get most of what we know into the self-service model as quickly as we can.

When do we turn on and promote web self-service? If our KCS articles are complimentary to content we already have in the self-service channel, then an incremental approach might work. If we are building a new self-service knowledge base, when do we have enough content in the knowledge base to ensure a 40-50% success rate? One key indicator of sufficient volume in the knowledge base is when the reuse rate of articles intersects with the create rate for a given domain. Plotting the team's create rate against the reuse rate over time gives us a sense of how often the users within the support organization find something useful in the knowledge base (reuse) versus how often they are creating new articles. When the lines cross it means that they are re-using as often as they are creating, or 50% of the time they are linking to an existing article. It is now time to enable and promote the self-service model.



The point at which the create activity equals the reuse activity indicates there is sufficient content in the knowledge base to enable external requestors to find something useful 50% of the time.

Three caveats: first, the linking quality for the domain needs to be at 90% or above and the linking rate has to be in the 60-80% range. This means the internal users are using the knowledge base (creating, re-using, and improving articles) in the problem-solving process a high percentage of the time. Second, articles must be making it to External visibility/ audience. And third, the content has to be in the requestors' context which reinforces the finadability factor.

Access

Designing access to self-service to be obvious and low effort is another key to engaging the audience in self-service. Making the self-service mechanism easy to find for the requestors is not trivial. As mentioned earlier, integrating access to self-service knowledge into the user interface for the product is ideal but typically requires a sizable investment. If you are using a web-based self-service portal it is important to make it easy to find by the audience you intend to serve.

Navigation

Research has shown that "no dead ends" is the number one factor for users in deciding if they would use self-service again. "No dead ends" means once the requestor has started the problem-solving process in the self-service channel, they don't have to stop and start over if they don't find something helpful. An example of "no dead ends" in the self-service interface is the click-to-submit (create an incident) or click-to-chat functionality. If the self-service model isn't helpful, there is a graceful transition to the assisted model. Because the self-service activities of the requestor are captured and made available to the responder, the requestor doesn't feel like they are starting over. An in-depth research project at Microsoft found that even when customers were unsuccessful with self-service, they were far more willing to go back to try it again if there were no dead ends.

Another key factor in requestors willingness to use self-service is the availability of multiple ways to find things. People use different methods of finding information based on a number of factors. Options for finding articles include:

- A list of product specific, frequently asked questions or "top ten" articles
- An index or table of contents
- Basic search
- Advanced search

Good user interface design enables both browsing and searching and is critical to self-service success.

The other design criteria is understanding the audience's intent in using self-service. What are the top three to five reasons people use the self-service mechanism?

Marketing

The "build it and they will come" model doesn't work for self-service. Once we have taken care of the first four success criteria: volume and speed, findability, and access/navigation, we have to think about how to get requestors to use self-service. Trying to change our requestors' behavior is not trivial. Engaging a marketing specialist is recommended. Get advice from those who understand messaging and communications and build a marketing plan.

In addition to a marketing plan, below are some tactics that have been successfully used to encourage their customers to use self-service. We offer these as observations, not recommendations; these tactics must be evaluated based on the business and customer engagement model.

- Recorded message promoting self-service (when requestors call for support)
- Extended hold times - make self-service the path of least resistance and best results
- Turn off the phones - make self-service the only path. Requestors can only open an incident via the self-service portal (we must have high confidence that the requestors' self-service experience will be positive)
- Co-browsing - as a responder solves issues, the requestor can see the responder's desktop and watch them search (teaching them to use the self-service tools)
- When sending a requestor a resolution, send them the link to the article in the online knowledge base (promotes exposure)

Requestor use of and success with self-service becomes two critical measures to assess the success and health of KCS

in the Optimize and Innovate phase. If the articles are not making it to the self-service model or if customers are not using self-service, the KCS implementation will stall.

For some examples of good support web sites see the Association of Support Professionals (ASP) list of Ten Best Web Support Sites. The ASP conducts an annual assessment of support sites and the criteria they use is available on their web site. It is a great collection of attributes to use in designing your self-service support mechanism.

Technique 5.12: Self-Service Measures

Measuring self-service success and the self-service experience is hard. In the assisted model, we can count events or cases and the linking of articles gives us a view of article use. In online communities, we can count posts and responses, which have a strong correlation to requests and responses. In the self-service model, we can count activity like searches and page views and sessions, but they don't have a one-to-one relationship to issues pursued and resolved. So, we have to infer things about the self-service experience from a number of different sources. And, just like in Performance Assessment, where the creation of value can not be directly counted, we find a triangulation model is very useful.

There are a number of things we want to measure about our self-service mechanism.

- User's view
  - What value is being realized by those who use self-service?
  - What is the experience of those who use it?
  - How often is self-service used before a case is opened?
  - How often are users of self-service finding things that are helpful?
- Internal view
  - What value is the organization realizing:
    - How much demand is being satisfied through self service success?
    - How much demand is being satisfied through self-service success that would have come to the assisted model (cost reduction)?
  - What is the pattern of article use - what articles are valuable to the users?
  - What impact is self-service having on the nature of the work that still comes into the assisted path (new versus known ratio)?

The Measures

Assessing the self-service experience and value relies on a combination of data analysis, user feedback, and observation.

Data analysis:

- User behavior patterns
  - Click stream analysis

- Volume variation

Direct user feedback:

- Surveys
- Comments and feedback from users

Observation:

- Usability tests

Because none of the self-service mentioned above are precise - that is, none of them by themselves directly represent the user experience, we have to look at them together using the triangulation concept. For the above measures, it is the trends that are most important, not the absolute value. And, it is our ability to correlate the different perspectives to gain confidence in our assessment of the user experience.

As we discussed in Technique 5.10: Content Health Indicators, we need a way to assess the value of the articles in the knowledge base as it grows. The three perspectives discussed in Assessing the Value of Articles are relevant here as well: frequency of reuse, frequency of reference, and the value of the collection of articles. The articles available through self-service should be included in value assessment.

Integrating Feedback

The most powerful and valuable feedback about KCS articles comes from the audience using them. Every time a user acknowledges getting value from a KCS article, that feedback should be visible to all who contributed to the KCS article: the creator, as well as people who reused and modified the article. If an end-user flags a KCS article as incomplete or confusing, that KCS article must be queued for rework.

In order to promote trust and to increase the credibility of the KCS articles, some organizations are making feedback visible to all audiences. A ranking system can be put in place similar to what Amazon.com does with product reviews, or Trip Advisor and Yelp provide for user reviews of hotels and restaurants. This information can feed into the triangulation model for assessing the self-service experience.

An underlying premise of KCS is "the best people to create and maintain the knowledge base are the people who use it every day." As organizations begin to Build Proficiency in KCS and make the majority of what they know available to users through a self-service model, that premise still holds.

This raises the question of how to engage users as part of the process. In fact, as organizations mature to the point where a large portion of their articles are external in a just-in-time manner (lots of KCS Publishers across the organization publishing in the moment), good user feedback mechanisms become critical. Users become part of the quality management process for KCS articles. Here are some of the ways member companies have implemented this when allowing users to comment on articles:

- Some make comments private and ask the user if they want to be contacted about the comment. If the user checks the "contact me" box, the system opens an incident for that customer and it goes into the normal incident handling process. This approach is probably feasible only for high complexity/low volume environments.

- Some make the comment public with a wiki-like section on each article that allows users to contribute their experience and opinions and see the comments of others
- Some allow trusted users (often identified through the community forums) to create and modify articles in the knowledge base. The source of the article or modification is indicated in the article.
- Some have segmented the knowledge base and have a governance model in place that allows all users to contribute to open-source type content.

Practice 6: Process Integration

Overview

The Process Integration Practice aims to make the Solve Loop frictionless. The goal is a tight integration of the Solve Loop practices, the incident management tool (CRM - or any tool that captures the request), and the knowledge management tool. This proves to be a daunting challenge. We strive for transparent integration of the request for assistance with the problem solving and knowledge creation and maintenance processes. What we describe in this section is the ideal. However, we know of no company that started with the infrastructure and integration described here; it is a goal, not the starting point. You do not have to have all the functionality or the level of integration we describe to start your KCS journey. Many have been successful with very crude integration. The key is to be constantly improving the infrastructure and moving toward the ideal state.

Tight integration is helpful so the responders can move quickly through the necessary functions with a minimum number of screens and clicks. The workflow is a real-time problem solving and knowledge capture process; the tools, navigation, and infrastructure used to support the process must be real-time as well. Once the initial workflow is defined, it must continually be improved as we gain experience in using it.

Techniques

- Technique 6.1: Structured Problem Solving
- Technique 6.2: Seamless Technology Integration
- Technique 6.3: Search Technology for KCS
- Technique 6.4: Closed Loop Feedback for the Whole System
- Technique 6.5: KCS Process Adherence Review

Technique 6.1: Structured Problem Solving

As we have mentioned, the Evolve Loop defines and encompasses the Solve Loop. The structured problem solving process puts some context around the practices of the Solve Loop. The Solve Loop is not intended to be a linear or sequential model. The practices operate as independent entities, and they are used as needed in responding to requests. The structured problem solving process provides direction on how to use the Solve Loop practices in an effective way.

In some respects, problem solving is an art. However, we have found that a little bit of structure in the problem solving process can help improve the outcome. The structure of the KCS article also helps reinforce an effective approach to problem solving.

> Consider a crime scene: the first thing the police do when a crime is reported is to preserve and record the situation. The first officers to arrive on the scene are trained to secure the area; they mark the location of the evidence and bodies and take pictures. When the detective shows up to solve the crime, they first seek to understand the situation, then begin to ask clarifying questions, and then eventually go off to do research.

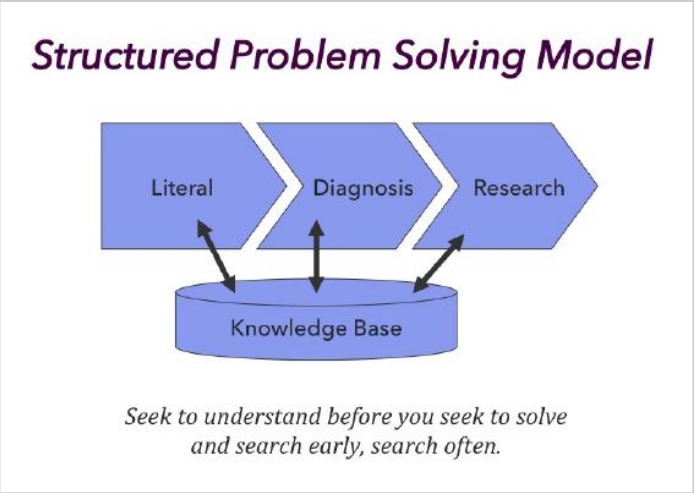
The structured problem solving process involves application of the four practices in the Solve Loop. It helps the responders collect, organize, and analyze the information used in solving the issue. Note that there are different skills used in different steps in the problem solving process, and, as a result, different responders or collaborators may be involved in each step.

Having explicit techniques in the workflow not only improves the problem solving process, but also creates a KCS article as a by-product of the problem solving process. The structured problem solving process in KCS includes two simple, yet powerful, concepts:

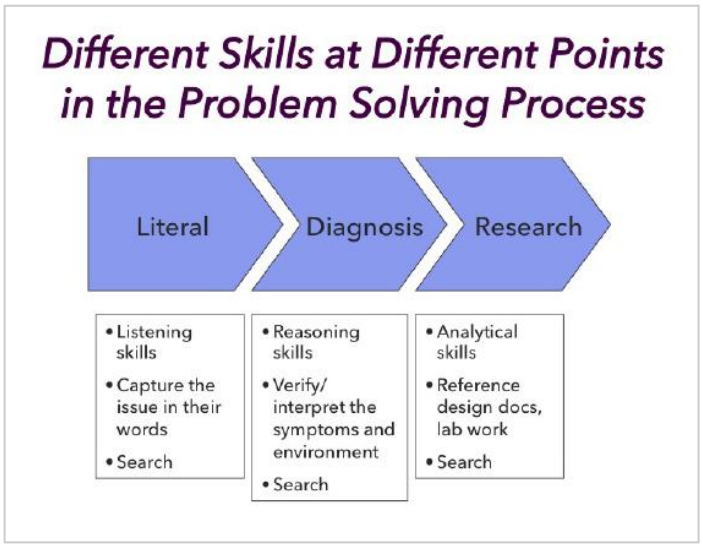
- Seek to understand before we seek to solve (a Core Concept)
- Search early, search often (a Solve Loop technique)

First, we seek to understand the situation in the requestor's context, and we capture it to preserve it. Then we seek to understand what we collectively know about the issue (search the knowledge base). These concepts are not unique to KCS; Charles Kepner and Benjamin Tregoe outline these same problem-solving methodologies in The Rational Manager, as does Stephen R. Covey in The 7 Habits of Highly Effective People.

Just as in the crime scene, we start by preserving the perspective the requestor has of what is happening. This is a very literal process. Next, we search the knowledge base to see if this is a known issue or to see what we collectively know about this type of situation. The "search early, search often" technique is a key part of the Solve Loop. If an article is found that provides the resolution, then we are mostly done (we still need to link and modify the existing article). If not, then we refine the search by collecting additional information from the requestor.



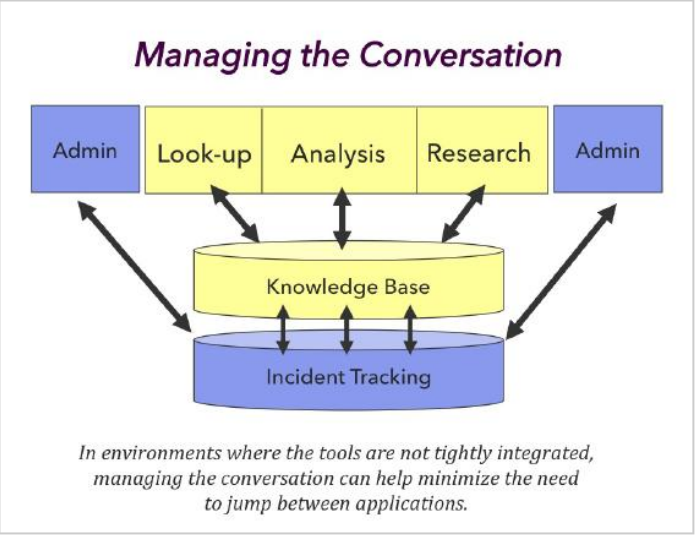
Searching will sometimes result in finding articles that describe similar situations. While perhaps not perfect for our situation, articles about similar issues can provide additional insight or trigger qualifying questions that we had not thought of. This complements what we know about analyzing this kind of issue. If an existing article is not found after refining our search a few times, we start the *diagnostic* process. We tap into our problem solving experience and use whatever tools are relevant. We continue to ask clarifying questions. As we build a richer understanding of the issue, we check the knowledge base frequently. If we do not find anything pertinent to the situation in the knowledge base, and we cannot resolve the problem, we then collaborate with others or escalate the issue for more additional *research*.



Many responders are too quick to move into the diagnosis phase of problem solving. If we move too quickly into diagnostics, we are likely to jump to conclusions, stop listening to the requestor, miss the fact that there is already a KCS article in the knowledge base, or miss the opportunity to learn from others' experiences in solving similar problems. KCS reinforces the need for the literal step to be the first step in the problem solving process.

Managing the Conversation

We are seeing better integration of the various systems the responders use to resolve issues. However, if systems are not integrated and we have to use multiple systems and screens to handle issues, this section is relevant. In environments where we need to use multiple applications to get the job done - for example, a case or incident management system that keeps track of the events, and a separate knowledge management system that houses the KCS article - it's helpful to design the KCS workflow to manage the conversation in order to minimize the need to jump back and forth between systems.



In environments where the tools are not tightly integrated, managing the conversation can help minimize the need to jump between applications.

Deal with the administrative elements at the beginning (contact initiation) and end of each contact (wrap up) - not interspersed throughout the resolution process. This approach will allow focus on the objective of problem solving.

Enabling Collaboration

Problem solving is a collaborative process. Ask any responder what they do when they realize they are working on something new or unfamiliar and they will tell you they reach out to their peers: they collaborate. All too often they do it i*n spite of* the traditional processes and escalation rules. What if our process and infrastructure facilitated collaboration instead of inhibited it?

Support Analysts have collaborated for years using tools like email and instant messenger or just asking others nearby: the "prairie dog" support model (over the cubicle wall). These are helpful but limited in their effectiveness. We are seeing some significant infrastructure improvements integrated into the responder user interface that facilitate collaboration.

The opportunity to improve the effectiveness of collaboration lies in our ability to know things like availability, who knows what, and who is interested in what. Effective collaboration, or what we call Intelligent Swarming, is a function of relevance. By relevance we mean that for a given issue, we want to bring together the best resources we have (people and/or content) to solve the issue. To accomplish this we have to know something about the issue and something about our resources, content, and people. Earlier versions of KCS focused on capturing the collective experience of the organization in a KCS article (content). What is emerging is the idea of people profiles that capture both the experiences and interests of the people.

Just as a search gives us access to the past experience of others through the KCS article, we could improve the relevance of collaboration by providing access to the people profiles. Where KCS helps connect people to content or knowledge for known issues, Intelligent Swarming helps connect people to people for new issues.

The Consortium members have been working for some time to bring the concept of Intelligent Swarming to operational reality. An increasing number of members have moved their organizations from an escalation-based model to a

collaboration-based model. They are realizing amazing benefits. For more information, see the Intelligent Swarming initiative on the Consortium web site.

We have learned some things from skills-based routing. Most organizations that have done it report mediocre results. The issue is that if the profiles are detailed enough to be helpful in getting an issue to the right person, they are difficult to create. If they are created, the dynamics of the environment make them impossible to maintain. On the other hand, if the skills profiles are at the level of detail where they are creatable and maintainable, they are not specific enough to be very accurate in routing.

We have come to the conclusion that the people profiles must be largely programmatic or maintained by the system and tunable by the people in order to reflect interests. The experiences of a Support Analyst, or any responder, change on a week-to-week basis.

Some operational examples of enabling collaboration:

- Collaboration capability built into the user interface
  - Simple version - launch instant messenger (without leaving the problem solving environment - see the prototype user interface on the next page)
  - Sophisticated version - finds relevant people based on the information captured in the incident or WIP article
- People finder capabilities
- Directed swarm - a team of people triage all incoming issues or a team of people work on any reported severity 1 issues. This takes the KCS concept of collective ownership of knowledge and applies it to incidents. A different view on incident ownership: distinguish ownership of response from ownership to solve. An individual is responsible to respond to the customer but the team owns resolution of the issue. (See the BMC case study on the KCS Academy Resources page.)
- Enabling visibility to all open incidents and filters that allow responders to see the incidents they might be able to solve or assist with. This enables an opt-in model; people choose to help.

Technique 6.2: Seamless Technology Integration

Technology is a critical enabler for KCS. It would be possible to follow the process of KCS using paper and pencil, but only if there were no more than two people engaged in the same room at the same time and they agreed on a filing system! Once there are more than two people participating, or we want to collaborate across different locations, we need technology. It supports the scaling of the processes to multiple people in multiple locations who are most likely working at different times. Technology and the KCS methodology allow us to collaborate independent of space and time.

KCS can be enabled with many different technologies. The Consortium has developed a KCS *Verified* program to help in the tool selection process. Software products become KCS *Verified* by demonstrating support for the following sample criteria:

- An article object and search engine
- Supports distinction between problem content and environment content
- Search engine granularity
  - Search problem content against problem content
  - Search environment content against environment content
- Ability to link/point/relate incidents to KCS articles and KCS articles to incidents
- KCS article audience management
- KCS article state categories
- Search arguments are preserved as the basis for a new KCS article
- Reporting and metrics

To find out which vendors' products are verified or to get a detailed list of all the requirements, visit the KCS Academy web site.

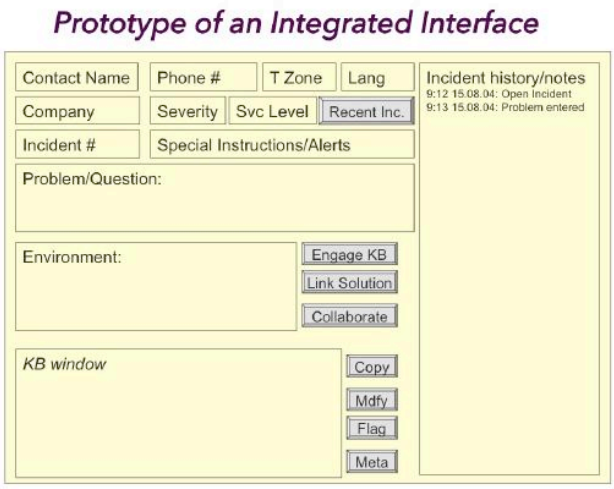
Integration of Workflow and Technology with CRM, IM, and Other Tools

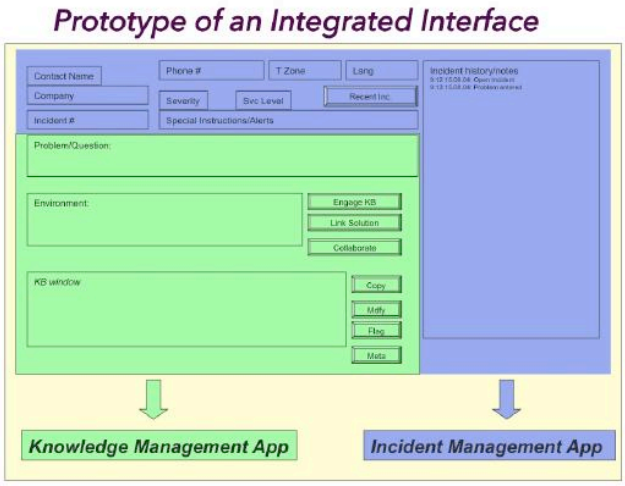
Ideally, technology enables the problem solving process in the Solve Loop at the speed of conversation, or real-time. Responders become more proficient at the process and solve problems faster by using the experiences of the entire organization and minimize rework.

