Appendices
## APPENDIX - A

### KNOWLEDGE ACQUISITION FRAMEWORK GRID

<table>
<thead>
<tr>
<th>Concept</th>
<th>Task</th>
<th>Problem-Solving Method</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Determine domain area</td>
<td>Talk to the domain expert and find out what problem-solving process may be amenable to automation. Perform evaluation analysis to determine if the domain area is likely to use a knowledge base system.</td>
<td>Perform evaluation analysis using the checklist approach.</td>
<td>Qualified domain areas for the construction of a knowledge base system.</td>
</tr>
<tr>
<td>Decompose the knowledge</td>
<td>Identify the type of knowledge being gathered (i.e., tacit, explicit, declarative, procedural). Break the domain into smaller subtasks. If the knowledge is contained in a database, apply data mining techniques to extract the knowledge.</td>
<td>Several methods can be used here. To gather tacit knowledge the use of decision trees is recommended. To obtain knowledge from databases, the use of data mining techniques are highly desirable. Each method will facilitate the knowledge being broken into smaller, more manageable subtasks.</td>
<td>The knowledge of the domain area is broken into smaller subtasks.</td>
</tr>
<tr>
<td>Determine interdependencies.</td>
<td>Analyze how individual components of knowledge are related and integrated when they are used to solve a task. Point out missing pieces of knowledge. Determine what pieces of knowledge are related and how. Detect inconsistencies among the various aspects of the knowledge.</td>
<td>Incorporate the use of Interdependency Models (IMs). Tools such as EMeD (Expert Method Developer) allow users to specify problem-solving knowledge and identify the interdependencies between the various aspects of the knowledge.</td>
<td>Missing pieces of data are filled in, inconsistencies are corrected, and a more complete view on how the knowledge components are related is established.</td>
</tr>
<tr>
<td>Concept</td>
<td>Task</td>
<td>Problem-Solving Method</td>
<td>Result</td>
</tr>
<tr>
<td>--------------------------------</td>
<td>----------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------------------</td>
<td>------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Recognize knowledge patterns</td>
<td>Identify patterns within the various types of knowledge gathered. Apply the knowledge gathered to similar situations that the domain expert may have encountered or to similar cases (CBR) that may have been solved previously.</td>
<td>Using CBR, developing decision trees, or data mining techniques to discover patterns is the recommended approach.</td>
<td>The knowledge becomes more consistent and the knowledge base is able to be constructed more efficiently.</td>
</tr>
<tr>
<td>Determine judgments in knowledge</td>
<td>Determine if the knowledge being captured is judgmental (uncertain or fuzzy) in nature. A determination must be made if there are conflicts between rules.</td>
<td>If we have conflicts between rules, we should ask the expert for an explicit, relative judgment between the active rules in that context.</td>
<td>Conflicting rules are identified or conflicting knowledge about an area of a domain is identified.</td>
</tr>
<tr>
<td>Perform conflict resolution</td>
<td>If the knowledge being captured has uncertainty or is fuzzy in nature, the first step in resolving this uncertainty is to specify preconditions in the contexts of one or more of the conflicting rules to prevent them from being considered.</td>
<td>The effect of this is to add expertise, restricting the number of situations in which the rules are applicable.</td>
<td>Conflicts between rules or conflicts between knowledge are resolved.</td>
</tr>
<tr>
<td>Construct knowledge base</td>
<td>Perform knowledge modeling of the domain utilizing one or more of the techniques mentioned in the previous concepts.</td>
<td>Construct knowledge models utilizing UML.</td>
<td>Knowledge-based system KMS is constructed for the domain areas under consideration.</td>
</tr>
</tbody>
</table>
APPENDIX - B

KNOWLEDGE ACQUISITION TOOLS

Acquire

This knowledge-based authoring tool and expert system shell provides a step-by-step method for acquiring and structuring knowledge without the use of programming. Acquire® software, provided by Acquired Intelligence in Canada, is a knowledge acquisition system and expert system shell. It is a complete development environment for building and maintaining knowledge-based applications. It provides a step-by-step method for knowledge engineering that allows the domain experts themselves to be directly involved in structuring and encoding the knowledge. (The direct involvement of the domain expert improves the quality, completeness, and accuracy of acquired knowledge, lowers development and maintenance costs, and increases control over the form of the software application.) Features include a structured approach to knowledge acquisition; a model of knowledge acquisition based on pattern recognition; knowledge represented as objects, production rules, and decision tables; handling uncertainty by qualitative, non-numerical procedures; extremely thorough knowledge bases; sophisticated report-writing facilities; and self-documenting knowledge bases in a hypertext environment.