To achieve this "speed of conversation" goal, the tools being used must be integrated to enable a seamless workflow where interaction with the knowledge base and KCS article creation are integrated with problem solving. For example, an ideal integration between a knowledge management system and an incident management system might include the following capabilities:

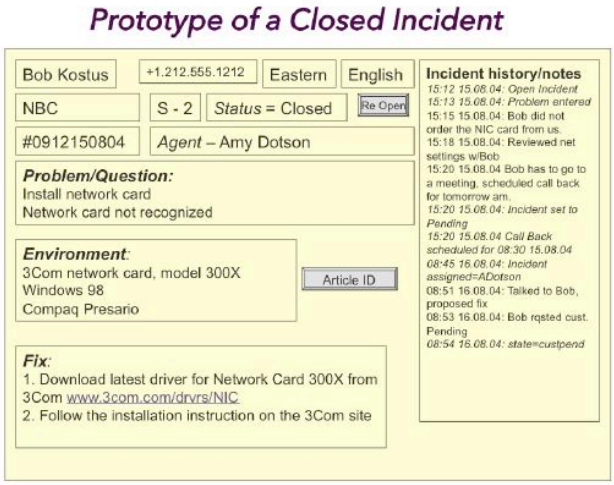
- Search the knowledge base using information in the incident record to launch or refine the search
- Link an existing KCS article to an incident and to retrieve information from the KCS article, such as the resolution, to populate the incident record. Linking can occur with two types of knowledge: reference information and resolution/fix information. Reference information is information found in reference materials such as service manuals or installation manuals. The specific sentence or paragraph must be findable by the search engine, the information must be accessible by the audience being served, it must be in the context of the audience being served, and it must be in a maintained repository.
- View a KCS article that has been linked to an incident, and vice versa
- Modify existing KCS articles in the process of reuse ("reuse is review") based on the KCS licensing model
- Create a KCS article in the knowledge base from information in the incident record
- Collaborate with the subject matter experts who are relevant to the problem and quickly contact them through email or chat

The following is a prototype of a Support Analyst's view (a knowledge worker's interface) to the system. The typical service environment we see has an application user interface that most often demands the users' behavior align to the needs of the application. For organizations in a position to adapt the user interface, we are instead proposing a user interface that aligns more directly to the needs of the user.



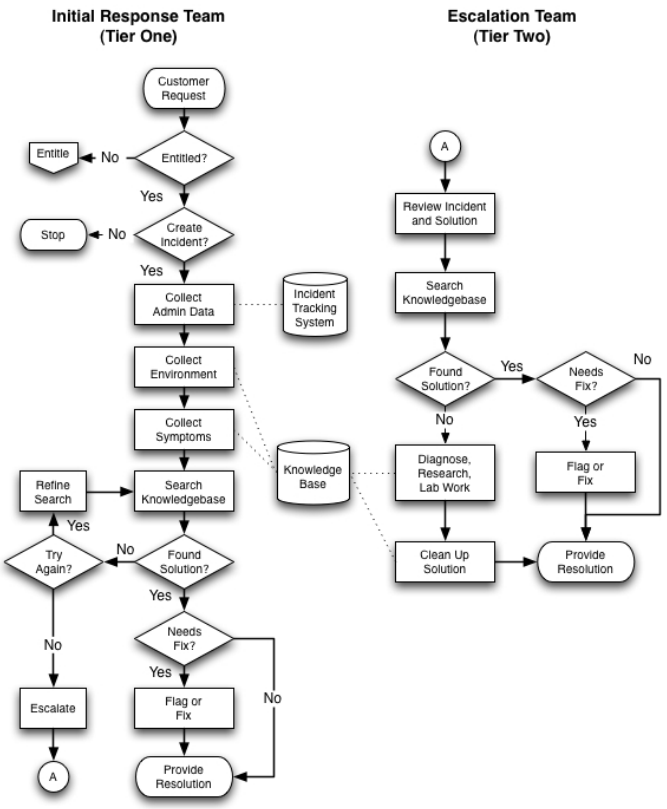


The closed incident captures the KCS article, in the form of Problem, Environment and Resolution, as it was given to the customer—a snapshot of the KCS article. The KCS article continues to evolve as it is reused.



No support organization that has adopted KCS has started with this "ideal" user interface. This level of integration should not be viewed as a requirement to start out; many have created great benefit with little or no integration between their incident management and knowledge management applications. However, sustainability of the KCS Practices requires that the users see continuous improvement in the level of integration. The KCS Coaches and Knowledge Domain Experts should provide requirements to the owners of the user interface to promote continuous improvement in the design and functionality of the infrastructure.

The workflow below shows how an individual resolution might unfold.



The workflow above is offered as an example; it is a place to start the design. Every organization will have variations. The workflow designed must consider the tools being used and the nature of the products and customers being supported.

The two key considerations in designing the workflow are complexity and volume. High complexity and low volume environments typically have longer resolution times and more frequent use of the knowledge base throughout the problem solving process. The workflow above is an example of a medium to high complexity environment. In a low complexity, high volume environment the average resolution times may be five or ten minutes and the level of redundant work is very high. That is, responders are resolving the same issue over and over again. In this environment the workflow described above does not make sense. Responders are not going to search for an answer in the knowledge base that they already know and use every day and can resolve in a matter of minutes. However, we still want to know how often issues are being resolved and we need a way to notify a responder if there is new information about the issues or resolution.

Organizations that deal with frequently used and widely known resolutions will often create a "quick click" list or a favorite list of articles. This is most effective when the list is unique to each user, although it can be done at a team level. The "quick click" feature enables the responder to record, with a single click, the fact that this issue was handled again. The click increments the use count for the article. This is critical information to capture as it is used in the Evolve Loop to identify patterns of reuse.

Also, clicking the article creates an opportunity to offer the responder the latest information about that article through a pop up window or by opening the article. This should only be done to communicate changes in what is known about the

issue and once the new information becomes widely known the "pop up" feature for that article should be turned off.

Design criteria:

- Eliminate the notion of separate tools: the integration of case management, knowledge management, and collaboration functionality.
- Make it easy and obvious for the responders to do the right thing
- Minimize context switching, screen changes/application changes - create "a single pane of glass" or a "one page interface" that has the functionality needed for the majority of the incidents (don't waste real estate on seldom used functionality).
- Align with and support the responders' problem solving process
- Capitalize on all that is known and already entered (no duplication of work)
- Integrate what is known from all/multiple sources: about the customer, the account, entitlement, the product and the problem/question

We can have the best tools and user interfaces in the world but... if we don't understand why and how to use them, it won't matter. A good user interface must be complimented with good measures (based on outcome and value creation - not activity), understanding and buy-in on the part of the responders, and coaching to support behavior change.

Five things that influence behavior (in no particular order):

- The tool: functionality, navigation, integration - make it really easy to do the right thing
- Measures: how are people measured
- Recognition and reputation: ego food
- Understanding: the extent to which the responders understand WIIFM (what's in it for me) as well as the bigger picture: what's in it for the company and the customers.
- Coaching: peers who are trusted change agents and role models

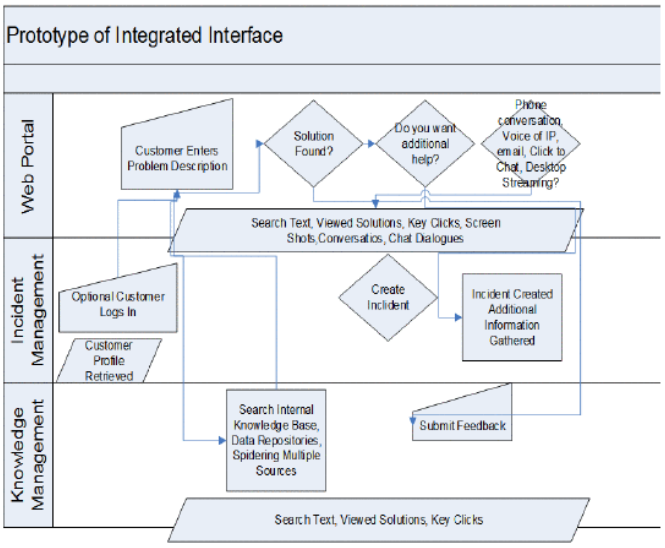
Again, the optimal environment has a tight integration between incident management and knowledge management systems such that users do not have to move between applications. However, most of the KCS adoptions have not had the luxury of starting with this kind of tight integration. Success has been achieved with very crude integration. Don't feel you cannot start on the KCS journey unless you have what is outlined here. This represents the ideal—something to aspire to as your adoption and technology infrastructure mature.

Integration With Web Portals and Online Processes

While there are compelling benefits from the adoption of KCS internally, even more value can be created by delivering articles directly to the users via self-service mechanisms. We have learned, however, that self-service users follow a different workflow. The self-service workflow takes advantage of easy access, the presence of online communities, and economies of scale.

Although the self-service workflow is different from the traditional call flow, the technology used for the portal must integrate seamlessly with and support the process of submitting a request for assistance to the support center (an incident). A key point in the design is the idea of "no dead ends." If a user starts the problem solving process using the

self-service knowledge base but does not find anything helpful, it is important to allow them to easily move from the self-service path to the assisted path. A "click to submit" or "click to chat" button should take them from self-service mode to the assisted mode and preserve the content of their earlier searches, including the documents they have reviewed. This seamless, logical transition improves the user's experience and encourages future use of the self-service path because it is the path of least resistance and best results.



Technique 6.3: Search Technology for KCS

Because searching is an integral part of the KCS Solve Loop, it's important to provide technology that allows users to search the knowledge base effectively. Rightly or wrongly, users often blame their search technology for the difficulty they have finding relevant content. If users aren't confident of search, they're less likely to seek to understand what we collectively know, less likely to review and improve content while using it, and more likely to capture duplicate knowledge.

Search engines are designed to return relevant content on the basis of a query. Search engines will sort the list of documents they return on the basis of how closely they calculate each returned document matches the query. When search works well, the most relevant documents will be at the top of the list of results.

In their simplest form, search engines look for literal matches between words in the query and words in the document. Search engines can be made more sophisticated, for example by matching simple variations on terms (for example, matching "run" with "running"), irregular variations (matching "run" with "ran"), synonyms (matching "run" with "jog"), or concepts ("run a program" matches "execute software" but not "a jogging fitness program").

Sorting by relevance, or ranking, is very important because users rarely look at more than the first several results (or, at most, the first several pages of results). So documents that are ranked low are effectively excluded from search results.

Relevance ranking can use many factors to assess the closeness of match between query and document. For example:

- How many of the query terms appear in the document
- How often those terms appear
- How rare or meaningful those words are in the documents being searched (e.g., "0x32565" is more unique than "Error," so the query "Error 0x32565" will be a closer match to "Code 0x32565" than to "Error -135")
- The proximity in which the words appear to each other
- The location of the words; for example, words in the title are presumed to be more meaningful than words buried in the text. The Consortium has proposed that good practice may be to rank matches in the Issue and Environment section higher than matches in the Resolution or Cause section, because the user is presumed to not yet know the Resolution or Cause.
- The closeness of match of concepts (not just the words themselves) contained in the query terms and documents
- The presumed quality or reputation of the document, based on link counts, ratings, age of the last view, or other similar factors

Though there are as many algorithms as there are vendors, search quality must be measured by the success users have navigating through the knowledge base.

It is important to understand how the search engine works, so trainers and Coaches can advise all knowledge

contributors and users on the best ways of using search. For example, should we use many words or few? Should we use sentences and natural language, or just keywords? How sensitive is search to specific words, or are general concepts sufficient? Coaches must be prepared to model, and provide feedback on, technology-specific aspects of search.

"Search" for Support: What's Different

The nature of human languages—and especially English—makes search challenging in any domain. For example, if we say "stock," are we asking about a financial instrument, part of a gun, or a soup base? And is "running in to the bank" a common errand, or a navigational error in a kayak? Humans unconsciously disambiguate competing meanings based on context, but context is hard to program into machines.

Internet search engines leverage the structure of the web itself, and the behavior of users, to increase relevance. With over 100 million websites and hundreds of millions of users searching every day, Internet search has an almost inconceivably large dataset to mine. Unfortunately, KCS knowledge bases have neither the web's structure nor its volume of use, so Internet search approaches don't work well for them. We often hear, "Can't search work just like Google?" Because organizational knowledge bases do not have the volume of activity, our answer is "no."

If search is hard in general, search for support is doubly so. Users know some symptoms of their problem, and they may know something about when and where the problem occurs, but they don't really know the answer they're looking for. This is the basis for the Consortium's contention that search should look first in Problem and Environment sections, at least for articles using the KCS proposed structure. The search technology also needs to support people who know something about the resolution or cause of an issue and allow them the option to search the Resolution and Cause fields.

The good news is: support domains are constrained. People will ask about anything in Internet search, but in KCS knowledge bases, they're typically asking about exceptions that occur with a defined set of products and services. This simplifies the "stock" problem, if technology knows how to take advantage of it.

Key Considerations for Search Technology

The sophistication of search technology required for a sustainable KCS implementation varies based on the size of our knowledge base, the complexity of the domain (i.e., how subtle can the nuances be between non-duplicate content), the technical astuteness and the persistence of our users. Generally speaking, very simple technology often suffices for a knowledge base of fewer than 1,000 basic articles, while collections over 100,000 articles in a deeply technical subject area strain the limits of current technology.

Here are some considerations for selecting search technology:

- Is it important to be able to search other resources at the same time as the knowledge base? In other words, should a single search return results from documentation, community forums, and defects?
- Will a simple keyword search suffice, or do we need to support synonyms or concept-based search? Does the size and complexity of our domain require even more advanced approaches to finding results?
- How much of a burden does the search technology impose on the content developer who is capturing, structuring,

and improving content? Must they enter careful metadata or keyword fields, or will search handle the content automatically? Can knowledge be captured "at the speed of conversation?"

- What reports are available to drive Evolve Loop content development, especially to fill self-service gaps?
- What options does the KCS program team, or another team, have to tune and refine the search experience? What reports are available to help them do this?

Planning for the Ongoing Effort of Search Tuning

Sophisticated search tools may deliver excellent experiences, and in some cases, they're the only way to sustain KCS. But they do require ongoing effort to maintain and tune. Since KCS content changes and evolves over time, so too must search.

Planning for this maintenance effort is a key component of the Process Integration practice in the Evolve Loop. Generally, a person on, or working in partnership with, the KCS program team, coordinating closely with knowledge developers, should be responsible for this ongoing end user experience optimization. Failure to plan for this task can turn a "smart" search tool into a dumb one, indeed.

The following tasks should be performed in an ongoing cycle:

Identify Search Experiences Weaknesses

Sources include:

- Informal conversations with knowledge developers
- Search analytics - looking for "no match found" queries
- A formal hill climbing process which evaluates the results of frequent requestor queries.

Take Action

- Is there a knowledge gap? Let a Knowledge Domain Expert know.
- Are multiple articles with different resolutions being returned for a set of symptoms? This is usually because the environment statements do not include the characteristics that distinguish one article from another. Use these as examples for the Coaches and KCS Publishers to highlight the importance of including the distinguishing characteristics in the article.
- Is content difficult to read, or not in the requestor's context? Diagnose why this isn't being fixed naturally in the Solve Loop, and take corrective action. Also, consider revising the search engine's dictionary or concept map to bridge the gap between different users' terminology.
- Are important or definitive articles not showing up at the top of results lists? Implement search tuning options such as "best bets," "managed answers," or other ways of making important (generally Evolve Loop) content more prominent in results.
- Are requestors struggling to troubleshoot using search results in particular important areas? Consider creating value-added Evolve Loop content such as multimedia, "active" content, or diagnostic KCS articles that link together in a resolution paths.

Evaluate the Effectiveness of Your Actions

- Make sure the initial problem has been corrected, using the same methods used to identify the problem in the first place.

Technique 6.4: Closed Loop Feedback for the Whole System

A major benefit of KCS is the opportunity to continuously improve the user's productivity and experience. KCS creates a system of persistent learning that is based on experience. Patterns and trends in the knowledge base can be used to drive documentation, product and service improvements. The Evolve Loop, a process of analysis and reflection, generates insight for the whole interaction network. The Knowledge Domain Expert extracts the learning from the patterns of information in the system.

Root Cause Analysis and Evolve Loop Content

Just as the healthcare industry has moved from reactive-only services to more proactive, preventive care over the last decade, many organizations have started to supplement reactive support with preventative actions—eliminating the source of issues in the first place. This has become possible as self-service has off-loaded responders' time, making them available to spend more time identifying issues for elimination. In order to find and diagnose problems, Knowledge Domain Experts perform root cause analysis. The patterns and trends of the articles in the knowledge base are the source of information for the analysis.

Knowledge articles are very transactional in nature. They represent what we have learned from an interaction. Looking at a collection of articles in a domain allows us to identify patterns and trends. We can assess the closeness or distance between articles. Articles that cluster around common themes or have similar causes represent opportunities to improve products or services. Removing the source of a frequent request is the ultimate level of success for an organization as it improves the customer productivity.

80% of the incident volume relates to 20% of the content in the knowledge base.

The Knowledge Domain Expert, product engineering, and product management must be part of the workflow and become engaged as patterns and trends in the Solve Loop content start to emerge. Through understanding the patterns and trends, we can pursue another form of intervention. Perhaps we can improve the documentation or create Evolve Loop articles - ones that merge the experience represented in many related KCS articles into a single KCS article. This idea is discussed further under the role of the Knowledge Domain Expert and in the Creating Evolve Loop Articles technique. (See KCS Roles and the Licensing Model for a complete description of Knowledge Domain Expert responsibilities.)

Continuous Improvement of the Workflow and Content Standard

The closed loop nature of the workflow makes it easier to monitor and maintain its effectiveness. This is done by continually sharing best practices. Gathering the best practice feedback must be part of the continuous improvement process. Because KCS articles are effective for procedural information, many organizations use their knowledge base to collect information about the KCS workflow or the content standard. Questions and issues about the workflow, the tools, or the content standard are captured as articles in the knowledge base (often marked as internal use only).

For example, if we are seeing many incomplete or duplicate articles, then we most likely have a workflow problem. The most common cause of duplicate KCS articles is a breakdown in the problem solving process; people are not searching before they create (search early, search often). Here again it is the role of the Coaches and Knowledge Domain Experts to investigate these kinds of issues and work to improve the level of understanding of the KCS practices across the organization. We should also mention, relative to this example, that management can be the source of unwanted behaviors by putting the wrong metrics and goals in place (more on this in the Performance Assessment section), or by not revoking the KCS license when an individual's performance slips below acceptable levels. A system of continuous improvement is critical to sustain the practices and must involve the Coaches, Knowledge Domain Experts, and managers.

Early in the adoption phase of KCS, the Coaches play a critical role in refining the workflow. As the adoption matures, the organization needs to establish a way to manage a continuous improvement process. This is often done through a KCS Council (see Performance Assessment for descriptions of these roles and responsibilities).

This continuous improvement can affect both the process and the content quality itself. For example, a software company may ask a Knowledge Domain Expert to focus on using knowledge management tools to monitor search strings, KCS articles found and considered, and the incident record to understand the KCS article process. With this insight, the Knowledge Domain Expert will be able to create high value Evolve Loop content.

Variations on a Theme

Process is important, but be smart about it. Success with KCS requires an understanding of the concepts such that the practices can be tuned to meet the needs of a specific environment. For example, what if we have an environment where we experience an extremely high frequency of a few problems or questions? Articles for these problems exist in the knowledge base. Normally we would expect those articles would make their way quickly to the web due to the high reuse.

But what if the environment is such that web delivery is not an option, or just not ready yet? In this case, does the full process of search, capture, and structure make sense for these few frequently raised issues? Since the responders answer these issues many times a day, they know the answer, and they know it is in the knowledge base. It does not make sense to force the standard KCS workflow. In this situation, the standard workflow offers no value.

For these frequently raised issues, we should enable an easy way for the responders to record that they answered this question again. This shortcut is often a "quick click," or a favorites list that allows them to quickly record the reuse of the KCS article. For high volume issues the important thing is to capture how often the KCS article is being used, so the

"quick click" button needs to drive the reuse counter for that KCS article.

This list of frequently provided answers may be unique to any group of responders. There should not be more than ten items in this list. Also, our processes should include ways to update the responders if the preferred answer changes over time. For example, if a better answer evolves, when the user hits the "quick click" button for that KCS article, the new information displays to update the responder.

Technique 6.5: KCS Process Adherence Review

Integrating use of the knowledge base into the knowledge workers' workflow is a Core Concept of KCS. The frequency and quality of our Solve Loop activities dictates the level of benefit we will realize. The Process Adherence Review (PAR - formerly called Process Integration Indicators or PII) enable us to assess how often and how well we follow the workflow. PAR is most valuable as a tool to promote learning and growth: to identify coaching moments. It provides insight to people's behaviors and the degree to which the Solve Loop activities have become a habit for knowledge workers. Most of the Process Adherence Review items are activities, and we do not want to put goals on them.

The activities or behaviors that create value require judgment. The Solve Loop activities must be done in the context of the bigger picture: an understanding of the long term potential value of what we are doing. Goals on activities drive behavior in the absence of judgment and will corrupt the knowledge base. **Do not put goals on activities!**

In order to promote the behaviors that create value we need to understand how the activities relate to the outcomes. This is a primary focus of the KCS Coach. While observation is an important element of effective coaching, there are a number of indicators that can help the coach identify areas for learning and growth. The combination of the Content Standard Checklist (formerly Article Quality Index or AQI) and the Process Adherence Review (PAR) provide coaches with the perspective they need to help people improve the value they create.

The Process Adherence Review checklist needs to be tailored to the workflow the organization has defined. To provide an example of these indicators we will use the generic workflow that is defined in the Process Integration Technique 6.1.

Search Indicators

Search early, search often. It is interesting to know how often and when we are searching the knowledge base. This is especially true early in the adoption of KCS as people are learning the KCS workflow. Here again, observation by a coach of knowledge workers doing their work and how they are using the knowledge base is important. A helpful complement to observation is having reports on search activity. The timing of the first search compared to case open and the frequency of searches done while responding to the request can be helpful information to inform the coaching activity. Searching is an activity, so don't put goals on it! The reports should be used to help identify coaching opportunities about the knowledge worker's behavior. The conversation should be about the behaviors, not the search indicators.

Contribution Indicators

Knowledge workers contribute to the health and value of the knowledge base when they:

- Reuse and link accurately
- Modify when appropriate
- Create if an article doesn't exist

We want to know how often we are capturing the experience and learning from our interactions. Our contribution shows up in a couple of different ways. If we reuse an existing article, we want to record the use of the article by linking. If we learned additional information about a known issue, we want to add that to the existing article: modify. And, if no article exists that reflects the request and resolution, we want to create one. Reuse (linking), modify, and create all represent the primary ways we contribute.

Reuse and link: How we record reuse will vary from organization to organization. In environments that have cases or incidents that reflect the interaction, we want to link the resolution article to the case or incident. In environments that do not use a case or incident, we need to provide a way to record the reuse of articles. Analyzing the patterns of article reuse helps us identify opportunities for high impact business improvements in the Evolve Loop. Link rate is the percentage of closed cases with an article linked. Link rate is an activity: do not put a goal on link rate! Link accuracy is the percent of articles linked that are actionable, specific, and relevant to the case. Link accuracy is an outcome (not an activity), so it should have a goal.

Link rate and accuracy are both helpful indicators. While linking is a good habit to develop across the organization, the reason we care about link rate is because the pattern of article reuse can drive business improvements in the Evolve Loop. It is sufficient to have a link rate of 60%-80%. A linking rate of 60% or greater will allow us to determine the pattern. Driving the linking rate to 90% will not change the pattern. For most organizations, 60% is well above the threshold needed to determine the pattern. The same is not true for link accuracy. If our link accuracy is below 90%, the pattern of reuse of articles is not likely to tell us anything useful. Some member organizations give zero credit for link rate if a knowledge worker's link accuracy is not 90% or higher. Maintaining a high link quality is far more important that maintaining a high link rate.

Modify rate: As we reuse articles, we often have additional information that could improve what already exists. It may be information that can improve the clarity or usefulness of an article. More often, it is additional context we can add, based on how the requestor experienced or described the issue. Adding this to an existing article will improve article findability. The modify rate reflects the percent of the time we modified an existing article when appropriate. Divide the number of articles that have been modified by the number of articles reused where a modification was warranted. Here again, calculating the total articles reused where a modification was appropriate can be difficult and a sampling technique can used to assess the modify rate.

Create rate: What percentage of the time are we creating new articles when appropriate. Similar to above, the math is to divide the number of times an article was created (subtracting any duplicates) by the opportunity to create.

Contribution Index

The contribution index tells us: of all the opportunities where linking, modifying, or creating was appropriate, what percentage of the time did we link, modify, or create? It is the ratio of the number of times we contributed as a percentage of our total opportunity to contribute. Or more simply, how often did we do the right thing? So, the math for

the contribution index is the number of times we linked, modified, and created divided by the total opportunity where link, modify, or create was appropriate. One important thing to consider in this calculation is duplicates. We want to subtract any articles that are duplicates from the number of articles created. If we can easily find a preexisting article that should have been reused, then creating an article was not appropriate and should not be included in the article creation count. The links to a duplicate article should not be counted in the relevant link count either. Assessing the total opportunity to contribute is difficult. However, just like the Content Standard Checklist process, we can assess it based on a sampling technique.

In defining the contribution index we have described the ultimate model for assessing contribution. Members are at various states of maturity with this model.

These concepts were pioneered by David Kay, KCS Certified Trainer and Consortium Innovator in his work on the Resolution Quality Index (RQI) in developing an indicator for "how often did we do the right thing."

Sample— Mature Process Adherence Review (PAR) check list

		Activity								Outcome	
Knowledge Worker	Requests Reviewed	# KB Potential	# Articles Linked	Link Rate	# Articles Modified	# Modify Opportunity	Articles Created	# Create Opportunity	Contribution Index	Relevant Link	Link Accuracy
Al	18	16	15	94%	9	10	1	4	83%	11	73%
Beth	15	12	11	92%	9	9	1	1	95%	11	100%
Chuck	16	14	14	100%	8	10	1	5	79%	10	71%
Hector	15	13	12	92%	7	8	3	3	92%	11	92%
Grace	18	12	9	75%	3	5	2	5	64%	6	67%
Joe	15	15	14	93%	4	10	1	8	58%	7	50%
Kim	16	15	14	93%	0	7	2	2	67%	5	36%
Group Process Adherence Review											
	113	97	89	92%	40	59	11	28	76%	61	69%

Note: Articles created excludes duplicates

Relevant Link = Articles linked-(Articles Created-Create Candidates)-Existing not Linked

Practice 7: Performance Assessment

Overview

We have talked about the need to shift the culture and values of the organization to align with the desired behaviors of collaboration and shared ownership. Culture and values are reflected in what the organization measures.

> If we look at how people's contributions are measured, the teamwork factor is traditionally overshadowed by individual metrics.

There are often disconnects between the stated values of an organization and their metrics. For example, most organizations talk about valuing teamwork, yet, if we look at how people's contributions are measured, the teamwork factor is overshadowed by individual metrics. There is often a hero mentality in the organization that rewards people for what they know and the fires they fight (whether or not those fires should have been avoided proactively with knowledge and product improvements). These same organizations will also practice "stack ranking" of employees (an exercise of ranking employees against one another), which is a simplistic, linear mindset that promotes competition, not collaboration.

Our traditional management practices for performance assessment are at odds with the KCS practices. They have taught us what does not work. Our management practices have evolved over the years from a manufacturing model. In manufacturing, we create tangible products like toasters or TVs or cars. When the output is tangible, we can count it discretely. The activity of individuals on the production line is directly linked to the outcome. "How many levers I put on the toasters" is directly related to how many toasters came off the line. With KCS, however, we create knowledge, relationships, and loyalty. These are intangible outcomes, and they cannot be counted discretely. Activity in an environment of intangible outcomes is only loosely related to value.

A key lesson learned from our members' experience with performance measures is.... if we put goals on activities (such as article creation, modify, or linking), it will corrupt the knowledge base.

We know a lot about measures and goals that do not work ... so what do we measure?

This section will provide an understanding of performance assessment techniques:

- KCS roles and the licensing model
- The concept of adoption phases and how the measurement system must evolve from phase to phase
- How to assess who is creating value through triangulation

- A scenario to show the value of some of the key measures, with example reports
- A list of measures, their definitions, and their uses

KCS introduces new roles and responsibilities for the organization that emphasize collaboration, sharing, using, and improving our *collective* knowledge rather than *individual* knowledge. Rather than rewarding people for what they know, these roles help to emphasize learning and collaboration as well as contribution to organizational knowledge. Retooling the way individual and team contribution are assessed reinforces these new roles and the behaviors that will create a healthy and valuable knowledge base.

The concept of assessing value through a process of triangulation is based on the work of Kaplan and Norton and their book The Balanced Scorecard. From this multi-perspective view, we have identified the key measures and reports and provide a scenario to show how the measures are used.

Another characteristic of KCS is a strong emphasis on providing leadership versus traditional day-to-day management. Leadership means taking responsibility for organizational alignment and motivating the team through performance drivers (more on this in the Leadership and Communication section). Leaders play a pivotal role in:

- Defining and setting goals for the desired outcomes (not activities)
- Ensuring the tools and infrastructure are integrated and align with the workflow, making it obvious and easy to do the right thing
- Providing knowledge workers with visibility to performance indicators so that they can take responsibility for managing their performance
- Providing knowledge workers with visibility to the impact of their contribution

For this system to work, performance measurements must be clearly linked to the strategic objectives of the organization. The team should understand these objectives. The leaders can then encourage participation in the KCS processes and recognize both the individual and team accomplishments in the creation of value.

Techniques

- Technique 7.1: KCS Roles and the Licensing Model
- Technique 7.2: Coaching for Success
- Technique 7.3: Assessing the Creation of Value

Technique 7.1: KCS Roles and the Licensing Model

A Licensing Metaphor

We have been describing two major efforts: learning to capture and structure in the workflow, and practicing the structured problem solving process (as described in the Workflow section). For most responders, these represent significant changes in behavior and activities. If we use a driving metaphor, adopting KCS is like learning to drive a car. When we first started, we got a learner's permit, and we had to have a licensed driver along as a coach. Initially, we had a great deal to think about and driving took all of our attention. We had to watch the road, other cars, our speed, and the rear view mirror. Most of us learned rapidly how to deal with all these dynamics and went on to pass a test to get our licenses.

Earning a driver's license gave us new rights and privileges and enabled us to drive on our own. For most of us, this represented a significant increase in our independence with a corresponding increase in our parents' anxiety. Our parents worried about the risk and whether we would make the right decisions. Along with this increase in our autonomy came an increase in our responsibility and the need for auto insurance. Eventually, with enough experience, driving became second nature to us, to the point where we now (unwisely) attempt to do other things while we are driving, like eating or talking on the phone.

Adopting KCS is like learning to drive. People learn how to do the KCS processes as they are resolving requests. Like learning to drive, we can be taught the concepts in a classroom setting, but only with practice can we internalize the behaviors and competency so they become second nature.

The KCS licensing model defines system rights and privileges for each role: KCS Candidate, KCS Contributor, and KCS Publisher. As with driving, some people will choose to progress farther, study more, and gain more advanced skills. Performance assessment must include guidance on how, when, and whether to advance. As with driving, each license is earned based on demonstrated consistent behaviors that align with the KCS role. For example, a provisional license is issued after the knowledge worker attends a class and demonstrates a basic understanding of the KCS practices and techniques. The KCS Contributor and KCS Publisher competency levels are achieved after consistently demonstrating the respective incremental competencies defined in those roles. Many organizations also require a coach's recommendation to move from one level to the next. As with a license to drive a car, an knowledge worker who frequently breaks the rules or demonstrates poor judgment should lose their license.

Note that the levels of KCS competency are **not** linked or related to the levels, job roles, or positions in an organization. In a support organization, each level of support (tier 1, tier 2, tier 3) should have a mix of analysts with a range of KCS competencies.

The KCS Council

Perhaps the most difficult hurdle in adopting the KCS methodology requires changing the culture of the organization to a truly collaborative, knowledge-sharing environment. KCS identifies some critical roles to support the organization in this transition. The roles help redefine the way knowledge is created, valued, and shared. Additional details can be found in the KCS v6 Adoption & Transformation Guide.

The critical roles for KCS are:

LEADERS—managers must become leaders. They must define the vision of what success looks like at their level of the organization, and then support the knowledge workers in deciding how the work should be done (workflow) and defining the standards for findable and usable KCS articles (content standard).

KNOWLEDGE WORKERS —Anyone responding to an interaction or request are knowledge workers.

- KCS Candidate
- KCS Contributor
- KCS Publisher

KCS COACHES—change agents and KCS Practices experts who support the development of the KCS competencies and the proficiency development of knowledge workers from KCS Candidate to KCS Publisher. Generally, a peer working part time as a Coach—a "player coach."

KNOWLEDGE DOMAIN EXPERTS—responsible for identifying Evolve Loop content based on KCS articles created in the Solve Loop workflow, look after the health of the knowledge base, usually focused on a collection or domain of content, has both technical expertise in the domain and profound understanding of KCS processes.

Evolution of the KCS Council

Over time the adoption activities diminish and the KCS Council shifts its focus to continuous improvement through sustaining and improvement activities. As the organization gains experience with KCS, there will be a need to tune the KCS processes based on that experience. This review and enhancement works best through a KCS Council that meets on a bi-weekly basis to discuss issues and improvements. The KCS Council is a cross functional group with global representation, which continues to include the KCS Coaches, the Knowledge Domain Experts, and representatives from management. The KCS Council provides the forum for continuous improvement to the content standard, the workflow, tool functionality and integration, and the feedback and reporting systems. This critical continuous improvement sustains and optimizes KCS benefits.

Distribution of Roles in the Organization

There are two dimensions to consider when thinking about the number and type of roles for an organization:

- The nature of the work being performed, including things like complexity, volume, and rate of redundancy in incoming requests.
- The maturity of the existing KCS processes in the organization.

In a start-up environment, the majority of knowledge workers have Candidate or Contributor licenses, there is a Coach to knowledge worker ratio of 1:5 to 1:8, and there probably is not enough content to warrant a Knowledge Domain Expert.

In a mature environment, the Coach to knowledge worker ratio usually evolves to something like 1:50, and knowledge domains evolve to the point where a Knowledge Domain Expert has sufficient content volume to look at patterns and trends.

Knowledge workers at the KCS levels reside in each level and role in the organization. It is important to distinguish between KCS competency and technical depth—there is not a 1:1 correlation.

For example, each support tier should have members at the KCS Contributor and KCS Publisher levels of competency that can create and validate KCS articles for the problems solved at their level. As the KCS processes mature in the organization, all but new knowledge workers in training should be at least at a KCS Contributor level.

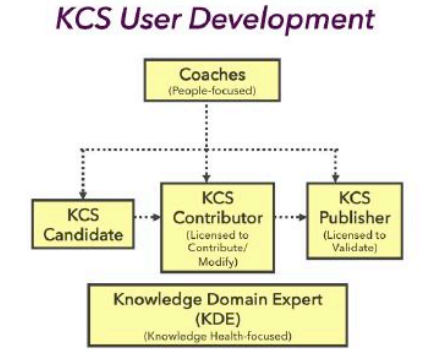
The ultimate goal for high complexity environments is to get the majority of the knowledge workers to a KCS Publisher level. For low complexity environments the goal is to have enough KCS Publishers in the organization such that at any point in time we have zero articles in the queue waiting to get published. Or stated another way, if we have articles that are identified as externally-usable waiting to get published, then we don't have enough Publishers.

Evolution of the KCS Roles

The licensing model is one important part of the quality assurance model for KCS. The organization must monitor the quality of the work being done and be willing to revoke the KCS license if the quality of work slips below an acceptable level (see the Content Standard Checklist section in Content Health for more on this).

The KCS user development diagram below shows the typical evolution path from role to role. Not everyone is appropriate for or interested in taking the step to the next role. Anyone an organization would trust to come up with a new answer for a customer should, in time, become a KCS Contributor. Use the descriptions and characteristics detailed in the next section to help identify the right people to move along the KCS path.

KCS Roles for the Team: Detailed Responsibilities and Qualifications



KCS knowledge and competency defines an individual's rights and privileges in the system

The following section provides guidance on the type of knowledge, the skills, and, in some cases, the personality traits that are necessary for success with KCS. The Consortium and its partners offer training specifically geared to acquiring the skills for these different roles. Additional details can be found in the KCS v6 Adoption & Transformation Guide.

The KCS Council is responsible for defining the roles and responsibilities as a part of building the foundation before the pilot. This includes updating job or role descriptions as well as defining the expected competencies for each KCS role within the KCS Competency Model (i.e. KCS Candidate, KCS Contributor, etc.) They may also modify these as a result of analyzing the pilot or during early adoption. Once KCS has been implemented and the KCS Council takes on the responsibility for evolving the maturity of the KCS Practices, they may identify the need to update the competency model and the competencies for specific roles. Normally they would propose the changes for management approval.

KCS Candidate

The KCS Candidate understands the basics of KCS and knows how to interact with the knowledge base in a way that captures their experience and capitalizes on the collective experience of the organization. A KCS Candidate must be able to recognize relevant information in the knowledge base and exercise judgment in their interaction with it. They should not use or deliver a KCS article that they do not understand. Since articles in the knowledge base are created with a specific audience in mind, dictating vocabulary and level of technical content, a Candidate adapts knowledge to suit the profile of the target audience.

Upon completion of training (often web-based training or an element of new hire training), the KCS Candidate should:

- Understand the structured problem solving process
- Accurately and consistently capture the requestor's context in the workflow
- Search for and find existing KCS articles
- Review and either link or flag articles in the problem solving workflow
- Modify their own KCS articles
- Frame new KCS articles (Work in Progress or Not Validated) which will be reviewed or finished by a KCS Contributor or KCS Coach

KCS Contributor

The KCS Contributor reviews (as they reuse) or finishes KCS articles that are framed by themselves or others, making

sure the articles adhere to the content standard. The KCS Contributor has the capability and authority to create or validate articles in their product area without review by a Coach. They may also author and approve articles for broad audience visibility. They may directly improve articles that have article audience set to Internal and should flag articles in an External state that need to be updated or improved.

While the KCS Candidate creates articles that are Work in Progress or in a Not Validated state, the KCS Contributor can create content that is in a Validated state. A Validated article implies a high degree of confidence in both the technical accuracy and compliance with the content standard. A KCS Contributor can put articles that are in a Work in Progress or Not Validated state into a Validated state if in his or her judgment the article is "sufficient to solve."

The KCS Contributor competencies are incremental to those of KCS Candidate and involve a detailed understanding of the importance of the context of the audience, the content standard, the Content Standard Checklist, and the KCS processes. They should be able to work independently by creating well-structured KCS articles and be adept at enhancing others' articles to make them visible to a wider audience. The KCS Contributor should also be able to demonstrate understanding by passing an exam.

KCS Publisher

The KCS Publisher is authorized to set the article audience to External or publish content to an external audience, typically on the web, as well as to modify externally-facing content. In KCS environments, "publish" means making the KCS article visible to partners or customers. Compared to a KCS Contributor, the KCS Publisher takes a more global, outward view of the audience and the content. The KCS Publisher knows the technical implications of the knowledge being published, has an understanding of what material is priority information, and has an understanding of copyright and trademark policies enforced by his organization. The KCS Publisher is also responsible for understanding the external audience and publishing requirements outlined in the content standard. Because External KCS articles may be linked from other websites and may be visible to a large audience, the KCS Publisher must exercise good judgment about modifying External articles.

In determining readiness to move to the KCS Publisher level, consider that the KCS Publisher should be consistently adhering to the Content Standard Checklist, following the KCS workflow (measured by the Process Adherence Review), and have consistently positive feedback on and high reuse of article content. They should reliably focus on the success of the team and the customer over individual success.

The KCS Publisher may flag External content for archival or deletion, but because removing externally-facing content from the web is an activity with difficult-to-assess implications, typically the KCS Publisher can't personally archive or delete.

As KCS matures in the organization, a high percentage of the knowledge workers should be at the KCS Publisher level. This percentage allows the just-in-time publishing of content that drives a high level of customer success with web-based self-help. This is especially important in order not to create a backlog of flagged External content, because KCS Contributors may not directly edit External articles. The KCS Publisher should also be able to demonstrate proficiency by passing an exam.

The KCS License

The KCS Contributor and KCS Publisher roles need to have a well-defined path to achieve those levels. The knowledge worker should have to demonstrate proficiency at each level and pass an exam, much like drivers need to take a written exam and also pass a behind the wheel test. And like the driver's license metaphor, the KCS licenses should be renewed on a regular schedule.

A KCS license isn't the end of the licensing process, just like getting a license to drive isn't the end of that process. Drivers must obey the rules of the road and demonstrate good judgment, and so must KCS license holders. If a KCS Contributor consistently shows poor judgment or a lack of compliance with the content standard, they should lose their license.

Variations on KCS Licensing

While both the coaching model and the KCS licensing model are common components of successful KCS adoptions, we should note that there is considerable variation in how companies have implemented the KCS licensing model. Some have rigorous criteria and tests that knowledge workers must pass, while others rely solely on Coach recommendations. Some organizations require an annual renewal and some issue the KCS license for life.

Organizations also use a variety of combinations of levels of license. Some combine the responsibilities and competencies of the licensing levels we have outlined here. Following are a few of the variations we have seen. They all work; the variations reflect the level of trust the leadership has in the knowledge workers.

- Two-level model where the KCS Candidate and KCS Contributor rights are combined and KCS Publisher rights are distinct. Knowledge workers can create and modify Not Validated and Validated/Internal articles and, once competent, are licensed to publish External.
- Two-level model with a KCS Candidate and the KCS Contributor and KCS Publisher rights are combined. Knowledge workers have very limited rights in the system while they are learning KCS (Work in Progress and Not Validated articles only) and when they are competent they are licensed at the KCS Publisher level.
- One-level model where everyone has all the rights and privileges of the KCS Publisher and people lose their license if they consistently demonstrate poor judgment or a lack of compliance with the content standard.

KCS Coach

Coaches are critical change agents in the KCS adoption process, invaluable in helping knowledge workers develop their KCS competencies. In the KCS environment, the Coach is successful when people are moving from KCS Candidate to KCS Contributor or KCS Publisher. Although a KCS Publisher needs very little coaching, Coaches should be doing periodic quality checks on their articles.

The Coach's focus should first be on evolving an individual's KCS skills, then over time, shift to developing team capabilities. Although organizations recognize the need for training, they often overlook the need for effective coaching. An investment in training becomes largely wasted without the follow-up provided by a Coach's on-the-job reinforcement and support. This is especially true with KCS, which requires knowledge workers to develop and foster a set of new work habits, not just skills. An effective coaching program will shorten adoption time. In fact, the benefits the

organization will achieve are directly proportional to the time they invest in coaching.

Coach responsibilities include:

- Promote user skill development through effective skills coaching.
- Help the KCS Candidate understand the problem solving workflow and how the KCS article management process is integrated with the thinking process.
- Influence knowledge workers to practice good knowledge management.
- Influence knowledge workers to apply standards for creating and improving knowledge within the knowledge base.
- Review KCS articles framed by the KCS Candidate until they reach required levels of competency.
- Perform internal validation of KCS articles to ensure accuracy for the described context and adherence to the quality standards set by the organizational unit.
- Provide ongoing feedback to knowledge workers and management about organizational KCS skill development.
- Provide feedback to the knowledge developing organization, within the defined processes, to improve KCS article management.
- Develop and monitor their own coaching skills through work with head Coaches.
- Participate in the KCS Council.

The benefits realized by the organization are directly proportional to the time invested in coaching.

The Coach must have a profound knowledge of the KCS Practices and processes as well as strong communication and influence skills. We have found it most effective to have Coaches be part-time KCS Coaches and part-time in the role of the peers they are coaching. A few organizations have tried full-time Coaches and have found that the Coaches quickly lose touch with the reality of issue resolution. As a result, the Coaches lose credibility with those they are coaching. A good rule of thumb is for Coaches to split their time equally between handling requests and KCS coaching. For more on coaching, see Technique 7.2: Coaching for Success.

Knowledge Domain Expert

As the organization matures in its use of KCS, a fourth important role evolves: the Knowledge Domain Expert (KDE). KDEs are the ones who do or facilitate the Knowledge Domain Analysis (KDA) that is defined in the Content Health Technique 5.4: Creating Evolve Loop Articles. This critical role is responsible for identifying high value articles, identifying and driving improvements in products, documentation, processes, and policies, and contributing to improvements in the workflow and content standard.

The Knowledge Domain Expert must have both technical depth in their area of responsibility and a profound understanding of KCS. The KDE looks after the health of a collection or domain of knowledge, usually a subset of the knowledge base that aligns with their general expertise. To help achieve business objectives, the Knowledge Domain Expert drives the value of the knowledge by paying attention to both the quality of KCS articles and the effectiveness of the workflow that produces the articles. The Knowledge Domain Expert assists colleagues in the collection, storage, and distribution of knowledge within and outside the organization. He or she will help determine what knowledge is important for the organizational memory and help to ensure that mechanisms exist for assessing the patterns that emerge from the content.

The Knowledge Domain Expert works closely with the Coaches and teams who have direct responsibility for maintaining

the quality and flow of content as well as owners of the products, documentation, processes, and policies. This role is instrumental in the maintenance of a coordinated worldwide team effort. The Knowledge Domain Expert also contributes input toward process automation to push information externally. Their success is measured by the impact they have on:

- Effectiveness of reuse analysis and pattern identification
- Minimizing duplicates
- Improving article utilization (internal and external)
- Increasing customer success with self-service for their domain
- Influencing the owners of products, documentation, processes, and policies
- Identifying improvements in workflow and content standard
- Identifying and supporting specific corrective actions in conjunction with coaches

Most organizations have multiple knowledge domains, depending on the variety and granularity of the products and services being supported. Knowledge domains are virtual collections of KCS articles about a product family, a function, or relating to a technology or group of technologies. Knowledge domains are seldom about one product. They are not precise or absolute in their boundaries; knowledge domains often overlap. A knowledge domain is the collection of content that makes sense to look at for pattern and clustering analysis. Therefore, the purpose or intent of the analysis defines the collection of articles that is relevant.