CommonKADS

CommonKADS is a leading approach to support structured knowledge engineering. It has been gradually developed and has been validated by many companies and universities in the context of the European ESPRIT IT Programme. It now is the European de facto standard for knowledge analysis and knowledge-intensive system
development, and it has been adopted as a whole or has been partly incorporated in existing methods by many major companies in Europe, as well as in the United States and Japan. CommonKADS enables developers to spot the opportunities and bottlenecks in how organizations develop, distribute, and apply their knowledge resources, and so gives tools for corporate KM. CommonKADS also provides the methods to perform a detailed analysis of knowledge-intensive tasks and processes.

The core of CommonKADS is formed by its knowledge analysis framework. CommonKADS provides all the tools required to analyze knowledge-intensive tasks at different grain-size levels. The analyst is supported in the modeling process by powerful templates, which constitute pre-defined reusable knowledge models that have been proven to work in the past. The templates enable a top-down approach and provide handles for quality control and feasibility assessment. The results of knowledge analysis are documented in the knowledge model. It contains a specification of the information and knowledge structures involved in a knowledge-intensive task. The knowledge model plays a key role in both KM work and in consecutive system-development activities.

ModelDraw is a drawing tool that can be used to create four UML diagrams—use case diagrams, activity diagrams, class diagrams, and state diagrams. The tool also supports the construction of CommonKADS knowledge modeling diagrams. The tool can generate diagrams in both Windows® WMF/EMF, and eps formats for inclusion in Word and other documents.

KADS22 is an interactive interface for the CML2 (Conceptual Modeling Language) knowledge modeling language. KADS22 provides an interactive graphical
interface (much like the familiar Windows programs) with the following functionality:
parsing CML2 files, pretty-printing, hypertext browsing, generation of the graphical
notation, search, glossary generation, and HTML (Hypertext Markup Language)
generation.

CML2 is the CommonKADS knowledge modeling language. The syntax
description and the parser correspond to the draft CommonKADS book as submitted to
the publisher. The CML2 parser is a standalone executable. The parser can be used to
check a file containing CML2 for syntactical correctness.

**Epistemics**

Epistemics designs, writes, and markets a number of software toolkits. These
toolkits aid knowledge engineers and those working in KM when performing a number
of key tasks associated with knowledge acquisition, analysis, modeling, validation,
publishing, and implementation. Products include the following: PCPACK4, SOPHx-
PACK PC, PACK 2, MetaPACK.

**Expect—An Integrated Environment for Knowledge Acquisition**

This integrated acquisition interface includes several techniques previously
developed to support users in various ways as they add new knowledge to an intelligent
system. As a result of this integration, the individual techniques can take better
advantage of the context in which they are invoked and provide stronger guidance to
users. The Expect project has focused on acquiring problem-solving knowledge for users
for the last decade, using an expressive language that is open to inspection. The aim has
been to alleviate the bottleneck in creating knowledge-based systems by providing
support for both knowledge engineers and end users to specify problem-solving knowledge. See http://www.isi.edu/expect.

**Protégé-2000**

The original Protégé was developed for domain-specific applications. Now in its latest version, Protégé 2000 is a modeling technique developed by Musen and colleagues from Stanford Medical Informatics. The Protégé 2000 knowledge modeling environment is a frame-based ontology editing tool with knowledge acquisition tools that are widely used for domain modeling. The frames are the main building blocks for a knowledge base. The Protégé ontology (that models the domain) has classes, slots, facets, and axioms.

Classes are abstract representations of domain concepts. “Classes in Protégé 2000 constitute a taxonomic hierarchy and are templates for individual instance frames.” A subclass can have all the instances of the class. Protégé 2000 allows multiple inheritance: a class can have two or more superclasses; it also supports a metaclass concept. Slots are properties or attributes of classes. There are two forms of slot. “Own slots define intrinsic properties of class or individual instance frames. Template slots are attached to class frames to define attributes of their instances, which in turn define specific values for slots.”