For each domain, one or more subject matter experts emerge as Knowledge Domain Experts—knowledge workers with enthusiasm for the technology or function and the KCS principles and practices. They are usually experts who continue to have the same responsibilities, but take on additional responsibilities for the overall health of the knowledge and success of the team. They are often excited about being able to provide development with actionable information based on a broader view of customer experience. Knowledge Domain Experts often become experts in the knowledge base tool being used and develop an understanding of the subtleties of the search technology.

The reporting structure for this role can be designed in several ways. Consider the focus of the Knowledge Domain Expert's role—that of creating organizational value through externalization of content outside the organization. They will work closely with product development and product management. You may want to consider filling this role through a cross-organizational position.

Responsibilities include:

- Ensure efficient and effective problem solving by the team.
- Apply expertise in data mining to perform trend analysis and find the significant patterns in the data.
- Assist in the fundamental development and maintenance of knowledge base quality and flow, including the knowledge base quality methodology, article standards, and process guidelines.
- Perform Known vs. New Analysis
- Develop and analyze reports on key metrics for business value of the knowledge base, such as article reuse rates, web-enabled call avoidance, and improvements to resolution times.
- Ensure effective knowledge base operations by monitoring related information (organizational effectiveness, resource allocation, new article creation trends) and making recommendations to management to accommodate changing conditions.
- Advocate for changes necessary to maintain the knowledge base as an effective tool for achieving business

objectives.

- Provide input for items that have a worldwide impact. For example, monitoring and defining the KCS article metadata, prioritizing enhancement requests, coordinating training efforts where feasible, and planning for upgrades and systems integration enhancements.
- Influence the owners of products, documentation, processes, and policies to make improvements
- Participate in the KCS Council.

Technique 7.2: Coaching for Success

Benefits of Coaching

Studies show classroom or online training alone will increase productivity by 22%. Training combined with coaching results in a 86% increase in productivity. The average ROI is *six times* the cost of coaching, and leads to consistent, replicable bottom-line results. Consortium members who have invested in training are more successful than those who compromise the training program by not making it a priority in the organization or not giving the coaches the time to coach. For more on the role of coaching, see Technique 7.1: KCS Roles and Licensing Model.

The Coach's focus should first be on evolving an individual's KCS skills, then, over time, shift to developing team capabilities. Although organizations recognize the need for training, they often overlook the need for coaching. An investment in training becomes largely wasted without the follow-up provided by a Coach's on-the-job reinforcement and support. This is especially true with KCS, which requires knowledge workers to develop and foster a set of new work habits, not just skills. An effective coaching program will shorten adoption time. In fact, the benefits the organization will achieve are directly proportional to the time they invest in coaching.

The Coach must have a profound knowledge of the KCS principles and processes as well as strong communication and influence skills. We have found it most effective to have Coaches be part-time KCS Coaches and part-time in the role of the peers they are coaching. A few organizations have tried full-time Coaches and have found that the Coaches quickly lose touch with the reality of issue resolution. As a result, the Coaches lose credibility with those they are coaching. A good rule of thumb is for Coaches to split their time equally between handling requests and KCS coaching.

The intent of coaching is to develop individual habits of proficiency and team performance, not simply to ensure the correctness of KCS article content. The quality of content is promoted through the development of individual proficiencies.

KCS Coach Competencies

Effective coaching relies on:

- A thorough understanding of the KCS Practices
- Ability to articulate the why we are doing KCS and what's in it for the knowledge worker
- Understanding of support processes and tools
- Inquiry, Advocacy, Appreciation and Reflection
- Excellent communication skills, particularly in the following areas:

    - Listening skills, seek to understand
    - Explaining and describing concepts
    - Providing feedback
    - Influencing to generate results
- Mindfulness of feelings
- Demonstrated ability to:
    - Manage time effectively
    - Ability to identify coaching moments - use data and measures to help others become more proficient
    - Appropriate communication with management
    - Deal with objections like (see Objection Handling in Technique 8.7: Communication is the Key)
        - Can't capture in the workflow
        - Don't have time to create articles
        - "Dumbing down" my job
        - Giving away my value
- Demonstrated commitment to the success of team members

Using Organizational Network Analysis to Select Coaches

Selecting the right Coaches is a critical component to the success of KCS. The goal of the Coach is to increase the competencies of others, not to showcase their own skills or expertise on a subject. The Coach is:

> *"A trusted role model advisor, wise person, friend, Mensch, steward, or guide - a person who works with emerging human and organizational forces to tap new energy and purpose, to shape new visions and plans, and to generate desired results. A coach is someone trained and devoted to guiding others into increased competence, commitment, and confidence."*
>
> - Frederic Hudson

Selecting the wrong Coaches can lead to the following symptoms:

- Inconsistent participation among groups or geographies
- KCS articles that are not findable
- Duplicate articles
- Bottlenecks getting articles published
- Poor quality articles
- A great start to a KCS deployment, followed by a downturn in activity

Many organizations have made the mistake of picking the subject matter experts, technical leads, or documentation editors as the Coaches without considering social skills. Oracle, Novell, Microsoft, Quest, and Openwave have used

Organizational Network Analysis (ONA, sometimes called Social Network Analysis or SNA) to view their organizations' trust network in order to gain insight into who to select as Coaches, to identify collaborators in the organization, to validate Coach selections that have already been made, and to diagnose the cause of inconsistent KCS results.

Organizational Network Analysis is a mathematical and visual analysis of relations, flows, and influences between people, groups, and organizations. The nodes in the network represent people and the lines represent connections through various social behaviors. ONA input data can be gathered through surveys, behaviors or analysis of electronic communications (social media, email, etc.). The members of the Consortium collected data through the use of surveys.

Sample Survey Questions to Select Coaches:

- I would be more effective in my job if I could interact more with this person(s): _____.
- Whom do you go to for technical advice or problem solving?
- Whom do you go to for non-technical advice; process or policy information or general issues?
- Whom do you go to explore new ideas?
- Whom do you trust to keep your best interests in mind?
- If you don't know who to go to.... whom do you contact to find out? (who knows who knows?)

Upon the completion of the survey the nodes (or people selected in the survey) are input into the ONA tool. The output of a tool in a relationship map and several measures, including:

DEGREE CENTRALITY - Number of connections a node has, more is not necessarily better, we want to connect the otherwise unconnected

BETWEENNESS - Connection between groups (broker), high degrees on betweenness could indicate single points of failure

NETWORK CENTRALIZATION - Less centralized networks have no single points of failure

NETWORK REACH - These measures have proven effective not only to select Coaches but also to find who knows what faster, or to find connectors when companies are merging, built innovation teams and learning communities and to support partners and alliance

Technique 2: Capture the Customer's Context

With KCS, we seek to understand the customer's experience *before* solving. This understanding requires information about both the perceived issue (in the customer's words) and the environment in which the problem is occurring. By capturing the customer's perception of the situation in the first interaction, we dramatically improve the findability and relevance of the KCS article to other customers who might have the same or similar perception of this issues in the future (context is as important as content).

Capturing the customer's context – their words and phrases – improves findability for other customers.

The objective of capturing KCS article elements is to "frame" the customer's situation—to capture their need, perception, experience, and relevant aspects of their environment as input to the problem-solving process. The objective is to use this input to improve the KCS article set. We use the framing information to either create a new KCS article if one doesn't already exist or to improve an existing article by reflecting another customer's experience.

Even if the customer's perception of the situation proves to be wrong, preserving it will improve the findability of this article for other customers. The Support Analyst's job is to capture with precision and accuracy the relevant environment factors, the resolution, and the cause in the article, *not to correct* the customer's description of the problem. However, Analysts may capture additional technical details to more precisely characterize the problem. Technical accuracy in the problem statements is not required. Of course, technical accuracy is critical in the environment, resolution and cause statements.

Some of today's knowledge tools enable the capture of the customer's online search activity before they contact us for help. Having the words and phrases the customer used to search is very useful - this is their context. The customer's search terms can be used to create the problem statements in a new KCS article, if one doesn't already exist, or can be used to modify an existing article and improve its findability.

Every Search Matters

In the case of incidents submitted via a self-service model, good things can happen if we capture the customer self-

service activities (search strings, articles viewed, and in some situations product and version information). First of all, this guarantees that we are capturing the customer context, and secondly, making this information available to the Support Analyst helps the customer feel like the effort spent pursuing a resolution through self-service was not a waste of time (no dead ends). This information can help expedite the problem-solving process as we can review what the customer has already done.

Customer submission of issues via self-service also helps us create new or modify existing articles. If we need to create a new article for this issue, the search words and phrases the customer used to look for a resolution are valuable content for the new article. If, as the KCS practices suggest, we have captured the information the customer used to search the web in a Work-in-Progress (WIP) article as we were working the issue, we have already created a new article and simply need to review the environment statements and update the resolution field. Because the problem statements came from the customer search activity they are in the customer's context.

If, in working on the issue, we eventually find an article in the knowledge base, we should improve that article so customers can find it. We have already captured the information in the Work-in-Progress (WIP) article, which we can use to update and improve the findability of the existing article. If the existing article was available to the customer but not findable, we use the customer experience (and their context) to improve the findability of that article. If the existing article was not available to the customer, can we publish it? Are we licensed to publish, are we confident in the resolution, and is the article compliant with the content standard? See the Content Health section for more on article states and transitions.

Practice 8: Leadership and Communication

Overview

The adoption of KCS is transformational and requires strong leadership. Understanding and communicating the relevance of KCS to the organization—how KCS contributes to the organizational goals and what it creates for all the stakeholders—is critical.

To do this, we start by developing an organizational vision (which we define as a compelling purpose that people can relate to), a mission statement, and a brand promise. Next, we create a strategic framework that links the benefits of KCS to the higher-level goals of the organization, which align with and support the vision. With the vision and framework in mind, a leader helps their team focus on *what* people need to accomplish, the right objectives and goals, and supports them in figuring out the *how*.

KCS requires strong leadership, not strong management.

The knowledge workers should own the content standard and the workflow while leadership focuses on the strategic framework, communications to promote understanding and buy in, and the performance assessment model.

The next level of detail involves developing and articulating the WIIFM—what's in it for me—for each of the stakeholder groups. One of the most compelling things about the KCS methodology is that it is wholly beneficial. With proper adoption, KCS benefits all the stakeholders. No one is compromised. Thinking through the WIIFM for each audience and likely objections is an important part of leadership readiness.

Another element of the leadership model is the reward and recognition program. Most leadership teams need to rework their programs to align with the knowledge sharing, collaboration, and collective ownership themes of KCS. We want to recognize people for their contribution, not reward them. Embracing what really motivates knowledge workers requires a shift in thinking for most organizations.

However, having all the elements of the leadership model described above is of little value in the absence of an effective communication plan. We have asked leaders whose organizations have adopted KCS what, in hindsight, they would have done differently. Most say they would have communicated more about why they are doing KCS.

Effective leaders create a healthy culture that encourages participation, individual commitment, and accountability. People need to understand their role in the context of the bigger picture in order to contribute fully. Through a well thought-out communications plan, clear role definition, and a performance assessment model that rewards the creation

of value, knowledge-centered organizations realize increased levels of capacity, capability, and loyalty.

In this section, we describe techniques to help leaders:

- Define the vision, including:
  - A compelling purpose - a simple value proposition
  - A mission statement - our approach to achieving the purpose
  - The brand promise - attributes of the relationship with those we serve
  - Explicit organizational values - acceptable behavior for interacting with peers and those we serve in order to achieve the purpose
- Define organizational metrics and goals that support the vision
- Capitalize on the inherent motivation factors in KCS
- Nurture people in the collaborative environment
- Gain buy-in and support for the KCS initiative
- Communicate, communicate, communicate

Leadership is Fractal

When we talk about strong leadership, it must exist at every level of the organization. One of the biggest challenges of a successful KCS adoption is the transition of the first and second line managers to leaders. KCS represents a bigger change for the line managers than it does for the knowledge workers.

Fractals occur in nature. For example, the pattern of a snowflake is the same at a magnification of 1X (the shape of the whole flake) as it is at a magnification 10X or 100X looking at one small section of the flake. That is, the pattern repeats itself at each level of magnification. Its shape is made up of many smaller shapes just like it, which in turn are made up of many smaller shapes just like it... and so on. "Leadership is fractal" means that at each level of the organization the understanding of what the organization is trying to accomplish is identical (compelling purpose, mission, values, and brand promise). This must be true in order to realize the vision. As people throughout the organization make judgments on how to handle each task or interaction, the vision is the basis or criteria for those judgments. For example, it is each interaction and at the same time the aggregate of all the interactions that reinforces a brand promise... or destroys it. If people at different places in the organization have different views or interpretations of the vision, it creates dissonance. Creating the vision and ensuring it is understood by all is the challenge of Executive Leadership. Helping the line managers become leaders who live and promote that vision is not a trivial task.

In discussing leadership, we talk a great deal about metrics. It can be easy to spend more time with spreadsheets and dashboards than with people. Leaders need to define and understand these metrics, but their primary focus should be on creating an environment of success for employees. Knowledge management is primarily about people, their understanding, buy-in, and behavior. That is the focus of KCS leadership. The measures are a tool for learning and growth; they should trigger conversations about behavior, not about the numbers.

We have learned a lot about the role of leadership for KCS success and have identified the following Leadership and Communications techniques.

Techniques

- Technique 8.1: Develop and Communicate a Vision
- Technique 8.2: Create a Strategic Framework
- Technique 8.3: KCS Benefits and ROI
- Technique 8.4: Promote Teamwork
- Technique 8.5: Tap into Internal Motivators
- Technique 8.6: Recognition Programs
- Technique 8.7: Communication is the Key
- Technique 8.8: Leadership Accountability to the Knowledge Worker
- Technique 8.9: Leadership and Communication Indicators

Technique 8.1: Develop and Communicate a Vision

> *"Knowledge is personal... Business, generally, is not."*

People are much more likely to participate in a knowledge practice if they believe in the vision and purpose of the organization. Knowledge, our life's experiences, represents a large part of who we are as individuals; it is personal. Organizations that have a compelling purpose, one that people can connect with on an emotional level, have a stronger foundation for employee contribution of knowledge than those that do not.

Vision and Motivation - What Motivates People?

We have learned a lot about what motivates people in a knowledge-centered environment. It is not sticks and carrots or rewards and punishment. The foundation (or prerequisite) is alignment to a purpose. If we care about the value proposition of the organization and we care about the people we work with, it creates the foundation for feeling good about our contribution: a sense of accomplishment is a powerful motivator. We only feel a sense of accomplishment in doing things that we care about.

As organizations become excited about KCS the question often comes up.... "How do we incent people to use or contribute to the knowledge base?" There is compelling research on what motivates knowledge work and it is not about tangible incentives. In fact, tangible rewards can be extremely disruptive to what we are trying to accomplish. For more details on this important topic see Herzberg's HBR article "One More Time, How do we Motivate our Employees" and Daniel Pink's excellent book Drive: The Surprising Truth About What Motivates Us.

Creating a Vision

A vision can take on many forms. We see four common elements to an effective vision and define the elements as:

- A compelling purpose - a simple value proposition (a phrase, 1-5 words)
- A mission statement - our approach to achieving the purpose (a paragraph)
- Explicit values - defines acceptable behavior in achieving the purpose (a list)
- The brand promise - describes the attributes of our relationship with those we serve (a list)

Some organizations engage the employees in helping to create the vision. This promotes understanding and buy-in, but if we engage people across the organization, we have to be willing to use and act on their input. In other organizations the elements of the vision are developed by an executive committee. We have seen both be effective. The important point is that we have a compelling purpose, mission statement, explicit values, and brand promise and that there is a common understanding of them at all levels of the organization.

Engaged with a Purpose

A compelling purpose is a simple phrase that describes our value proposition. Alignment to a purpose is a result of understanding and caring. The purpose has to be something we care about, something we have a connection to. Motivation comes from a corresponding belief that one's actions will make a difference in achieving that purpose.

The purpose is what we are about. The values are the definition of acceptable behaviors in accomplishing the purpose. Our brand promise is the attributes of our relationship with those we serve.

How do we get there? To begin with, leaders have to have a strong sense of ownership and personal commitment to the compelling purpose, values and brand promise. An effective leader's enthusiasm for the purpose, values, and brand promise becomes contagious. Two key factors will make the difference: sincerity and consistency. People have an instinctive sense about leadership's integrity. That sense is reinforced or disrupted according to the consistency of the leaders' behavior with respect to the stated purpose, values and brand promise. Behavior that is consistent will resonate and create trust, buy-in, and engagement. Behavior that is inconsistent creates mistrust, dissonance, and disengagement.

People are inspired when they believe in what they are doing and feel good about their individual contribution and the contribution of the team. A powerful purpose has an emotional appeal. For example, if we ask the Support Analysts at VeriSign what VeriSign's purpose is, they will quickly respond, "Trust on the Internet." They feel a part of something that they value, it has meaning to them, and they are proud to be a part of it.

It is amazing how many employees do not know their company's purpose. It is also surprising how many companies have a purpose that is in no way compelling. What makes the difference?

A compelling purpose:

- Is known by all
- Is bigger than the company itself, not self-referencing
- Is brief, clear, concise
- Elicits an emotional response
- Is a value proposition

Some examples of compelling purposes:

- Trust on the Internet.—VeriSign
- Saving lives, one person at a time.—Sanofi
- We create happiness.—Disney

Two examples of non-compelling purposes:

- "To create the best video monitor in the industry"—this statement is self-referencing (not bigger than self), limiting, and does not have a strong emotional appeal.
- "To create wealth for the shareholders" —not much emotional appeal here either.

Equally important is understanding and buy in to the brand promise. The brand promise is reinforced or destroyed at each touch point with those we serve. Everyone must understand and buy-in to the brand promise as that must influence the knowledge workers' judgment in each interaction. Delivering on the brand promise means doing a lot of little things right. A strong brand emerges when the aggregate of those things reinforce the desired attributes of the relationship.

What *about* money? Producing a profit for the company owners or stockholders is a responsibility of the business in a for-profit model.

Profitability is a by-product of being good at delivering on a compelling purpose.

Understanding the Mission

The mission statement is a paragraph that describes what we do, how we do it, and who we do it for. While the mission statement typically does not have the emotional appeal that a compelling purpose has it must align with and support the compelling purpose. The mission statement is mostly about us and the compelling purpose is about the value proposition we create for others.

Aligned to the Values

Our values are the rules of conduct: the guiding principles for our activities and behaviors in achieving the purpose. For example Cisco's values are:

- Focus intensely on customers,
- Make innovation happen,
- Win together,
- Respect and care for each other,
- Always do the right thing.

Living the Brand Promise

When we think about brand we often think about a company name or logo. A brand promise is different from branding which defines the use of our logo and colors and fonts. A brand promise is the list of attributes that describe our relationship with those we serve (our customers). A brand promise is often declared by the marketing organization. In reality our brand promise is what those we serve say about the relationship they have with us. And, what they say is based on their interactions with us over time. If we have an intent with respect to what we would like those we serve to say about us, we are more likely to interact in a way that promotes those attributes.

For example the brand promise at Sage North America is:

- Peace of mind
- More control

- Smart and efficient
- Be there
- Enjoyable

Sage developed this list of what they want their customers to say about Sage by talking with their customers. These are the attributes of the relationship Sage customers want in the relationship.

Having a brand promise becomes a valuable touchstone for leadership decisions at all levels of the organization. It is equally important that knowledge workers understand and believe in the brand promise. By living the brand promise in each of our interactions both internally and externally we will create the desired image.

Summary

The importance of engagement with the purpose, mission, values and brand promise may seem remote to a knowledge worker responding to requests for assistance. However, as we make the transition to KCS, we will be asking people to change how they do their work and to exercise an increasing amount of judgment in what they do. The degree to which individuals understand the bigger picture not only gives them a basis on which to make good decisions, it encourages participation, gives them a sense of belonging, and enables them to feel good about their contribution and accomplishments. It is the foundation for what motivates us.

Leadership's role is critical in ensuring the elements of the vision exist and that they are well understood by all. Equally important is leadership's integrity with respect to the vision. Leaders must model the values and live the brand promise. The fastest way to render the vision impotent and negate all the ripple effect benefits is for leadership to live by different standards than what they espouse and to make decisions that do not align with the vision.

Technique 8.2: Create a Strategic Framework

The strategic framework is a simple yet powerful document that links the benefits of KCS to the goals of the organization. Sadly, many organizations don't pay enough attention to building and maintaining the strategic framework even though it only takes a small team an hour or two to develop. It is often passed over as "not that necessary." We cannot over-emphasize the value of creating one. It is a small investment with huge benefits. Having a strategic framework is helpful because it:

- forms the basis for the communication plan
- is a critical tool for gaining executive support
- can help sustain KCS focus across "executive turn over"

In this section, we will describe the framework and provide an example. Note that, while the document is important, equally important is the exercise of creating the strategic framework. The conversation and understanding that come from the process of creating the framework are extremely valuable for the KCS Council.

Link KCS Benefits to Organizational Goals

The strategic framework can improve the organization's engagement, helping people see the link between the KCS practices and benefits to the organization's top-level goals. These, in turn, cascade down to departmental objectives and justify the outcomes that team leaders use to build their balanced scorecards (see Performance Assessment for more information on balanced scorecards). This continuity in goals—from the big picture to the team to an individual is critical. The ability of the knowledge worker to see the link between their goals and the organization's is a key success factor in deploying KCS, sustaining the benefits, and promoting understanding and confidence across the organization.

The strategic framework is the foundation for a successful KCS adoption because it provides context for the three primary stakeholders: the requestors (customers), the knowledge workers and the organization. The framework enables us to talk with executives on their terms by linking the KCS benefits to the top-level organizational objectives. When it comes to communicating with team members and others, a strategic framework provides the central messaging document.

Here is an example of a strategic framework expressed in terms of the primary stakeholders (requestor, knowledge worker, and the organization) and the specific contributions of the KCS program:

Sample Strategic Framework

Customers (Requestors)

Goal	Approach	KCS Contribution	Measurements
Reduce Customer Effort	Develop Agents, Share tribal knowledge	Access to known knowledge faster, and creation of new quicker	AHT, FCR, VOC
Support as Trusted Partner	Reduce support errors to increase accuracy of answers	Greater access to knowledge faster, sharing tribal knowledge	Q/A Support Errors, VOC, FCR, NPR
Implement Self Service	Create customer self-service	Enable self-service of knowledge from the context of the customer	Future

Employees (Knowledge Workers)

Goal	Approach	KCS Contribution	Measurements
Reduce Employee Turnover	Recognition, Autonomy, Trust, Engagement	Greater findability of articles, sharing known knowledge, autonomy to write articles, and ownership of the knowledge	Employee Turnover, Conversion Rate
Improve Employee Engagement	Autonomy, Development, Trust	Ability to write own articles, ownership of knowledge base	EOS
Increase Employee Competency	Development, Trust, Engagement	Access to knowledge base, training, certification, licensing writing articles	Q/A Scores, Assist Errors

Business (Organization)

Goal	Approach	KCS Contribution	Measurements
Reduce Average Handle Time	Upskill Agents, share knowledge	Access to known knowledge faster, and creation of new quicker	AHT
Manage Cost	Reduce the time it takes agents to solve	Faster resolution time, lower turnover rate, reduced time to proficiency	Cost per contact
Increase First Call Resolution	Increase engagement with knowledge	Provide known knowledge quicker, accelerate the creation of new knowledge through allowing everyone to contribute	FCR

Here is how the process works. One of the most common organizational goals is customer loyalty. A big driver for this is the support experience and time to resolve (TTR). So, we need a support level metric in the requestor loyalty section.