Slots are first-class objects in Protégé 2000; they can be used globally or locally. Facets are properties or attributes of a slot and are used to specify constraints on slot values. The constraints include slot cardinality (i.e., it specifies the number of values the slot can have), value type for the slot (such as integer, string), and minimum and
maximum values for a numeric slot. Axioms define additional constraints on frames; these may link values together or exploit knowledge interchange format (KIF)—based predicate logic.

Instances information is acquired using online forms. They are composed of a set of graphical entry fields and provide an easy-to-use user interface—an important feature of Protégé 2000. It automatically provides a form to acquire instances of a class when the user defines a class and attaches a template slot to it. The user can customize the form by changing the layout or changing the form's field labels and can choose different ways of displaying and acquiring slot values. The knowledge acquisition process in Protégé 2000 consists of three steps: First, a class and its template slot have to be defined. Second, the form to acquire the instances of the class has to be laid out. Third, the class instances are acquired.
APPENDIX - C

THE IMPLEMENTATION

Using the MVC architecture, the various beans, Servlets and JSP pages were developed. The various package diagrams are given below:

Figure: Package Article

Figure: Package usermanager
Figure: Package request

Figure: Package beans
The development of the application is an extensible framework for building and deploying knowledge bases that utilizes existing relational database management systems, text indexing engines, and customer databases. It allows users to post articles containing known defect and work-arounds if they exist. Its benefit is that it allows users to search articles thereby reducing the amount of time spent troubleshooting problems. Out of the box, sample deployments of the framework will be provided that can be customized to fit the deployment environment.

The default username is “Administrator” and the default password is “admin”. After logging in, the user is presented with the application’s main screen:
The main screen makes the knowledgebase’s main function available to the user. Administrators are responsible for managing the users of the system.

**Administrative Role - User Management**

**Adding Users**

A new user can be added by selecting the “create user” option. The Add User screen is displayed:

![Add User](image)

Users of the knowledgebase are assigned roles. These roles determine which functional areas in the navigation menu are accessible by users. A user is not restricted to just one role, but can have all three different roles assigned to him. The main purpose of role assignment is to involve various levels of approvals and reviews before an article is published.

**Delete Users**

Users can be deleted by selecting the “View All Users” or “Search for User” option from the Support Menu. Selecting “View All Users” displays a table listing of the users of the system. The rightmost column lists available actions such as view, edit, and delete.

![View User](image)
Selecting the delete option deletes the user from the knowledgebase.

**Edit Users**

Users can be edited very similarly as they can be deleted. Selecting the “View All Users” option from the Support Menu displays the “View All Users” list. Selecting the “Edit” option in the appropriate user’s row from the table’s rightmost column, displays the “Edit User Information” screen.

All properties except for the user id and username can be changed. The most important aspect of this screen is the ability to change a user’s roles.

**Searching Users**

The “Search for User” option allows the user to search for users. Users can be searched for based on username and email address. In either case, it is sufficient to enter a substring of the username or the email address. Pressing the search button searches for users whose username or email contains the substring specified.
If the user enters the substring “dav” in the username window and presses the search button, the search results are displayed:

The “david” substring matches “david”, hence we get the result displayed above.

Submitter Role

The submitter role allows the user to submit articles and categories to a publisher for approval.

Submitting a Category

Categories are used to organize the knowledge base. They can be thought of as directories in a file system. A category can contain articles and other categories. Follow these steps to submit a category to publishers:

1. Select the Create a Category option from the Support menu.
2. Complete the “Create a Category” form:
Title: The title of the article to indicate the contents of the article.

Description: A brief description of the article.

Category: The category that will contain this article if it is approved.

Keywords: Keywords that help distinguish this article from others.

Article Text: The actual content of the article.

Pressing the submit button displays the “Verify Article Information” screen:

Verify an Article Information
Title Manual Test
Description Manual Test

Figure: Verify Article

If the contents of the new article are verified to be correct, the submit button submits this as an actual request to the publisher.