Leaders should select the measures and goals that link to requestor success (the desired outcome), not activities. This is the same message that we deliver for individuals, but the metrics are implemented at a group or organizational level. Some good examples of higher-level metrics that are only measured at the group level include metrics about requestor's frequency of use and success with self-service, as well as internal metrics like time to publish to self-service. If we measure the wrong things or fail to balance them well, we are likely to drive the wrong behaviors and wrong results.

Knowledge worker loyalty and engagement is another important factor. The usual practices of employee loyalty, buy-in and job satisfaction surveys, as well as measures for turnover and attrition, fit in here.

The operational efficiency, productivity, and capacity are also typical goals. We have noted the KCS factors that influence costs and contribute to profitability. However, pure cost reduction is a going-out-of-business strategy. The discussion should be about increased capacity and identifying ways to increase revenue, instead of cost reduction. (See the ROI section that follows for more detail.)

In considering organizational goals and the right outcomes to emphasize, we need to consider all the support paths our requestors use, like assisted support, self-service, and community-based support. How will KCS enable or impact these different paths? Which metrics can we capture? How can KCS reduce requirements for expensive channels (assisted) and increase the use of inexpensive channels like self-service or communities? Most organizations will need to expand the scope of their measurement model and add measures for things like the percentage of our requestors who use self-service first, and what is their success rate with that? Baseline measures for these kinds of things allow us to forecast the benefits and assess our progress.

Keeping in mind our organization's objectives, here are KCS-oriented measures to consider:

Process and knowledge health:

- Link rate (%)
- Article create vs reuse ratio (%)
- Size of the knowledge base (total number of KCS articles)

- Articles visible through self-service (% of total visible externally)
- Knowledge base maturity (calculated percentage)
- New vs known ratio (%)
- Average time to relief/resolution for known problems
- Average time to relief/resolution for new problems

Since the goal of KCS is to increase and leverage the value of the knowledge, this concept has many potential metrics:

Requestor Loyalty Drivers:

- Average time from request resolved to publish an article (to the requestor audiance, in minutes)
- Percentage of requestor-consumable KCS articles visible to customers
- Percentage of requestor success on the web (survey data)

Cost Management Drivers:

- Percentage of new vs. known issues being reported to the support center (assisted model) (manager's goal—focus resources on solving new problems, not known problems, drive down costs)
- Percentage of requestors who use the web first (survey data)
- Percentage of requestor success on the web (survey data)
- Number of product, policy or process improvements recommended to the respective owners
- Percentage of improvements accepted

In this document, we are focusing on the KCS-relevant metrics. They do not replace existing management metrics, but should be integrated into them.

Maintaining Balance

In many organizations, there is tension between the goals of requestor loyalty and operational efficiency. Improvements in one can starve the other. Typically, executives set goals for profitability and loyalty, but leave teams to determine implementation goals. Unlike traditional approaches to efficiency that can jeopardize customer satisfaction and loyalty, KCS demands a good strategic framework that reflects a healthy balance of these two goals. In fact, by implementing KCS to improve knowledge sharing, improvements such as self-service, proactive notification, and "treating known as known" can help organizations achieve both goals simultaneously.

For more information about measures in a KCS environment please see the Measurement Matters paper.

Technique 8.3: KCS Benefits and ROI

Operational Benefits and Return on Investment Considerations

Since most organizations need to invest money to adopt KCS, and we all live in a cost-sensitive climate, a normal organizational metric is return on investment (ROI). In assessing the operational benefits from KCS, we use the same stakeholders we defined above: the customer, the employees, and the business. Each will benefit differently from the adoption of KCS. The stakeholder benefits will be realized at three levels as the KCS practices become mature:

- *Direct*—operational improvements that are near term (3-9 months)
- *Applied*—new ways of delivering service and support enabled by KCS—for example, using knowledge that is created to power web-based self-service (6-18 months)
- *Leveraged*—the knowledge and the capacity created by KCS enables new kinds of value-added support services to be offered (12-30 months)

We will focus on the investment and benefits of the *Direct* and *Applied* levels of KCS as these represent the most credible near term benefits. Experience has shown that while KCS is valuable across a wide range of environments, the investment required and the benefits achieved from KCS vary based on the characteristics of the environment. The key factor in determining if KCS will produce value for the stakeholders is the degree to which knowledge or experience plays a role in responding to user's or customer's requests for assistance.

Additionally, as we look to measure the benefits, we have to consider both quantitative (objective) and qualitative (subjective) metrics.

Investment

Implementation of KCS requires investments in the following areas:

- Leadership and management—Sponsorship, program resources, and leadership development (training and coaching)
- Infrastructure and support—Modification of existing tools or the acquisition and implementation of new tools as well as the integration of those tools with existing systems
- Learning and growth—Training and coaching for knowledge workers
- Implementation and development of new roles and skills in the organization: Coaches, KDEs (Knowledge Domain Experts) and the knowledge domain analysis capabilities which should include access to Data Scientist
- KCS article management—The processes and resources for the Evolve Loop activities

Investments in these areas occur over the four phases of KCS adoption. These are:

- Plan and Design: Building the foundation elements like the content standard, workflow, and communications plan
- Adopt in Waves: Running Wave I to test the foundation elements
- Build Proficiency: Institutionalizing the practices across the whole organization
- Optimize and Innovate: Maximizing the benefits from self-service and improvements in process, policy, products, and services enabled by the Evolve Loop

Together, these areas and phases make up the total cost of ownership. Calculating the total investment can be difficult because many of the resources used are not incremental to the organization, in fact most are done with a redeployment of existing resources. Other elements, the incremental costs, are easier to identify. Each organization will have to calculate the non-incremental costs they wish to include. Once we have the total cost, we can divide it by the number of knowledge workers to figure the KCS investment per person.

One of the most challenging cost elements to estimate is coaching. During the startup of each wave of adoption, coaching represents a serious investment in capacity. In most organizations, no real cost is incurred because the organization simply lives with the reduced capacity. Other organizations that are under intense pressure may not be able to suffer through even a few months of decreased capacity and must back-fill knowledge workers during the coaching phase. Leaders must make a clear statement of their plans for allocating time to coach and their willingness to support coaching and perhaps compromise service levels in the short term for a dramatic long term gain.

Return on Investment

As we mentioned earlier, the return or benefits from KCS will be different for each of the stakeholders. The benefits are numerous, following is a summary of the benefits by stakeholder:

The organization:

- Increased capacity
- Organizational learning
- Improvement in the organizational image or brand

The knowledge workers:

- More interesting work (a high percentage of new issues, lower percentage of known)
- Increase in confidence, lower stress
- Increase capability and scope

The requestors (customers):

- Improvement in speed, accuracy, and consistency of responses
- Reduce the effort of finding resolutions (self-service)
- Increased loyalty

Sample KCS ROIs

Following are examples of the KCS ROI in three different environments. As we mentioned earlier, we must consider many factors in assessing the ROI. The complexity of the work most closely relates to the level of benefit. The following three examples are based on real organizations, and the actual results have been validated against this model. This ROI is solely based on the *Direct* benefits: the improvement in the operational efficiency of the group. It does not include the *Applied* benefits, such as self-service.

Examples of KCS ROI	**Complexity**		
	High	**Med**	**Low**
Average Minutes To Resolve Known Problem	10	7	3
Average Minutes To Resolve New Problem	90	40	15
Average Incidents Closed Per Day	4	12	44
Estimated % of Problems Which Are New	50%	35%	20%
Monthly Analyst Cost (Fully Loaded)	$9,000	$7,000	$5,500
Investment in KCS Per Analyst	$4,500	$3,500	$3,000
Number of Analysts In the Group	10	10	10
Cost Per Resolution (Base)	$107	$28	$6
Average % Analyst Participation Rate	70%	65%	65%

Months to KCS Article Set Maturity	8.6	4.2	2.1
Average Analyst Daily Capacity at Maturity	8.5	18.9	53.3
Average Group Daily Capacity at Maturity	84.7	188.9	533.3
Analyst Daily Capacity Improvement	112%	57%	21%
Cost Per Resolution at Maturity	$50.60	$17.65	$4.91
% Improvement in Cost/Resolution	112%	57%	21%

KCS ROI

ROI—First 12 Months	736%	802%	286%
ROI—First 24 Months	2450%	1564%	525%

The increased capacity is a function of the fact that it is much faster to find a KCS article in the knowledge base than create a new one. By consistently capturing the experience of solving issues in the knowledge base, a great percentage of the work moves to the known category (shorter time to resolve). We find that once KCS is implemented, most organizations are surprised at the level of redundancy in their work. The 80/20 rule (80% of the requests have already been answered somewhere in the organization) represents a good average.

This model forecasts the time it will take for "KCS article set maturity." Maturity is when most of what the organization knows is captured in the knowledge base. Maturity is measured in months. In the examples given above, it varies from 9 months for an environment with complex issues to 3 months for an environment with low complexity.

The assumptions used for this model are:

245 Work Days Per Year Average

21 Work Days Per Month Average

8	Work Hrs Per Day
168	Total Work Hrs Per Month
101	Available Work Hrs/Mo, Utilization rate 60%
288	Available Work Min/Day, Utilization rate 60%

Fully loaded annual cost/responder High	$108,000
Fully loaded annual cost/responder Med	$ 84,000
Fully loaded annual cost/responder Low	$ 66,000

A Few Words of Caution

KCS is a powerful best practice that initially improves the capacity of a support organization and over time creates new capabilities and benefits. At the outset, we need to consider how the increased capacity will be used and set knowledge worker and executive expectations realistically. These are the Leveraged benefits described earlier.

While the model shows a dramatic return on investment through the reduction in the average cost per problem, the savings may not be in a form the organization can or wants to directly realize. For most support organizations, the primary cost component is labor. While the adoption of KCS can reduce the labor costs on a per-unit-of-work basis, the savings can only be realized by reducing staff. This might be either difficult or undesirable due to shift coverage requirements or diversity of the technologies being supported. Reducing staff also diminishes the longer-term opportunity to create new capabilities and value-added services.

Capitalizing on increased capacity is a tricky thing. Organizations that do not have a plan for how they will use the capacity are at risk of losing it. Incremental capacity can be absorbed by an organization without even thinking about it. In the absence of a plan, the existing work will naturally expand to fill the time.

Capitalizing on increased capacity is a tricky thing.

Having a plan for how the increased capacity will be used is also an important element of setting executive and participant expectations. Executives like to see numbers that they can track, but some of these numbers (like incremental costs) are estimates. Be selective about the numbers touted. Placing the benefits of KCS in the bigger context of organizational goals including customer loyalty is very important here, to communicate the complete value of KCS

(see the section on the strategic framework).

Of no less importance, knowledge workers want to know that their work to adopt KCS will not be "rewarded" with a layoff notice. Show them a plan and document the executive support for implementing the future changes and, specifically, how the incremental capacity will be used.

Following are some options to consider for how the organization can leverage newly available capacity:

- Improvement in service level to requestors without incremental cost.
- Creation of predictive and preemptive capability that will reduce demand
- Creation of value-added services that will increase revenue
- Increased interaction with product, process and policy owners to drive improvements that are based on user or customer experiences.
- Higher levels of participation in product testing during alpha and beta phases of development (this participation is also the best way to seed the knowledge base with experience about the new products).
- In an environment with increasing workload, the need for additional headcount can be reduced or postponed.
- In an environment with flat or decreasing workload, a reduction in staff may be the right thing. KCS will enable a lower cost of support while sustaining service levels. But we must be sure the reduction in staff is feasible.

Technique 8.4: Promote Teamwork

Since the foundation of KCS is capturing and improving the collective experience of the organization, teamwork and collaboration are critical. While tools can enable collaboration, the behaviors are what drives the benefits. And, the behaviors are heavily influenced by the culture of the organization. Creating a culture of collaboration and collective ownership is the leader's responsibility. The leader needs to ensure the environment nurtures and recognizes collaboration that is creating value. In our experience, most teams could use some work in this area. Historically, organizations have focused so heavily on individual contribution the shift to valuing collaboration and teamwork can be a challenge. For example, the old management practice of stack racking individuals is hugely dysfunctional if we want create a knowledge sharing, collaborative environment. Fortunately, organizations are moving, albeit slowly, to recognizing collaboration and teamwork as important competencies. A fairly well established and successful example of this shift from individual contribution to team capability is the adoption of the Agile Methodology in software development environments. The same kind of shift that software development has made with Agile is necessary for long term KCS success.

Applying "The Five Dysfunctions of a Team"

Patrick Lencioni's book, The Five Dysfunctions of a Team, calls out the hallmarks to watch for when assessing the health of the team. From his website, www.tablegroup.com, we can see the following list of issues and strategies for overcoming them:

DYSFUNCTION	STRATEGY FOR OVERCOMING
Absence of Trust	• Identify and discuss individual strengths and weaknesses
	• Spend considerable time in face-to-face meetings and working sessions
Fear of Conflict	• Acknowledge that conflict is required for productive meetings
	• Understand each individual team member's natural conflict styles and establish common ground rules for engaging in conflict
Lack of Commitment	• Review commitments at the end of each meeting to ensure all team members are aligned
	• Adopt a "disagree and commit" mentality—make sure all team members are committed regardless of initial disagreements
Avoidance of Accountability	• Explicitly communicate goals and standards of behavior
	• Regularly discuss performance versus goals and standards
Inattention to Results	• Keep the team focused on tangible group goals
	• Reward individuals based on team goals and collective success

In the appendix of Lencioni's book, he offers a simple survey for assessing the health of an organization. It appears to be the best tool available. We have found it to be relevant to small teams as well as large teams. KCS is a team sport; *The Five Dysfunctions of a Team* is a great resource for organizations adopting KCS.

Technique 8.5: Tap into Internal Motivators

Extensive research has been done in the area of motivating employees. The research shows that the factors that contribute to job dissatisfaction are different from those that create satisfaction and are motivating.

We like the insight in an article published by the Harvard Business Review, One More Time: How Do You Motivate Employees? by Frederick Herzberg. This is one of Harvard Business Review's most requested reprints. Another great reference on this is the book Drive: The Surprising Truth About What Motivates Us by Daniel Pink (this should be required reading for all managers in a KCS environment). Both resources provide compelling evidence that "sticks and carrots" don't work!

Herzberg identified hygiene factors, things that can be a distracter to employee motivation if they are not taken care of, and motivational factors, the things that are actually motivating. While hygiene factors include things like salary, work conditions, company policies, and relationships with co-workers, the four top motivation factors are achievement, recognition, the work itself, and responsibility. Let us consider how Herzberg's motivating factors show up in KCS.

MOTIVATION FACTOR	APPLICATIONS IN KCS
• Achievement	• KCS proficiency level, earning the right to publish or becoming a KCS Coach • Creating KCS articles others are using • Expanding breadth of product knowledge • Contributing to the goals of the organization in a measurable way • Collaborating as part of a group that is creating value for the business
• Recognition	• Reputation based on creation of value in the knowledge base, others know you because of your KCS articles in the knowledge base • Acknowledged for knowledge contribution through KCS measures and reports that are visible to the group • Acknowledged by organization leaders as role model for others
• The Work Itself	• Less redundancy, always working on interesting new things • Confidence in taking broad range of incidents because the knowledge base complements existing knowledge
• Responsibility	• Licensed to publish (KCS competency) without review by others (autonomy with accountability) • Licensed to modify/improve content • Part of a team • Collective ownership for content—"flag it or fix it"

Daniel Pink is a New York Times columnist with a lot of curiosity. One of the things he was curious about was what motivates people. In *Drive*, he synthesizes his extensive research on this topic in to three key points:

- Mastery

- Autonomy
- Purpose

These 3 factors are offered in the context of distinguishing physical tasks from intellectual work. Surprisingly when we are talking about intellectual work - like knowledge work - tangible rewards (like bonuses) are not effective. In fact they are disruptive. For a great animated summary of the book see his YouTube video.

Technique 8.6: Recognition Programs

First, based on what we just discussed in the Internal Motivators technique, we have to recognize that recognition is far more powerful and effective than rewards. By rewards we mean tangible things like cash or prizes. We want to create an environment where we leverage the things that motivate knowledge workers to contribute to the knowledge base. The two most powerful motivators are a sense of accomplishment (how we feel about our contribution) and recognition (acknowledgement of our contribution by others).

The impact of the recognition is dependent on the value the organization places on it. If the organization puts high value on having a KCS Publisher license or being in the role of Coach or KDE, that makes those a high impact form of recognition. There are lots of ways and opportunities for leadership to recognize knowledge workers and their contribution.

Is there a place for rewards and prizes? Maybe. First, it depends on the culture of the organization. In some environments it can be an effective way to create awareness, excitement, and fun. The same program in another environment may be seen as silly or elementary. Second, the program has to be thoughtfully designed.

To motivate the right behavior and promote KCS adoption early on, many companies implement reward and recognition programs. Historically we have seen that these programs are hard to get right. In addition to frequently misunderstanding what really motivates people, the programs are often based on activities, not outcomes, and end up driving the wrong behaviors. They become outdated as the organization progresses on the KCS journey, but the programs seldom have an end of life plan.

KCS rewards and recognition programs aimed at getting started must have an end date. If they continue too long, they send the signal that knowledge sharing and reuse are not part of the job, but something "above and beyond" to be specially rewarded. In fact, the opposite must happen—KCS practices must be integrated into all participants' job descriptions and formal job evaluation programs. At this point, the primary reward and recognition for doing KCS well is the same as it is for doing any other part of the job well: continued employment, good reviews, and career advancement.

We have learned a tremendous amount about what works and what doesn't. Some of the design principles of successful programs include:

- Legitimate metrics—tied to independent feedback and requestor input, most often done through satisfaction or effort surveys
- Alignment to organizational goals—measures directly link to and reinforce desired outcomes and the strategic framework
- Time constraints—clear beginning and end, and a plan for what is appropriate for the next phase of the KCS journey

- Balance of individual and team rewards—consider virtual teams as well as geographical and subject matter teams
- Compatibility with the individual—tailored to the values, interest, and styles of the person or team (don't embarrass an introvert!)
- Equal opportunity for participation—include recognition for different positions, roles, and responsibilities
- Recognition of diversity of skills—good generalists are as valuable as good specialists; recognition for each of the skills needed for success
- Promote collaboration, not competition—measure and recognize each individual's own progress and achievements

Given these guidelines, most organizations develop programs to appeal to different motivational factors. Here are some motivation and reward examples:

- *Challenge*—Set new records for key outcomes (often a team recognition)
- *Attention*—A visit with/from a senior level executive
- *Affirmation*—Add meaningful job opportunities or new roles, like membership in the KCS program team or trips to industry conferences or events

One key to using rewards and recognition programs effectively is to view them as part of the communications plan. These programs are an effective way for leaders to draw attention to new practices and priorities.

Technique 8.7: Communication is the Key

Effective communication in today's environment is difficult—there is a lot of competition. Some recent market research shows that in today's world of hype, a message must reach an audience 36 times before being acknowledged or absorbed.

There is considerable evidence that we are in the midst of a trust crisis. The level of trust people feel towards leadership is dismal; less than one out of five people trust leaders to tell the truth (for more on this see the Edelman Trust Barometer). This trust crisis has a domino effect that undermines the effectiveness of the organization, which creates apathetic or disengaged employees. Recent Gallup research indicates that 66% of employees are not engaged in their work. Amazingly, this is a new high for employee engagement over the last three years (see Gallup research). Disengaged employees are not likely to contribute to the success of the organization or the knowledge base. Disengaged employees who interact with our customers are not likely to create the compelling customer experience that reinforces our brand promise.

Apathy is death to any knowledge management practice. The antidote for apathy is engagement, and the first step in engaging people is communications. If people understand the compelling purpose of the organization and are apathetic towards that, then they are in the wrong organization. More often the cause of the apathy is the absence of an opportunity to understand the organization's compelling purpose. The compelling purpose is a key part of the overall company vision. For more on creating a vision see the Develop and Communicate a Vision technique.

Our challenge is to deliver thoughtful and well-designed messages about the organization's vision and how KCS serves that vision. To get the message across we must deliver these messages frequently and through different delivery vehicles. Knowledge Workers also have to hear the same message from executive leadership as they do from their team leadership. Communication is a key part of any change management effort.

Develop a Communication Plan

A good KCS Communication Plan is really a marketing plan. It should include:

- **Target Audiences**—Whom do we need to engage and influence in order to have a successful adoption? The critical audiences are: executives, managers, and knowledge workers. The expanded list would include IT staff, product management, marketing, and sales personnel.
- **Key Messages**—Different audiences (i.e., executives vs. knowledge workers) require different messages because they have different roles and objectives.
- **What's In It For Me? (WIIFM)**—Clearly communicate relevant personal benefits, not just benefits to the organization as a whole.
- **KCS Q&A/FAQ**—As we discuss implementation, we capture the questions that are frequently asked and

develop thoughtful answers. An easily accessible written response ensures consistent messaging. Click here for a sample FAQ.

- **Overcoming Objections**—We must consider the real and perceived objections for each audience and include enough detail in our responses to alleviate concerns.
- **Elevator Pitch**—A short (about 10 seconds) overview of the KCS benefits and why it matters to the listener. A good elevator pitch captures the listener's interest, makes them want to learn more, and creates a positive perception.
- **Delivery Vehicles**—Different audiences will require different communication tools—on-line, in-person, and through organizational communication tools. We must think through the most appropriate delivery vehicles, by audience, so that the message will be heard and remembered.
- **Programs and Activities for Engagement and Socialization**—With messages and communication tools in mind, we need to consider how to socialize the ideas. Are there ways to create interactive conversations, solicit input, and encourage support? What programs or activities will reach the target audiences? Examples of programs:
  - Meetings (all-hands, group, 1:1)
  - MBWA ("Management By Walking Around"—casual skip-level communications)
  - Coaching and training
  - Conference calls
  - Themed giveaways
  - Newsletters (hardcopy and online/email)
  - Email updates
  - Bulletin boards
  - Posters
  - Videos
  - Websites
- **Project Plan and Timeline**—all these elements should be organized in a project plan with timeline (and appropriate budget).

Objection Handling

Raising awareness and increasing support for KCS are everyday activities for leaders and the program champions. How we handle objections is crucial. Objections are a gift. They give us some insight as to how people are thinking about KCS. The confidence and consistency with which we handle objections has a big impact on people's confidence about the KCS program. It is helpful to have a small team of people thoughtfully work through the likely objections and responses and capture them as part of the KCS communications plan. We have found a few basic rules to be helpful:

- Be sensitive to the feelings behind the objection
- Acknowledge the validity of feelings; empathize
- Seek to understand the issue from the other's perspective
- Offer an alternative perspective (don't debate or argue)
- Use WIIFM to craft a response that they can relate to

With these behavioral ideas in mind, we can effectively use the "objection handling" content that is pre-prepared. We handle objections much better when we are ready and consistent. For organizations where conflict is uncomfortable or

communication is limited, written objection-handling material can also proactively address concerns that people might be reluctant or unable to raise themselves.

When we ask organizations that have successfully adopted KCS what they would do differently next time.... they almost always say "we should have done more communications."

Technique 8.8: Leadership Accountability to the Knowledge Worker

Executive leaders in the organization must take accountability for the knowledge workers' success. The accountability covers a number of areas including:

- Leadership development. Leaders at the team level (formally called managers) need the training and coaching to make the transition from managing to leading and be accountable for supporting the knowledge workers' success with KCS.
- Ensure the Knowledge Workers have the perspective, training, and coaching they need to be successful.
- Ensure the technology infrastructure has the functionality, integration, and performance to support the Solve Loop.
- Report on the status and progress on the WIIFM (What's In It For Me) for the knowledge worker.
- Provide knowledge workers with continuous visibility to the impact of their contribution.

These leadership responsibilities are critical for a successful adoption as well as being able to maximize and sustain the KCS benefits over time.

Leadership Development

KCS represents a bigger change for the 1st and 2nd line managers than it does for the knowledge worker. Successful adoption requires a new way to think about process, measures, and how we assess people's contribution. As KCS matures, the need for management diminishes and the need for leadership increases. This means a shift from directing and judging to engaging and coaching. The single most common point of failure in a KCS adoption is the 1st and 2nd line managers not making the shift and not taking ownership for KCS success.

Executive leadership needs to support the managers in making the transition by changing how the managers are measured and providing them with training and coaching on how to become leaders.

The Consortium members are doing a lot of work on this topic. (See Nurturing an Adaptive Workforce as an example.) Here is a list of some of the key competencies that have been identified for team leaders in a KCS environment:

- Strategic agility - understand how tasks relate to the big picture (double loop thinking)
- Innovation leadership - encourage creativity
- Dealing with ambiguity - measuring things we cannot count; there is no one indicator for the creation of value
- Courage - willing to try new things and iterate for improvement
- Motivating others - understand the motivation factors at play in knowledge work
- Customer focus - living the brand promise

Executive leadership must see the investment in leadership development as necessary to sustain the benefits of KCS for the long term. If they don't, all the investment in early phases of adoption (the first 9-18 months) will have been wasted.

Knowledge Worker Training and Coaching

Knowledge work requires judgment. The basis for that judgment comes in part from understanding the vision (see Technique 8.1: Develop and Communicate a Vision) and in part by understanding the benefits of KCS in the context of the bigger picture and how the the Solve Loop feeds the Evolve Loop. This understanding is what enables people to create value, when they work tasks in the context of bigger picture. The keys to establishing this perspective lie in consistent communication, training and coaching.

Infrastructure Functionality, Integration, and Performance

The organization's infrastructure must support doing the Solve Loop at or near the speed of conversation. As we have mentioned in the description of the Process Integration Practice, the infrastructure does not have to be perfect in order to start on the KCS journey and realize some of the early benefits. If the knowledge workers are inspired by the benefits they will figure out how to do the Solve Loop even with really crude tools. However, it is extremely difficult to maintain the knowledge workers' interest if they do not see continuous improvement over time in the functionality, integration, and performance of the infrastructure they use to get their work done. Our experience shows that leadership has 9-12 months to move the user interface from crude to obvious and easy.

Progress on the WIIFM

We entice the knowledge workers into KCS with a promise of a better, more interesting work environment. We discuss the importance developing and communicating the "What's In It For Me" list of benefits in Technique 8.7: Communication Is The Key. And, the list typically includes things like:

- Less redundant work
- More interesting work, broader scope of work
- Smaller backlog
- Lower stress
- Recognition for creation of value

Because these benefits happen slowly over a period of time, their realization may not be obvious to the knowledge worker. If leadership does not have baseline measures established and reporting capabilities in place to help the knowledge worker see the change, it is an opportunity missed.

Visibility to Impact of Contribution

We talked about what motivates people in Technique 8.5. A critical enabler to motivating knowledge workers is providing

them with visibility to the impact of their contribution. If people cannot see the value they are creating, they will lose interest. The leaders in the organization have to provide visibility to article reuse: both internal and external (self-service). While article reuse in self-service can be difficult to figure out, it can be done. And it is critical that the people who create and maintain the knowledge have visibility to the self-service activity and success indicators.

Often one of the greatest contributions of the collective experience of the organization are the business improvements that come as a result of the Evolve Loop analysis. This is a case of deferred gratification and leaders need to make the effort to help people see that the collective effort over time has lead to specific improvements in products, services, processes, or policies. It takes time for the pattern to emerge, and it takes time for the root cause analysis and corrective actions to be implemented. It is not unusual for the whole process to take 9-18 months. Dramatic improvements may result, but it is far removed from the many well done Solve Loop events that it took to get there. The people who made it happen won't realize it was a result of their effort if leadership doesn't provide that visibility.

Technique 8.9: Leadership and Communication Indicators

Indicators of the health and effectiveness of the Leadership and Communications practice fall into two major categories: communication effectiveness and realization of value for the organization.

Communication Effectiveness Indicators

Getting a message across to all the key players in an organization is a challenge. It is the responsibility of leadership to ensure that the knowledge workers understand why the organization is doing KCS, and what the benefits are for both the organization and the knowledge worker. Communication Effectiveness Indicators assess the level of buy-in and understanding across the organization. Do the mid-level and team leaders in the organization understand the benefits and their role in supporting the knowledge workers success with KCS? Do the knowledge workers understand KCS in the context of the big picture as well as understanding the KCS at the task level? This is critical if we expect knowledge workers to make good judgments at the task level in the context of the bigger picture.

Communication effectiveness is measured through surveys and assessments conducted repeatedly over a period of time. The surveys need to be crafted in a way to assess understanding and buy-in. Some organizations have gone as far as having semi-annual interviews with knowledge workers to assess this. Based on the learnings from the interviews, they suggest actions to improve understanding and buy-in to the leaders of the organization and then track the improvement over time.

Organizational Value Indicators

We have discussed the three categories of benefit KCS creates:

1. Operational efficiency
    1. Increased capacity
2. Self-service success
    1. Increase the frequency users use self-service
    2. Increase the user's success rate
3. Business improvements (products, documentation, processes and policies)
    1. Number of improvements identified and submitted to the business owner
    2. Number of improvements implemented by the business owners

These benefits are realized at different points in time along the KCS journey. And, the way in which each will show up varies by organization.

The strategic framework (see Technique 8.2) is helpful in identifying how the KCS benefits line up with the organization's goals. If we use our sample Strategic Framework we can identify the following Organizational Value Indicators:

- Customer
  - Loyalty (NPS)
  - Customer effort
  - Renewal rate
- Employee
  - Attrition rate
  - Employee engagement/loyalty
  - Employee effort
- Business (financial)
  - Revenue growth
  - Expense management
  - Profitability

We want to establish baseline positions for these items and then track progress.

Summary

KCS is not something we do in addition to solving problems. It is the way we solve problems.

In order for our service organizations to deliver higher value to our customers, we must focus on the source of value: people and knowledge. We must transition from a call-centric, transaction-oriented model to a knowledge-centric, relationship-based model. We must recalibrate our measurement systems to measure the creation of value, not just activity. We must rethink the role of information, knowledge workers, managers, and customers to take full advantage of the knowledge that emerges from the experience and interactions of the service and support organization.

KCS requires many changes in an organization, and Consortium members have demonstrated the great benefits that emerge. In this guide, we have attempted to capture the best we know to date on the techniques for embracing the core practices that make up KCS. These practices, organized in the individual-focused Solve Loop and the team-focused Evolve Loop, represent the essence of KCS—sharing and reusing knowledge to create value for the organization.

- Solve Loop
  - Capture
  - Structure
  - Reuse
  - Improve
- Evolve Loop
  - Content Health
  - Process Integration
  - Performance Assessment
  - Leadership & Communication

As we work to adopt KCS, we usually discover that many mental shifts are happening. The way we think about our efforts and those of the group change:

- Focus on knowledge rather than experience.
- Emphasize Knowledge-Centered Service over knowledge engineering.
- Knowledge is best managed at the point of creation, in the context of demand.
- Linear, production line processes become a collaborative, knowledge-enabled network.
- Emphasize just-in-time review over just-in-case creation.
- Take individual ownership of actions and performance and have collective ownership of the knowledge base.
- Look to managers for leadership and motivation rather than direction and instruction.

Lessons Learned

During adoption, we often over-engineer workflows and content standards. They become too complicated and cumbersome. We also spend too much time converting legacy data rather than creating KCS articles. This overhead slows down and demoralizes the participants, who are often eager to embrace new areas of knowledge and practice their newfound KCS skills. We should not dwell on the laggards who might be resisting change, but reinforce the success of the early adopters through positive support.

KCS introduces a new way to think about and manage the business or support.

Other proven success factors make a difference in KCS adoption. Perhaps the most obvious and visible one is *executive sponsorship*. Executives need to populate the strategic framework with goals that support the KCS workflow. They need to fund the infrastructure and resources and maintain support for the team while communicating (in a sustained way) the benefits, strategic impact, and progress of the KCS program. These communications, across their peers, upward to senior management, and downward to the team, provide leadership and motivate the team.

Communication from executives and within the team should be part of a formal program to help each stakeholder group—the requestors, responders, and organization—understand and support the transition. Glitches and ditches are common, and communications help manage expectations, overcome objections, and deal with exceptions. Audience-appropriate messaging through multiple communication avenues will smooth adoption and help people understand, "What's in it for me?" Do not let the communications plan be too short or superficial—communications are central to change management and need to be both thoughtful and sustained to have a real impact.

The *coaching* role, new to many teams, has an enormous effect. When the right people are selected (ideally, by their peers) and given the time and support to coach, KCS prospers. Coaches must have strong interpersonal and influence skills as well as an operational understanding of the KCS practices.

Did we mention...

Do Not Put Goals on Activities!

The right *metrics* are based on goals for the outcomes or results and enable trend analysis for the leading indicators or activities. Do not put goals on activities. Through the strategic framework, we empower and motivate knowledge workers (and managers) with a clear association of their individual activities to organizational goals. A balanced view of the metrics and trend analysis should enable continuous improvement and performance assessment.

Performance assessment should focus on the *creation of value* by individuals and teams. It must consider both qualitative and quantitative measures and reinforce the transition from an activity-based model to a value-based model.

Another important mental shift is the *attitude* about the KCS deployment. The team and its enablers and stakeholders must see KCS as an organizational change, not just a tool. KCS can be assisted by technology, but is dependent far more on the changes in organizational values, interactions, and processes. Knowledge management and KCS are about people, not technology. They offer a new way to think about and manage the business of support.

The adoption of KCS inevitably changes the nature of interaction between the levels or tiers in the traditional support

model. The distinction between support levels should diminish, creating a sense of one team aligned to a common purpose.

It's a Matter of Balance

As we said in the introduction, the adoption of KCS represents a transformation. It requires a shift in the organization's culture, values, and focus. It requires a balance of:

Individual	and	Team
Activity	and	Outcome
Completion	and	Evolution
Content	and	Context
Knowing	and	Learning
Management	and	Leadership

We hope this guide has helped launch you your KCS journey. KCS adoption is not a trivial undertaking. It requires both patience and persistence; change is hard!

The *KCS v6 Practices Guide* reflects the collective experience and thinking of the many dedicated individuals and organizations who make up the Consortium for Service Innovation, as well as the academic work reflected here. We continue to learn and evolve our understanding. The evolution is a function of our members' courage to try new and therefore unproven ways to address the challenges of change. If you like what you see in the KCS Practices and would like to contribute to its ongoing evolution, please join us. The journey continues!

About the Consortium

The Consortium for Service Innovation is a non-profit alliance of service and support organizations. Through a process of collective thinking and shared experience, the members develop and validate innovative ways to improve the customer experience. This work bridges emerging academic thinking and research to tangible, tactical practices that optimize business results.

If you or your organization would like to participate in the continuing evolution of the Knowledge-Centered Service models or other programs under way, consider joining the Consortium for Service Innovation.

Appendix A: KCS v6 Update Summary

Description of the difference between KCS v5 and KCS v6 available on this web session (YouTube).

Topics	Link
Introduction/Summary/General	
KCS Principles and Core Concepts	KCS Principles and Core Concepts
Knowledge-Centered Support replaced with Knowledge-Centered Service	Section 1 Knowledge-Centered Service
Renumbered techniques to contain practice number.technique number	Throughout
Moved KCS benefits	Technique 8.3 KCS Benefits and ROI
Removed Phases of Adoption references throughout replaces the link	KCS v6 Adoption Guide
All references to customer service specific techniques are in block quotes	Throughout
Move ITIL and KCS to appendix	Appendix F: KCS and ITIL
Replace Customer with requestor and Support Analyst with responder	Throughout
KCS is a Risk Management System updated	KCS Manages the Dynamics of Knowledge
Removed Knowledge Assets will address in Predicative Customer Engagement	
KCS Article changed Customer to Requester changed Support Analyst to Responder added	The KCS Article and

Topics	Link
knowledge worker	througout
Replace participation with link rate	Throughout
Incident or case changed to system of record when referring to non-customer service related interactions	Throughout
AQI update to be positive	The Article Quality Index
Changed vocabulary from capture, structure, reuse to reuse, improve, create	Throughout
Replace Draft article with Validated confidence	Throughout
Capture	
Capture Relevant Content technique combines with Searching is Creating	Technique 1.3: Searching is Creating
Structure	
Reuse	
Content Health	
Content Health Indicators	Technique 5.10: Content Health Indicators
Article Quality Index (AQI) replaced with Content Standard Checklist	Technique 5.10: Content Health Indicators and throughout
Removed web-success replaced with self-service success	Throughout
Article State becomes Article Confidence new definition of Article State is combination of Article Confidence, Audience (formerly Visibility), and Governance	Throughout
Archiving Old Articles new technique taken out of Managing KCS Quality	Technique 5.5: Archiving Old Articles
Dealing with Legacy Data new technique taken out of Managing KCS Quality	Technique 5.6:

Topics	Link
	Dealing With Legacy Data
Priming the Knowledge Base with New Information new technique taken out of Managing KCS Quality	Technique 5.7: Priming the Knowledge Base With New Information
Global Support Considerations new technique taken out of Managing KCS Quality	Technique 5.8: Global Support Considerations
Knowledge Domain Analysis new technique and combined with New vs. Known	Technique 5.9: Knowledge Domain Analysis
Self-service Measures new technique replaces Assessing the Value of Articles	Technique 5.12: Self-Service Measures
Process Integration	Throughout
Process Integration Indicators added, then name updated to Process Adherence Review	Technique 6.5: KCS Process Adherence Review
Performance Assessment	
Added Coaching for Success technique	Technique 7.2: Coaching for Success
Changed Balanced Scorecard Technique to Assessing the Creation of Value Technique	Technique 7.4: Assessing the Creation of Value
Updated individual radar charts to include link accuracy, contribution index	Radar Charts: Creating a Value Footprint
Updated team/team leader radar chart with link accuracy, contribution	Radar Charts: Creating a Value Footprint
Updated scenarios to include Process Adherence Review (formerly Process Integration Indicators)	A Scenario—Examples

Topics	Link
	of KCS Reports
Leadership and Communication	
Replace reward with recognize	Throughout
Updated Compelling Purpose to include vision, mission, and brand promise	Technique 8.1: Develop and Communicate a Vision
Moved KCS benefits and ROI from Strategic Framework into new technique	Technique 8.3 KCS Benefits and ROI
Added technique on the Leadership for knowledge workers	Technique 8.9: Leadership Accountability to the Knowledge Worker
Leadership and Communications Indicators	Technique 8.10: Leadership and Communication Indicators
Appendix	
Summary of updates	Appendix A: KCS v6 Update Summary
Glossary of Terms updated	Appendix B: Glossary of KCS Terms
Article Quality Quick Reference Guide updated	Appendix E: Article Quality Quick Reference Guide
Right to Use with Attribution updated	RIGHT TO USE WITH ATTRIBUTION

Click here for the KCS v5.3 update summary.

Appendix B: Glossary of KCS Terms

Term	Definition
Activities	Leading indicators that count activity (# of articles created or reused). Activity alone is not an indicator of value.
Add it	Creating a new KCS article in the workflow, if one does not exist.
Article Quality Index (AQI)	See "Content Standard Checklist"
Articles	The collective experience of the support organization in solving problems and answering questions. Articles can include a variety of issues: usage or "how to", configuration, interoperability, performance, defects, procedural, or diagnostic information.
Author	The initial creator of a KCS article.
Average work time to resolve	Number of minutes consumed per incident in developing an answer, fix, bypass or workaround. Determined by dividing the total minutes worked by the number of incidents resolved.
Balanced Scorecard	A method of goals and metrics that links individual and organization goals. Examines performance from multiple points of view. (see The Balanced Scorecard by Kaplan and Norton)
Call deflection	The number of customer issues solved through self-service that would have

Term	Definition
	become incidents (this is a subset of self-service customer success)
Candidate Knowledge	Search words become part of the knowledge base as standalone article or part of an existing article.
Capture	A Solve Loop practice. While solving the problem, the knowledge worker captures the customer's context.
Cause	The underlying or root cause of the problem.
Citation	Reuse of your articles by others
Closed loop feedback	Insight generated by the Evolve Loop that continuously improves the customer experience as well as the information in the knowledge base.
Coach	KCS role that supports the development of KCS Candidates and KCS Contributors.
Communication Effectiveness Indicators	Used to assess the level of buy-in and understanding across the organization
Compelling purpose	The purpose of an organization; people should connect with it on a personal, emotional level.
Competency profile	Percentage of knowledge workers at each level of the KCS competencies: KCS Candidate, KCS Contributor, and KCS Publisher.
Content	Various forms of content that solve a specific issue, including KCS articles, white papers, documentation etc. This content must be searchable, answer the specific issue in the context of the audience asking, and in a maintained repository.

Term	Definition
Content Health	An Evolve Loop practice focused on article structure, content standards, content quality, and article life cycles.
Content Standard	A formal document which describes decisions the organization has made about KCS article content and structure.
Content Standard Checklist	Formerly known as the Article Quality Index or AQI, the Content Standard Checklist provides the structure to ensure a KCS Contributor understands and can create articles based on the criteria defined in the KCS Content Standard.
Contribution Indicators	How often a knowledge worker is reusing, modifying, or creating a new knowledge article.
Contribution Index	The ratio of the number of times a knowledge worker appropriately contributed as a percentage of their total opportunity to contribute.
Context	Captured in the Solve Loop, it is the description of the needs and experiences of the requestor in their own terms.
Cost per incident	Total support costs divided by the number of incidents closed
Creator	The author of a KCS article.
Cross-functional measures	Measures to which multiple functions within the organization contribute. For example, product improvements require support to capture the interactions and recognize trends to give development credible input on high leverage opportunities for product improvement. Development must execute on these opportunities. The measure is shared by support and development. (See Transforming Performance Measurement by Spitzer)
Cultural Health	Knowledge workers' attitude with respect to trust, commitment, conflict resolution, accountability, and focus on results. Measured through surveys. See Patrick Lencioni, The Five Dysfunctions of a Team
Customer loyalty	The level of emotional connection a customer feels towards the company: a longer-term measure of overall relationship. Indicators include renewal rate,

Term	Definition
	new product/upgrade adoption rate, and reference ability.
Customer Satisfaction	Transaction-based measure of the degree to which we have met the customer expectations. This is a short-term measure of the customer experience with support. Indicators are speed or average work time to resolve, "percentage first contact resolution," technical knowledge, and politeness of the knowledge worker.
Data	Unorganized words or numbers.
Demand-Driven Visibility	As articles are found and used they are improved. Articles are migrated to a broader audience as they are proven accurate and useful.
Employee turnover rate	Internal attrition rate at which knowledge workers are leaving the support organization.
Environment	An element of a KCS article that contains product information or technology (hardware, software,network, etc.) the customer has that is relevant to the issue. Has anything been changed recently?
Evolve Loop	A continuous improvement process that integrates individual and organizational processes.
Evolve Loop content	Content that is created outside of the workflow from a collection of articles created in the Solve Loop. Includes FAQ's, Hot articles, and Root Cause Analysis.
Exception	Any type of issue including • "How to" or Q&A • Interoperability issues • Configuration issues • Defects

Term	Definition
	• Diagnostic procedures • Procedural
External	Indicates a KCS article is available outside an organization. The audience metadata field is set to either partner, customer, or public.
Executive sponsor buy-in	The executive champion for the KCS program, who understands KCS and is vocally committed. This is a qualitative measurement, but may be judged by the willingness of the champion to present the plan for the project to executive management, to host a kickoff with the project team, and to support communication efforts with email and other outreach.
Findability	Term used to describe the effectiveness of a search to returning relevant results.
Findable	A search engine returns a relevant KCS article.
Fix it	A KCS Contributor or Publisher modifying an existing article in the workflow.
Flag it	A non-licensed user commenting on an article so that an authorized knowledge worker can modify it.
Improve	A Solve Loop practice. Knowledge maintenance and updates happen through real time, demand-driven reuse and review.
In the moment	Knowledge is shared and created while solving an issue.
Incident	A request for an answer to an exception or issue, tracked through an incident management system, CRM, or call-tracking tool.
Incident volume	Number of incidents, cases, or tickets opened over a period of time.

Term	Definition
Information	Organized data without an associated action
Internal Feedback	Sharing of information with knowledge workers; i.e. Monthly assessments, Content Standard Checklist assessments, and feedback attached to articles.
Issue	A field in a KCS article template where the question from requestor is recorded. Any type of interaction or exception including • "How To" or Q&A • Interoperability issues • Configuration issues • Defects • Diagnostic procedures • Procedural
Just-in-Time article quality	Everyone interacting with the knowledge base is responsible for the quality of the articles; the articles are updated as they are being used to solve an issue.
KCS	Knowledge-Centered Service
KCS article	Is the physical document living in the knowledge base. It contains the issue, environment, resolution, cause, and attributes. It represents the collective experience of the organization in solving problems and answering questions. KCS articles can cover a variety of issues: usage or "how to," configuration, interoperability, performance, defects, procedural or diagnostic information.
KCS Article Confidence	The process a article undergoes as it matures (work in progress, not validated, validated, archived). Article confidence indicates how sure we are in the resolution and content standard of the article.
KCS Article State	A combination of article confidence, audience (formerly visibility), and governance.