View My Pending Requests

To verify that the category has indeed been submitted to publishers, the submitter can select the “View My Pending Requests”. This allows the submitters to view requests they have submitted.
Create a Category

Category name: 

Parent Category: [None]

Description: 

Figure: Create Category

The form asks for the new category’s name, its parent category, and a description of the new category. The description may contain the purpose of why this category was created. To verify that the category has indeed been submitted to publishers, the submitter can select the “View My Pending Requests”.

View My Pending Requests

Figure: View PR

Submitting an Article

Submitters can submit articles for publishing by selecting the “Create an Article” option from the Support Menu.

Create an Article

Title

Description

Categories: [None]

Keywords

Article Text

Figure: Create Article

The “Create an Article” form prompts the user for the following information:
Approving a Suggested Article

As a publisher, select the “View All Pending Requests” from the Support Menu. Selecting this option will display all pending requests that have been submitted by submitters.

The “View Pending Requests” screen lists the pending requests. These requests can be categories or articles. In a previous section, we saw how a category is approved. Approving an article consists of viewing the article by selecting the “view” action from the rightmost column, and accepting the request by selecting “accept”.

Searching

The ultimate value of the knowledge base is for end users who are able to search the knowledge base for articles. The search functionality is accessible from the following URL: http://localhost:8080/kb/jsp/searchArticle.jsp

Searching for an Article

The search screen offers the ability to search for articles by specifying keywords, what should be searched, and any date limits on the creation date of the article.
Publisher Role - View All Pending Requests

Submitting the request makes it available for approval or rejection by the publishers once the publisher selects the “View All Pending Requests” from the Support Menu. If the publisher accepts the article, it is immediately available for searching by end users. However, the publisher can also reject an article by choosing the reject option from the “View All Pending Requests” screen.

View All Pending Requests

<table>
<thead>
<tr>
<th>Request ID</th>
<th>Submitted Date</th>
<th>Type</th>
<th>Action</th>
<th>Publisher's Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>2002-05-09</td>
<td>Article</td>
<td>New</td>
<td>View, Reject</td>
</tr>
</tbody>
</table>

Figure: View PR

Approving a Suggested Category

Publishers are users with the publishing role assigned to them. Publishers are the authorities who approve or reject submitters’ suggestions. As a publisher, select the “View All Pending Requests” from the Support Menu. Selecting this option will display all pending requests that have been submitted by submitters.

View All Pending Requests

<table>
<thead>
<tr>
<th>Request ID</th>
<th>Submitted Date</th>
<th>Type</th>
<th>Action</th>
<th>Publisher's Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>2002-05-09</td>
<td>Category</td>
<td>New</td>
<td>View, Accept</td>
</tr>
</tbody>
</table>

Figure: View PR

Each request can be viewed, accepted or rejected by simply selecting that option from the publisher’s action column. Once the item has been accepted, it is stored in the database. Submitters can now submit articles to the knowledge base in the category.
Selecting the View action displays the following screen:

View a Category Request

| Request ID: | 4 | Submitted Date: | 2003-06-01 | Requested On: | | Requested By: | | My Actions: | View | Edit | Delete |

Edit a Category From My Rejected Requests

Rejected categories can be edited by selecting the "Edit" action. Editing a category allows the user to make changes to the request. When the changes are submitted, the request is immediately filed to the publisher for approval:

Delete a Category From My Rejected Requests

Rejected categories can also be deleted, by selecting the delete action from "Viewing my Rejected Requests."
Edit an Article From My Rejected Requests

The submitter is also allowed to edit an article when viewing the rejected requests. Any edits are re-submitted to the publisher for approval. The editing screen looks as follows:

![Edit Article](image)

Delete An Article From My Rejected Requests

Pressing the delete button from the My Actions column in the "View My Rejected Requests" deletes the request:

![Delete RR](image)

After deleting the request, the request is removed from the request queue.

View a Category From My Rejected Requests

Viewing categories from the rejected request list is similar to viewing articles.

The "Requested On" column indicates whether the request is for articles or categories.
View My Rejected Requests

Submitters can view and change their requests that were rejected. Browsing the rejected requests allows submitters to make corrections to their submissions and re-submit them.