Term	Definition
KCS Article Audience	Audience that the article is available to. Internal, Customer, Partner, Public. Formerly called article visibility.
KCS Article Governance	User or user group who can create and edit articles.
KCS Candidate	Provisional contributor to the knowledge base; a basic user of the knowledge base who is familiar with capturing and structuring content. Can create internal articles and modify their own but cannot modify others.
KCS Coach	KCS role that supports the development of the KCS Candidates and KCS Contributors.
KCS Contributor	Creates, modifies, and reviews articles for publishing to a defined audience.
KCS Council	A cross-functional group that meets on a regular basis providing a forum for the continued improvement to the content standard, workflow, technology integration, and feedback systems. Early in the adoption process, they are focused on the implementation activities. Later on the journey, they are focused on continuous improvement.
KCS Publisher	Publishes content to an external audience.
KCS Verified	Software tools that have demonstrated certain criteria set forth by the Consortium. Currently Verified tools are listed at the KCS Academy.
Knowledge	Information that has an action associated with it and a context and experience related to its use. Information becomes knowledge at the moment of use. Gained through interaction and experience. Attributes of knowledge include: Constantly changing, Never 100% complete or 100% accurate, validated through use, experience, and interaction
Knowledge base	A technology built for the storage and retrieval of a collection of knowledge

Term	Definition
Knowledge Domain Analysis (KDA)	Evolve Loop activities performed to maintain knowledge base health and assess and increase knowledge base impact.
Knowledge Domain Expert (KDE)	Looks after the health of the knowledge base, has both technical expertise and extensive understanding of KCS processes.
Knowledge worker	Anyone who does intellectual work as opposed to physical work. People who use data and information to make judgments and decisions and/or take action. Knowledge workers can play the role of responder or requestor. Often used to refer to people doing Solve Loop work.
Lagging indicators	Qualitative outcomes: a measure of value.
Leaders	Define the vision of success within the organization; they support knowledge workers in determining the workflow and the content standard.
Leadership & Communication	An Evolve Loop practice concerned with communication, promoting an understanding of KCS, and the performance assessment model.
Leading indicators	Quantifiable activities: should be measured only to identify trends.
Legacy Data	Old knowledge content, usually stored in disparate systems and not structured according to KCS standards.
Licensed Users	KCS Contributors and KCS Publishers
Linked	Attaching content that solves an issue from a request.
Metadata	Additional fields in KCS articles used to capture information such as article state (confidence, audience, governance), date created, number of times modified, history, number of times article has been reused etc.
Not Validated	A KCS article state that indicates low confidence in the article content or structure.
Organizational Value	Used to assess operational efficiency, self-service success, and product,

Term	Definition
Indicators	process, or policy improvement that benefit the financial state of an organization.
Outcomes	Lagging indicators, difficult to measure. Outcomes are the end result of activities and are an indicator of created value.
Percentage first contact resolution	Percentage of incidents resolved on the first interaction. Used as a customer satisfaction indicator as well as an employee proficiency or process goal.
Performance Assessment	An Evolve Loop practice that involves the ways in which performance is quantified and measured by the organization. Performance measures should be clearly linked to the strategic objectives of the organization.
Plan and Design (phase of adoption)	The Plan and Design phase of a KCS adoption builds the foundation for a successful implementation. This includes assessing the current state of knowledge practices, getting the right people involved and trained, gathering baseline measurements, and setting realistic internal and external expectations. (see KCS v6 Adoption & Transformation Guide for more on phases of adoption)
Adopt in Waves (phase of adoption)	The goals of Adopting in Waves are to develop KCS competencies in the knowledge workers, create excitement through early success stories, and establish internal referenceability. This is typically done with a small first wave in order to create internal credibility of the KCS program.
Build Proficiency (phase of adoption)	An organization expands their focus toward Building Proficiency when the knowledge base for the knowledge domain has reached critical mass; most of what the organization has learned from their interactions has been captured in a way that is findable and usable by the audience making the request. Building Proficiency includes coaching, developing Evolve Loop content, and delivering value to the requestor through self-service and business improvements.
Optimize and Innovate (phase of adoption)	Phase of KCS transformation that enables continuous improvement and a sustained focus over time. Optimization includes increased importance on

Term	Definition
	reporting and feedback mechanisms in order to provide visibility to knowledge workers of their contribution and impact.
Problem	The situation in the customer's words. What are they trying to do, or what is not working?
Process Integration	An Evolve Loop practice focused on designing and optimizing the KCS workflow.
Process Adherence Review (PAR)	PAR is an indication of how well a knowledge worker is following the Solve Loop practices. Formerly called Process Integration Indicators or PII.
Product improvements	The rate at which suggestions for product, documentation, or service offering improvements are implemented by development: an indicator of influence. (Number of RFEs accepted by product development)
Profiles	The people in the system: profiles include information about who knows what.
Published	The article is available externally as indicated by the article audience attribute.
Quick Reference Guide	A one-page document that provides knowledge workers with a brief overview of the content standard.
Ratio of known to new	Reuse of existing articles versus new articles created in the knowledge base.
Readability	Articles are easier to read by using complete thoughts instead of complete sentences.
Requestor	A knowledge worker seeking information or resolution to an issue.
Resolution	The element of a KCS article that contains the fix.

Term	Definition
Resolution capacity	How many incidents can the support organization handle in a period of time? Indicators are incidents/month/analyst or average work time to resolve (work minutes, not elapsed time).
Resolution or Fix	The steps required to solve the problem or answer the question or request.
Responder	A knowledge worker providing a resolution to an issue or assisting in the development of a resolution to an issue.
Reuse	How many times an article has been applied. Article reuse is a valuable method of measuring an article's value.
RFE	Request for product enhancement
Search Indicators	A indication if a knowledge worker is searching early and often.
Searching is Creating	Content used for searching is saved and is used to enhance existing articles or frame new articles.
Self-service success	The percentage of time users find what they need on their own through a self-service mechanism.
Self-service use	The percentage of time requestors use self-service before they submit a request. Can be web based or integrated into the user interface.
Solve Loop	Represents the individual workflow that is driven by the problem solving process.
Structure	A Solve Loop practice: involves breaking down the problem or issue description into the appropriate information and structuring it according to the appropriate format.
Subject Matter Expert (SME)	A person who is an expert in a particular area or topic.

Term	Definition
Support Analyst	Someone handling a customer exception or issue.
Support cost as a percentage of revenue	The ratio of support costs to total company revenue. Used to normalize the cost of support in a dynamic environment. Other possible ways to normalize the support costs include against products shipped, licenses sold, customers subscribed (cross functional measure).
Interaction Network	Anyone in contributing to support including employees, partners, customers, and players in on-line communities and social media.
Support Organization	A group of individuals within an organization who handle exceptions.
System of record	A case, incident, email, and any type of recorded interaction of a request and response.
Tacit Knowledge	Implicit information that should become explicit in during the conversation with the requestor.
Time to adopt new/upgraded products	Rate at which customers adopt new releases or products.
Time to close	The elapsed time from request open to request closed.
Time to proficiency	The number of weeks or months required for a knowledge worker to work with a high degree of independence: the learning curve.
Time to publish	Time from initial issue discovery to the time an article is available externally.
Triangulation	Reflects the intangible nature of value - the idea that the creation of knowledge cannot be directly measured or counted. Involves looking at things from at least three different perspectives: Trends in Activity, Results/ Outcomes, and the Article Quality Index.

Term	Definition
Validated	A KCS article that is considered complete and reusable. We have confidence in the resolution and it complies with the content standard.
Work- in-progress (WIP)	An incomplete KCS article. The problem or question has been captured but the resolution is not known.

Appendix C: Metrics Matrix

The comprehensive metrics matrix below (or in .xls format) provides a snapshot of everything we have learned to date about measures. We are not recommending organizations use all of these; this is a list of possible measures and some of their attributes.

		Adptn Phase	Audience		Data Sources	View	
			Team	Individ			
Activity (leading indicators)						**DO NOT PUT GOALS ON ACTIVITIES**	
	Assisted (support center)						
	Article create/ modify	2	**X**	**X**	KM tool	Trends	
	Reuse of others Articles	3		**X**		Trends	
	Competency levels	1	**X**		Manual	Trends	
	Participation	2	**X**	**X**	CRM and KM tools	Trends	
	Workflow alignment monitoring	3		**X**	Manual - Coaches assessment		KCS

Currency trends (obsolete, modify)	3	**X**		KM tool or data mining tool	Patterns	Use patte base class
Incidents closed	1	**X**	**X**	CRM tool	Trends	Num into t
Web						
Sessions/sign ons	1	**X**		Web reports	Trends	Rela
Searches/ queries	1	**X**		Web reports	Trends	
Page hits/views	1	**X**		Web reports	Trends	
Incidents opened within 24 hours of web session	3	**X**		Web reports and CRM	Number	Link indiv
Avg # of page views/exception	3			Survey or usability studies, web analytics	Number	Som
Community						
Sessions/sign ons/visits	3	**X**		Web reports	Trend	Heal total
Posts	3	**X**		Web reports	Trend	Heal
Valued players	3		**X**	Manual	Trend	Num the c

Outcomes (lagging indicators)						
Demand based view - Whole system health (customer experience)						
Total support demand	3			CRM, Web, community	Trends	Supp Cust the t
Demand satisfaction by channel	3	**X**		CRM and Web and community	%	Optir dema
Process - Support Center (assisted support)						
Time to resolve/ relief	2-3	**X**	**X**	CRM tool		Not t the c work
Known Vs new	3	**X**		CRM and/or KM	%	Helps KCS Ideal are b
Time to relief - known	3			CRM	Avg. minutes	An in the K conte provi
Time to relief - new	3			CRM	Avg. minutes	Indic
First technical contact resolution	3	**X**	**X**	CRM tool	%	Thes succ servi

Cost/Incident (and/or exception)	4	**X**	**X**	CRM and financials	$	resol go up costs
Citations (Reuse by others)	3-4			KM tool	Number	Articl for ea
Time to publish	2-3	**X**		CRM and KM tools	Avg minutes	Helps servi get a Typic "relie articl
Collaboration (assisted support)						
Team health	4	**X**		Survey	% satisfied	Used Trust acco Cons
Organizational network Analysis	4	**X**		Manual	Network map	Ident of ov
Communications and Alignment						
Employee understanding	2		**X**	Survey	Score, trend	Asse leade
Employee buy-in	2				Score, trend	Asse leade

Communications effectiveness	2				Score, trend	Asse lead
Article Quality						
Quality index (finished Articles)	3	**X**	**X**	Manual	Score	Goal their
Framing quality index	3	**X**	**X**	Manual	Score	Input
Article life cycle	1	**X**		KM tool	Pattern over time	Mont time the li
Customer success with self-help	3	**X**	**X**	Web and manual	%	Can surve base case defin
Diversity of source; internal, external	3	**X**		CRM, KM, Web, Community	%	Indic total
Value of content (Articles)	2				Index	Two conte conte
The value of the KB	4	**X**		CRM, Web	$$	Self- woul they time both

Value of a Article - internal use	3	**X**	CRM, Web	Score	Asse calcu activ when weig impa quick
Value of a Article - Web use	3			Score	Asse Exan the la servi analy
Customer sat w/ KB use Vs without KB use	4	**X**	Survey and CRM/KB		Incid satis the in was
Self-Service Success					
Customer use of web first	3		Survey, web analytics	%	% of first, Meas samp
Customer success on the web	3		Survey, web analytics	%	% of and s a sur
Customer visit w/o incident opened	3			%	Cust open rang this i in a s withi

Value of web						
Triangulation method						Asse one r value web a tru
1. Click stream analysis	2			web analytics	%	First going succ
2. Customer experience	2			Survey	% satisfied	Seco custo
3. Case/incident volume	2			CRM, financial reports	#	Third Case of lic
Community Success						
%posts with community response	3		**X**		%	Indiv
Time to response	3		**X**		Avg. minutes	
Health of community	3	**X**	**X**	Survey	Index	Leve
Reach	4	**X**		Network analysis	Index; size and diversity	Asse Two audie of the

Relevance	4			Network analysis, survey	Index	Asse often are r
Loyalty						
Customer loyalty	3	**X**		Survey	Score	See
Renewals	3	**X**	**X**	CRM tool	%	
Employee loyalty	3	**X**	**X**	Survey	Score	Loya custo
Collaboration/ team health	3	**X**		Survey	Score	
Employee turnover rate	3	**X**		HR reports	%	
Community health	3	**X**		web reports/surveys	Score	Onlin
Organizational Learning						
Time to fill knowledge gaps on the web	3	**X**		Web analytics, click stream analysis	Avg min/days	
% of issues promoted by support implemented by Development	4	**X**		Manual	Issues promoted Vs implemented	Indic Mgm

Time to cure (time from id to removal of problem)*	4	**X**	CRM, KM and release dates		Supp mana to im expe
Time to proficiency - new Analysts	2	**X**	Manual	weeks/months	Curre
Time to proficiency - experienced Analysts, new products/ technologies	3	**X**	Manual	weeks/months	Curre
Time to equilibrium* (new release)	4	**X**	CRM+Web+community reports and product installed reports	Trend - exception rate per installed product per week	New
Time to adopt/ install	4	**X**		Trend, install rate of new release/ product	Cust of tim
Financial					
Total support costs as a % of total company revenue	3	**X**			
Support margins (contract rev)	3	**X**	Financial systems	%	Supp base
Cost/exception	3	**X**		$	Acro exce

Cost/incident (assisted)	2	**X**		$	Supp
Cost/incident - known (assisted)	3	**X**		$	Supp
Cost/incident - new (assisted)	1-3	**X**		$	Supp

Appendix D: KCS Roles and Competencies

KCS Candidate (KCS I)

Topic	KCS Candidate - Range of Knowledge Describe/Explain/Demonstrate
Incident management and knowledge management functions	Call management is for the incident related information needed for call administration; knowledge management is for the reusable elements of the problem solving experience. Identify where pieces of information belong: • Customer name, contact, contract/entitlement, severity level are all call/incident related • Problem description, relevant environment information, the answer/fix to the problem and cause information are reusable and go in the knowledge base
Knowledge and the purpose of a knowledge base	Knowledge is actionable information; it is a collection of data that describes activities that will produce a desired outcome. The knowledge base complements the knowledge worker's experience, use of a knowledge base requires judgment and skill, and a knowledge worker should never deliver an article to a customer that they do not understand. A knowledge base is the collection of experiences to-date of the organization; at any point in time it represents the best understanding of what we have collectively learned.
The concept of an "article"	An article is: - The name we use for the knowledge object - The place we capture the problem solving experience

Topic	KCS Candidate - Range of Knowledge Describe/Explain/Demonstrate
	Articles contain the problem description as experienced by the requestor, information about the environment in which the problem occurred, answers, fix or work-around for the problem, and the cause of the problem Articles have a life cycle, at the outset they may only contain a description of the problem (Work in Progress), when the problem is resolved they contain the fix/answer and the cause (Validated) Articles are dynamic; they are constantly being updated through use. "An article is complete when it is obsolete"
KCS, the workflow and the structured problem solving process	KCS is a problem solving methodology that includes searching and updating a knowledge base. Capture individual experiences in solving problems to create a collective/organizational memory.
Capturing the requestor's experience in the workflow	Capturing the requestor's experience, in their terminology, is critical for future findability Literal element of the structured problem solving process
Searching techniques	First capture requestor perspective and search using requestor language Use your own words to refine the search Keyword searching and Boolean commands Queries, looking for criteria fit, date range, created by, status Natural language searching Associative searches Browsing
Content structure - the power	Identify good content structure, in the context (vocabulary) of the target audience

Topic	KCS Candidate - Range of Knowledge Describe/Explain/Demonstrate
of context	• Correct - Separate problem content from environment content • Concise - complete thoughts, not complete sentences • Clear - independent thoughts, not multiple thoughts The goal is findable, usable articles
When to initiate a search	Gathering sufficient information, a description of the problem and a few words/phrases about the environment. Search early, search often. This ensures you are not working on a problem that has already been solved.
When to STOP searching	When the search statements have been refined, the problem statement is complete and we have collected 2-3 characteristics about the environment that are believed to be relevant. If at this point the search response is not providing anything that appears relevant, then it is time to move into the analysis phase of problem solving.
Concepts of the content standard and article structure	Basic types of content • Problem description - symptoms, unexpected results, error messages, goal or description of what they are trying to do. The resolution answers/resolves the problem description • Environment - products involved (hardware, software, and networks) release or version, recent changes to the environment. The environment statements do not change when the problem is resolved. • Resolution - the answer to the question, a work-around, circumvention or by-pass, fix. • Cause - background reasons for the problem or question (optional)
The concept of reuse and the value of tracking reuse	Reuse of articles in the knowledge base drives: • Identification of content that should be made available to a wider audience • Identification of issues that need to be addressed by product or application development • Identification of process failures

Topic	KCS Candidate - Range of Knowledge Describe/Explain/Demonstrate
Structured Problem Solving (SPS)	Key elements of the Structure Problem Solving Process • Manage the call/conversation; deal with the administrative elements at the beginning (call initiation) and end of the call (wrap up). This will allow focus on the customer's objective of problem solving. • The SPS process [admin....Literal .... Diagnostic .... Research ...admin] • The SPS process involves application of a methodology for collecting, organizing, and analyzing details which develops a constructive outcome. The end-point should be an understanding of the situation and a resolution or answer.
The dynamics of article reuse	Reuse of articles is generally a good thing, however: • Low levels of reuse can be an indicator that the articles are not findable due to structure issues or problems with the search algorithms • High levels of reuse can be an indicator that the sources of the exceptions are not being removed from the environment.
Create a new article vs. reuse an existing one	Two key points about creating a new article vs. updating an existing article. • Article creation should occur when a unique entity is required to address a set of circumstances not yet documented in the KB • A newly created article may or may not be complete, but it adds value to the knowledge-sharing process
Article meta data and concepts of the article life cycle	Article creation involves adding attributes to a article that help organize the KB content, control visibility, and facilitate assessing the value of article entities. Managing both data and metadata is required for effective article creation.
Understands the organizational value of KCS, can explain the benefits of sharing knowledge	Benefits to each of the three stakeholders Responders - less redundant work, recognition for problem solving skills, individual learning and the learning of others. Confidence in working on new areas/technologies Requestors - speed, accuracy and consistency of answers Organization - cost savings through operational efficiencies, increased customer

Topic	KCS Candidate - Range of Knowledge Describe/Explain/Demonstrate
	loyalty

KCS Contributor (KCS II)

All of the KCS Candidate competencies plus the following:

Topic	KCS Contributor - Range of Knowledge Describe/Explain/Demonstrate
Article quality	• Consistently creates articles that do not require rework (based on performance in the environment) • Collective ownership "if you find it/use it, you own it". It is critical that the users of the knowledge take responsibility for what they see and use in the knowledge base - If an article is unclear they should "flag it or fix it." • Article review processes in the workflow and random sampling • Concepts of findability and usability, criteria for a good article; key things to look for: ◦ Correct - words and phrases are in the right place (problem vs. environment) ◦ Concise - complete thoughts not complete sentences ◦ Clear - single thoughts not compound thoughts • Requestor requirements are speed and accuracy
Improve, modify concepts	• The balance of diversity and consistency: problems should be described in as many ways as requestors will experience them, the environment should be described in a standard/consistent way. • Sensitivity to personal preferences and style differences vs. good statement structure and the quality requirements that support usability and findability ("sufficient to solve") • Don't over generalize. Article should evolve through use and should be specific to the experience of solving a requestor's problem. Generally,

Topic	KCS Contributor - Range of Knowledge Describe/Explain/Demonstrate
	attempts should not be made to extend articles to cover all possible situations that might occur. Article extension should be based on demand. • Ideally, there should be one article per problem. However, this is not an absolute and the criteria should be developed based on experience in the environment. Some exceptions that need to be considered are: ◦ Context - two articles may exist for the same problem but are targeted at different audiences (novice vs. expert)
Managing Article Audience	• Articles that are reused are candidates for a larger audience; they should be moved closer to the requestor. • It is important that not everyone be able to see everything that is in the knowledge base, visibility should be appropriate to the audience
Concepts of context	• Context - vocabulary and technical perspective/capability of different article audiences • Articles are created in the context of a specific audience
Fix/answer description format and context of the audience	• Balance between completeness and usability/brevity • Using numbered steps to describe a resolution process • Must be in the vocabulary and technical perspective/capability of the target audience (context)
Capture in the workflow and Structured Problem Solving	The value of capture in the workflow • Capturing the customer context, if not done during the conversation it will be lost. • Capturing the problem and some environment information in the workflow enables the "search early, search often" practice. This reduces the risk of spending time solving a problem that has already been solved.
Relevant vs. non-relevant statements	• The need for judgment in reviewing articles, customers will often

Topic	KCS Contributor - Range of Knowledge Describe/Explain/Demonstrate
	provide information that has no relevance to the situation.
Issues of redundancy	• A certain level of redundancy and diversity in a knowledge practice is healthy. Redundancy becomes a problem only when it adversely affects the findability and usability of the content. • Examples of acceptable redundancy ◦ Articles for the same situation but for different target audiences ◦ Articles that capture wholly different experiences but have the same resolution • The content standard should describe the criteria for unwanted redundancy and as redundant articles are found they should be merged into one.

KCS Publisher (KCS III)

All of the KCS Contributor competencies plus the following:

Topic	KCS Publisher - Range of Knowledge Describe/Explain/Demonstrate
External audience(s)	Understanding of the audience(s) for external content and their article quality and context requirements for each external audience: • Partners • Customers

KCS Coach

All of the KCS Publisher competencies plus the following:

Topic	Coach - Range of Knowledge Describe/Explain/Demonstrate

Concept of a KCS Coach	• KCS Practices expert
	• Change Analyst
	  ◦ Support and encourage learning the KCS Practices
	  ◦ Provide constructive feedback on work habits and articles created
	  ◦ Participate with other Coaches and the Knowledge Domain Experts on developing improvements to the workflow, the content standard and lifecycle, and identifying requirements for the infrastructure (tools/technology)
	• Monitor leading indicators (activities) for individuals - article creation, reuse and modify rates
	• Goal of the Coach - move people along the KCS path to become KCS II so that they can consistently create articles that do not need review or rework

Influence skills	• Fundamental principles of motivation for people - the two top motivators for people are a sense of achievement and recognition • Respect for the knowledge worker and the learning process • Mindful of the feelings of the knowledge workers • The power and benefit of collaboration - sharing what we each know gives us access to what we all know.
Article lifecycle	• Articles are intended to capture the collective experience of the organization and ultimately the customer. • An article has a lifecycle because at its inception it will only contain the question or issue that has been identified, it must be designated as a "work-in-progress" so its visibility is limited • Capturing everything in the knowledge base enables collaboration independent of space and time
Article quality	Criteria for reviewing article quality • The balance of speed and accuracy with article "beauty", articles only need to be "sufficient to solve" (to be found and useful) • The importance of the content standard • Good structure - complete thoughts not complete sentences, distinct thoughts

Topic	**Coach - Range of Knowledge** **Describe/Explain/Demonstrate**
	• Article states and the relationships between article audience, article confidence, and article governance
Capture in the workflow	• Can model it and teach others how to do it.
Dealing with objections	The top objections to KCS and the responses: • Can't capture in the workflow • Don't have time to create articles • Dumbing down my job • Giving away my value

KCS Knowledge Domain Expert

All of the KCS Publisher competencies plus the following:

Topic	**Knowledge Domain Expert - Range of Knowledge** **Describe/Explain/Demonstrate**
Role of the Knowledge Domain Expert	Health and continuous improvement of the knowledge base or a collection of articles in the knowledge base • Redundancy or overlap of content • Content gaps • Overall article quality • Article reuse • Article evolution/cycle time Health and continuous improvement of the KCS process and practices within the organization

Topic	Knowledge Domain Expert - Range of Knowledge Describe/Explain/Demonstrate
Concept of a collection or domain of articles	Articles associated with a technology or group of products that have the potential to be related to one another.
Pattern and trend recognition	• New vs. Known analysis • Identify articles with high reuse and initiate action to remove the source/cause of the issue • Interacts with and provides actionable information to product/ application development based on article reuse
KCS process/standards improvement	• In conjunction with the Coaches seeks to improve the KCS processes and content standards • Role and need for a global KCS Council • Accessible and responsive to suggestions from KCS Contributors, Publishers, and Coaches on improvements to the content standard and processes
Synonym concepts	• Define the power and risk associated with creating synonyms in the search facility
Article audience model	• Define who should have visibility to what

Appendix E: Article Quality Quick Reference Guide

Following is an example of an Article Quality Quick Reference Guide that should be tailored to the environment.
Download the editable Word Document here.

Issue/Question

As a matter of usability, it's helpful to see clear, unique issues or questions when searching for articles.

Issue: tell us what is happening in the requestor's words.

1. Don't create "compound statements" - keep the **environment terms** out of the **issue** if you can.

   No: Problem: 3Com NIC X1000 has the following error message: Comu.dll triggered an error in an invalid page in the module Comu.dll.

   Yes: Problem: Error: "Comu.dll triggered an error in an invalid page"

   Environment: 3Com X1000 Module – Comu.dll

2. Make the thoughts complete:

   Yes: Issue: Program crashes on startup with an error.

Environment Information

Naming platforms, products, versions, and/or functions.

Environment information should be formal and detailed, including as much information as necessary to *uniquely* identify the environment being described.

- *<Vendor> <Product>*, version *<Version Number>*

Examples of Good Environment Information

- OS X Yosemite version 10.10.5
- Microsoft Office 2016
- MacBook Air

Environment Information helps Classify Problems

1. Do ***not*** put *multiple* Environments in a *single* statement.

The Cause (optional)

There should be only one cause per article. If an article has more than one cause, it is likely that it should be multiple articles.

If you must decide between applying one fix statement or another (because only one will work for your customer), the article should be split into two!

Article Audience

Indicates who can see which articles.

Internal: Visible only to users identified as employees

Partners: Visible to people who are not employees but act as trusted extensions of the organization

Issue: Error: “Program crashes due to insufficient memory”

Error Messages

Error: "<exact error message text>"

Error: “Cannot start program. Required application not recognized”

Ordering Issues
If your article has multiple issue statements, order them in the article as follows:

- Less detailed first (generic)
- More detailed to follow (specific)

Example:

- Cannot print a file
- Error printing file to network printer
- Error: “Invalid page layout for this printer driver. (24301)”

The Most Reusable Error Structure

To structure error-statement-type issues for the greatest opportunity for reuse, structure Problems by breaking them into two "modular" statements:

1. a general statement that an error occurs, and the

2. Modify existing articles to **add** new Environments as needed:

Changes in the Environment

1. Think about what the user may have done:
   - Change: Installed the update to the software
   - Change: Reset the counter to zero
2. Changes are not the *cause* - don’t confuse the two.
3. Do not jump to conclusions:
   1. **No:** Change: *It worked before we replaced an XBTVA with an XBTVM*
   2. **Yes:** Change: *Upgrade from Release 3.2 to 4.0*

Add unique environment statements to differentiate this article from others with similar symptoms but a different resolution

Resolution

Fixes or answers should address the problem or answer to the question

- The Fix statement clearly lists what **steps** to take to resolve the issue
- There can be multiple ways to resolve a problem, a formal fix or ways to workaround the situation, these can be documented in fix statements and should be labeled

Customers: Visible to customers or users of our products or services

Public: Visible to unidentified individuals in the public domain

Article Confidence Transitions

Indicates progression of article through stages of confidence.

Work in Progress: Represents work in progress, no fix or resolution has been identified.

Not Validated: An article the author considers complete but they do not have high confidence in the resolution (not yet Validated). Or, the author is a KCS Candidate and is not licensed to create Validated articles. Not Validated articles can be Validated through reuse.

Validated: Assigned to an article when a Contributor is confident in the resolution *and* the structure of the article. KCS Contributors and Coaches can validate articles.

Archived: This article is no longer

conditions during which it occurs
2. the specific error statement, with no conditional modifiers

Example:
If a requestor reports getting the following error:
Error: “Out of memory”
Error: “Error writing UDP packet 8101”
Error: “No document libraries available

Things you don’t need to write!
Certain phrases are unnecessary when writing statements:

- "I want to", "The customer is trying to"
- "The customer is using…"
- "The customer is getting…"
- "It worked okay before I…"

Just get to the point!

Verb Tense

Write in present tense: don’t tell us what you did, tell us what to do!

Use Explicit Subjects

Implicit Subject: Won’t print.

(Unclear - What won’t print?)

Explicit Subject: Documents do not print. (Better)

“workaround: “

- Fix statements should include active hypertext links to maintained web sites that are searchable by the users and in the context of the user.
- Use a link when helpful to point to existing documents or more details.

Structure of a Fix Statement

- Keep the whole fix within one "statement”. If several steps must be performed in order, number the steps.
- Use tabs for formatting and readability.
- Write everything as a present tense list of commands, as if you were reading them step by step to the customer.
- **Do not include “if-then” statements in Fixes.** This is an indication that you need two separate articles differentiated by the environment statements.
- The article may contain more than one fix statement - *but all Fix statements must be applicable.*

relevant. It is a candidate for archiving.

Article Governance

Indicates who can make changes to which articles.

Experience-based: Article is open to modification by licensed KCS users.

Compliance-based: Article creation and modification is restricted to designated individuals.

Appendix F: KCS and ITIL

Some service delivery organizations, especially in IT organizations, have adopted a framework called ITIL--the IT Infrastructure Library®. ITIL® was created by the government of the United Kingdom in the 1980s to provide a structure to help enterprises organize and improve their IT Service Management practices. Organizations that have adopted ITIL, or are considering ITIL, often wonder how KCS and ITIL might work together.

The short answer is that KCS and ITIL are extremely complementary. They both seek to improve service delivery and management, are technology agnostic, and recognize the value of knowledge.

Here are a few considerations for implementing KCS and ITIL together:

- ITIL is a framework, not a methodology. In other words, it describes the organization of tasks in service management, but isn't prescriptive about how to do them. By contrast, KCS focuses on particular aspects of service delivery--in particular, the integration of knowledge into the workflow--and is much more specific about what staff and organizations should do. In this way, KCS "plugs in" to the ITIL framework, providing specific guidance for organizations, especially in what ITIL calls Incident Management, but also with Problem, Change, Release, and Deployment, and Service Level Management.
- ITIL and KCS each define their own terms, and in some cases, these definitions aren't the same. For example, in ITIL, "knowledge" refers to all information in the Service Knowledge Management System. ITIL promotes linking incidents to records in the problem database and a known error database, whereas KCS takes a much more expansive view of the types and nature of knowledge that can be linked to an incident record. To avoid confusion, organizations should be clear about their language choices.
- Practitioners who adopt service management processes based on ITIL and KCS should include knowledge capture, structure, reuse, and improvement in the ITIL Incident Management process, as well as integrate knowledge into other ITIL processes.

ITIL® is a registered trade mark of AXELOS Limited.
IT Infrastructure Library® is a registered trade mark of AXELOS Limited.
The Swirl logo™ is a trade mark of AXELOS Limited.KCS

Appendix G: KCS FAQs

Technique 8.7: Communication is the Key describes the importance of a Communications plan for a successful KCS adoption. This generic list of FAQs can be used as a starting point for organizations who are building FAQs as part of their communications plan.

Q: How do I distinguish between two articles as to which is the correct one?

A: The KB does not replace people's brains; it complements their brains. People have to be able to recognize a correct answer when they see it. A responder should never deliver/apply an article they do not know enough about. They must have some level of certainty that it fixes the problem.

Q: We don't have time to document everything the customer says and everything we do to solve a customer's problem. If we continue to run our business with the same workflows, structure for content, and recognition practices, we cannot add the capture of the information and the experience of the customer without adding more time per incident. The savings of re-using some articles will not offset the incremental time spent on every article. What is to be done?

A: It turns out that reusable articles can be created as a natural byproduct of solving a problem without adding incremental minutes to the problem solving processes. In order to accomplish this we must change how we do things. We must examine the workflow, content structure, and the recognition practices such that it facilities both the problem solving processes and the capture/structure processes. We have learned that focusing on a structured problem solving process and capture in the workflow can improve the problem solving time.

Q: When is it appropriate to create a new article rather than reuse an existing article?

A: The simple answer is: when the fix or answer does not exist in the knowledge base, a new article should be created. If the fix/answer is found in the knowledge base, then the existing article should be updated to include any new information or context that has become known because of solving the problem. Because of this experience, the article is improved or modified with additional information. There are some cases where multiple articles exist for the same fix. For example, two articles may exist but one is targeted at a highly technical user and the other is targeted at a novice user. These articles should be linked in the system so that the relationship between them is known. However, the link is visible only to those authorized to see it. Most often, the highly technical article will be for internal use only and is not

visible to the novice audience. The content standard for the organization must address the criteria and mechanics of create versus reuse based on the capabilities of the technology being used.

Q: How do we get our knowledge workers to capture their knowledge in a database?

A: To encourage the knowledge workers to capture their knowledge we must consider the ways in which the organization values the individual. We must examine what creates heroes in our organization. If the individuals are valued for what they know and the knowledge they have, then they will not have an incentive to contribute to a knowledge base. If, on the other hand, individuals are valued for their ability to learn and to contribute to the learning of others, it creates an incentive to collaborate. If heroes are created based on their participation in a knowledge management practice, and if they are valued and recognized for their ability to solve problems and to contribute that knowledge to a knowledge base through capture and structure practices, then there is an incentive to contribute.

Q: How do we manage the quality of the knowledge? We do not have time or the resources to review all the articles we create.

A: The most efficient way to manage knowledge is through reuse. If everyone who interacts with the knowledge base accepts responsibility to contribute to the quality of the knowledge (flag it or fix it), then quality management becomes inherent in the system.

Q: How do we know which articles should be made available to others? If we create articles within one level of our support structure, how do we know which ones should be migrated out to other levels of our support structure?

A: By monitoring reuse of articles, we can identify which ones might have value to other parts of the organization. Frequently-referenced articles should be flagged and reviewed for relevance and context for other audiences.

Q: We are already doing knowledge management! We publish our knowledge in product documentation and on the web. We have tech writers who clean up the information and publish it. What is different about KCS?

A: KCS strives for new levels of efficiency and relevance in the creation of knowledge. KCS is a methodology that is focused on capture and structure of knowledge in the workflow and on the findability and usability of that knowledge by a target audience.

Q: Should we let people modify articles while we are still trying to solve the problem?

A: Incomplete articles lead to complete articles. Creating or framing an article and allowing others with the right

privileges to modify them enables collaboration and collective problem solving independent of space and time. (People in different locations and time zones can help each other solve problems).

Q: At what point in the process do we start capturing information about the problem in the article?

A: The capture of information should start as early in the process as possible. Ultimately, the capture process starts with the requestor. For example, if the requestor has done a search on the web site, the search information should be captured as part of the article.

Q: How should we recognize people for their use of the knowledge base? What is the one objective we should focus on?

A: The health of the knowledge management system is based on many factors. No one single indicator can be isolated.

Q: Why would I capture the customer's opinion on what is happening when I know the way they are describing it is not correct? Doesn't it contaminate the knowledge base?

A: The ability to classify words and phrases the customer uses about the problem enables us to capture the customer context and experience and distinguish it from the environment and fix information. This is critical for findability.

Appendix H: The "Why" Behind Each Practice

Created in the spring of 2023 by a group of Consortium Members who were looking for a way to describe what was important behind each Practice.

- Use these in a communication plan to get buy-in and understanding.
- Refer to these as a sanity-check while designing a KCS program for your environment.
- Each Practice links to its section in the Practices Guide for more detail.

The Solve Loop

Capture

Capture and share knowledge in the moment, and in the context of the question-asker, because otherwise the solution is hard to find (or recreate) later.

- Capture Knowledge in the Moment because that's when the context is clearest, and when we are able to access tacit knowledge.
- Capture the Requestor's Context because that's what improves recognizability and findability.
- Searching is Creating: that's where gaps are immediately identified.

Structure

Structure knowledge in a consistent way to ensure readability and usability.

- Use A Simple Template because it supports both usability and readability. Knowing where to look for, and where to put, specific pieces of information improves efficiency.
- We use Complete Thoughts Not Complete Sentences because simple is better. A list of single, complete thoughts improves article "capture-ability" and readability, especially for those users whose native language is not the language in which the article is written (it also supports machine translation).

Reuse

Reuse knowledge to avoid rework and identify pervasive issues that are candidates for corrective action.

- Search Early, Search Often because we want to stop solving the same problems over and over: search first so that we aren't doing work that has already been done. Let's make use of the latest knowledge that's been captured.
- Seek to Understand What We Collectively Know because no one knows everything! Now we have a way to access the collective wisdom.
- We Link (or Attach) because we want to glean insights about the questions we're being asked. We want to know what knowledge we're using so we can understand the big picture of what is happening (and make improvements)!

Improve

Improve knowledge while interacting with it to keep what's getting used up to date.

- Reuse is Review means we're only paying attention to knowledge that is getting used, while we have the context to improve it.
- Flag It or Fix It because we own the knowledge collectively! Apply the campsite rule: leave knowledge in at least as good a state as you found it.
- License to Modify: this is how we enable people to fix the knowledge they're interacting with. We own (and maintain) the knowledge collectively!

The Evolve Loop

Content Health

Content Health gives us a way to communicate what we want the evolution of our knowledge to look like.

- We use a KCS Article Structure because we are capturing for (re)usability and readability. The article structure provides consistency for both readers and writers, and supports reporting, data driven insights, and allows us to "chunk" content for use in other ways (including machine translation).
- KCS Article States allow us to manage and identify the quality of the article we're looking at (from the perspectives of audience, compliance, and confidence) and give us a way to capture our experience regardless of its completeness.
- Developing a Content Standard is important because it allows us to maintain consistency in process, voice, and structure (and keeps us out of legal trouble). It defines what "good" looks like for us and acts as a guide for training.
- Creating Evolve Loop Articles is important because we want to look at trends and patterns to create/improve/promote often-visited or timely (or other high-value) articles.
- Archiving Old Articles gives us a way to maintain relevance of the knowledge base.
- The way we Deal With Legacy Data is important because we don't want to import unstructured content without context to our knowledge base: it's not searchable or usable.
- The way we Prime the Knowledge Base With New Information is important because it models how we want to create content going forward: based on the context of use, and rapidly fixing when we receive feedback.
- Global (multi-language) Support Considerations are important because it sure has a lot to do with context! Customer context + demand driven will identify need.
- Knowledge Domain Analysis work is important because we want to be identifying and taking action on the things we're learning in the Solve Loop. This analysis captures the complexity inherent in our processes and tools that the

Solve Loop has difficulty identifying.

- Content Health Indicators are important because we want to know how often we did the right thing. These indicators help us identify how we can better follow the Content Standard and workflow.
- Self-Service Success should be encouraged because we want our requestors to have access to answers in the easiest way (and to stop solving the same problems over and over)!
- Self-Service Measures are important because we want to know how well requestors are able to access our known answers. An improvement in Self-Service Measures is one way we know we are attaining our intended outcomes.

Process Integration

Process Integration says we want to integrate knowledge management activities into our workflow and make that workflow easy, intuitive, and repeatable through the functionality and integration of our tools.

- Structured Problem Solving is important because it reinforces the use and development of knowledge in the workflow: this is the way we do our work. It enables a level of consistency and effectiveness in the workflow and is reflected by quality and ease of reuse.
- Seamless Technology Integration is important because our tools need to facilitate the way we want to work. We want to make the right thing to do the easiest thing to do. Integrating tools reinforces the integration of knowledge as the way we are solving problems.
- Search Technology for KCS is important because it facilitates the ability to find and reuse knowledge based on the customer context.
- Closed Loop Feedback for the Whole System is important because we want to know what we are learning from both the Solve Loop and the Evolve Loop. It informs our capabilities to iteratively improve in reaction to what is happening (demand-driven!). This includes both how we can improve AND what we are doing well (visibility to impact is what keeps the loops going).
- KCS Process Adherence Review is important because we want to know how often and how well we are contributing to the Solve Loop. How often did we do the right thing?

Performance Assessment

Performance Assessment allows us to promote learning and growth for individuals and teams and assess the success of the KCS program.

- KCS Roles and the Licensing Model is important because it provides a framework for proficiency demonstration and responsibility distribution. KCS is not a free-for-all: the licensing model is a risk mitigation strategy to make sure that everyone in the system understands their ownership of the knowledge base.
- Coaching for Success is important because we can't improve in a vacuum. It helps change stick; coaches are change agents. Coaching sessions are an opportunity to practice new skills IN THE CONTEXT of doing the actual job.
- Assessing the Creation of Value is important because there is no one number that will let us know if we are being successful. We want to be able to identify individuals and teams who are doing the right things and contributing to our intended outcomes.

Leadership and Communication

Leadership and Communication promotes an understanding of why KCS is important and ensures those involved understand how their participation and contribution makes a difference in the context of the big picture.

- Developing and Communicating a Vision is important because we're looking to establish a common ground/goal and create organizational alignment and sustainability. "We want everyone to be excited about the same thing, over the long-term, and which empowers us to work smarter."
- Creating a Strategic Framework is important because we want to explicitly describe how KCS outcomes get you to organizational goals. How is KCS going to help? Write down (and talk about) the desired big picture.
- Identifying KCS Benefits and ROI is important because we want to know that the system is working as intended: we want to prove that we are having the impact we thought we would!
- Promoting Teamwork is important because knowledge-sharing is a mindset, and KCS is a team sport.
- Tapping into Internal Motivators is important because judgment is required. KCS is inherently a change management initiative: sustained change has to come from within.
- Recognition Programs are important because we want to celebrate what is working. Recognition programs SHOULD BE DESIGNED to support intrinsic motivations.
- Communication is the Key because we're looking to establish a common ground/goal. Making and sustaining change is driven by communication.
- Leadership Accountability to the Knowledge Worker is important because keeping our promises to the knowledge workers is how we build trust, and how we make sure the change sticks.
- Leadership and Communication Indicators are important because they allow us to 1) receive feedback 2) demonstrate that our KCS activities are resulting in the outcomes we want

Special Thanks

This project was born out of a Consortium Member session called "KCS with No Staff, No Funding, No Time, and Bad Tools" and reflects the collected experience of the following people. Thanks to each of them for their perspective and feedback!

- Tamara Amlung, pro accessio
- Arnfinn Austefjord, Consortium
- Denise Baldwin, SDL
- Jennifer Crippen, DB Kay & Associates
- Arun Daniel, Dell
- Judith de Jong, SDL
- Karen Eisen, Alida
- Sara Feldman, Consortium
- Victor Feyen, Topdesk
- David Hahn, Humana
- Bret Halford, SAP
- Jenna Hoffman, Intel

- Gerald Joiner, Humana
- Terry Kim, SDL
- Kelly Murray, Consortium
- Mitzi Newland, PAR
- Jason O'Donnell, Autodesk
- Nikki Peck, Humana
- Matt Seaman, Consortium
- Christian Tobbe, Humana
- Carla Verwijs, Sequal
- Jacob Watts, PAR

RIGHT TO USE WITH ATTRIBUTION

KCS v6 Practices Guide by the Consortium for Service Innovation is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License. (Please follow the link for more information and definitions.) For information about commercial use or any permissions beyond the scope of this license please email info@serviceinnovation.org.

You are free to:

- **Share** — copy and redistribute the material in any medium or format
- **Adapt** — remix, transform, and build upon the material
- The licensor cannot revoke these freedoms as long as you follow the license terms.

Under the following terms:

- **Attribution** — You must give appropriate credit, provide a link to the license, and indicate if changes were made. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.
- **NonCommercial** — You may not use the material for commercial purposes.
- **No additional restrictions** — You may not apply legal terms or technological measures that legally restrict others from doing anything the license permits.

Notices:

- You do not have to comply with the license for elements of the material in the public domain or where your use is permitted by an applicable exception or limitation.
- No warranties are given. The license may not give you all of the permissions necessary for your intended use. For example, other rights such as publicity, privacy, or moral rights may limit how you use the material.

Service Mark

The KCS® methodology is a registered service mark of the Consortium for Service Innovation.

The first mention of KCS in a written work must include the superscript ®. Please also include this footnote or parenthetical statement: "KCS® is a service mark of the Consortium for Service Innovation™."

The correct use of KCS is as an adjective, for example: “[Company Name] endorses the KCS® methodology….”

KCS cannot be used in the name of an offering without explicit written permission from the Consortium for Service Innovation. Please contact the Consortium with questions.

KCS v6 Practices Guide by Consortium for Service Innovation is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License. Permissions beyond the scope of this license may be available at info@serviceinnovation.org.

Special Thanks

With gratitude and appreciation to the members of the Consortium for Service Innovation.

KCS v6 is based on more than 25 years of collective thinking, investment, experimentation, and insight.

The journey started in 1992 with the simple premise - "What if we could capture, structure and reuse the support experience?" It is impossible to recognize all the contributors to the collective effort of developing and validating the KCS practices over the years. The early work on the KCS concepts and design was facilitated and documented by John Chmaj and Livia Wilson.

With continued focus and support from the Consortium members the practices continue to evolve. Version 6 of the KCS Practices Guide reflects the latest member experiences.

KCS v6 was written by Greg Oxton, Melissa George, and Kelly Murray based on the experiences and contributions from many Consortium members. Special thanks to David Kay for his contribution in many of the techniques and for his editing and comments on the entire document.

Thank you to the KCS v6 Practices team:

Arnfinn Austefjord

Russ Brooks

Monique Cadena

Jennifer Crippen

Amy Dotson

Beth Haggett

David Kay

Kristin Kline Hunter

Jennifer MacIntosh

David Stanley-Jones

Devra Struzenberg

For a summary of the updates that were made in v6 please see Appendix A