A submitter can view rejected submissions by selecting the “View My Rejected Requests”. This sequence looks as follows:

The “View My Rejected Requests” lists all of the requests that were submitted by the currently logged on submitter and were rejected by a submitter.

The user can view, edit, or delete the rejected request.

View an Article From My Rejected Requests

Users have an opportunity to view rejected requests. This allows the submitter to review the article without modifying it.
By pressing the search buttons, the search criteria are submitted to the knowledgebase engine. Articles that match the search criteria are displayed on the search results screen.

**Search Results**

The search results screen displays a list of the articles that match the combination of criteria specified by the search conditions. The results screen looks as follows:

The ability to edit the article depends on the user’s role. End users can only view articles, but not edit them. In this case, the edit hyperlink is inaccessible.
### APPENDIX - D

**Glossary**

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Acquisition</strong></td>
<td>Accessing one or more remote sites and retrieving digital content</td>
</tr>
<tr>
<td><strong>Activity</strong></td>
<td>A behavior diagram that illustrates the flows among activities and actions associated with a particular object or set of objects</td>
</tr>
<tr>
<td><strong>Actor</strong></td>
<td>A coherent set of roles that an entity (human or nonhuman) outside of the system being modeled plays when interacting with one or more use cases</td>
</tr>
<tr>
<td><strong>Agent</strong></td>
<td>A system that searches for available information and filters incoming information based on specified characteristics</td>
</tr>
<tr>
<td><strong>Artifact</strong></td>
<td>Within the UML, an artifact is a classifier that represents a physical piece of information, such as a model, a file, or a table, used or produced by a software development process. An artifact can also contain other artifacts as part of a composition relationship. An artifact represents the manifestation of one or more packageable elements</td>
</tr>
<tr>
<td><strong>Artificial intelligence (AI)</strong></td>
<td>The use of computer algorithms, models, and systems to emulate human perception, cognition, and reasoning</td>
</tr>
<tr>
<td><strong>Association</strong></td>
<td>An association is a static (structural) relationship among two or more classifiers (typically classes). An association contains an ordered list of association ends. An association can have a name that describes the nature of the relationship. A link is an instance of an association. The UML defines two kinds of associations—binary associations and n-ary associations</td>
</tr>
<tr>
<td><strong>Case based reasoning (CBR)</strong></td>
<td>A branch of AI that attempts to combine the power of narrative with the codification of knowledge on computers. Involves extraction of knowledge from a series of narratives, or cases, about the problem</td>
</tr>
<tr>
<td><strong>Class</strong></td>
<td>A class is a named description of a set of objects that share the same attributes, operations, relationships, and semantics. These objects can represent real-world things or conceptual things. A class may realize one or more interfaces. A class can be either an active class or a passive class</td>
</tr>
<tr>
<td>Field</td>
<td>Value</td>
</tr>
<tr>
<td>---------------</td>
<td>------------------------------</td>
</tr>
<tr>
<td>Title</td>
<td>No Category Specified</td>
</tr>
<tr>
<td>Description</td>
<td>Missing Category</td>
</tr>
<tr>
<td>Categories</td>
<td><img src="image" alt="Error Message" /></td>
</tr>
<tr>
<td>Keywords</td>
<td></td>
</tr>
<tr>
<td>Article Text</td>
<td></td>
</tr>
</tbody>
</table>

Figure: Create Article
<table>
<thead>
<tr>
<th><strong>Class diagram</strong></th>
<th>A class diagram is a structure diagram that shows a set of classes, interfaces, or collaborations and the relationships among these elements.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Collaborative tools</strong></td>
<td>Tools such as groupware that enable both structured and free-flow sharing of knowledge and best practices. An example is Lotus Notes® software.</td>
</tr>
<tr>
<td><strong>Collaboration</strong></td>
<td>The term collaboration refers to a description of a structure of collaborating classifiers, instances of which each performs a specialized function (in other words, serve some role) that collectively accomplishes some desired functionality.</td>
</tr>
<tr>
<td><strong>Communities of Practice</strong></td>
<td>Aka affinity groups. (A) Informal networks and forums, where tips are exchanged and ideas generated. (B) A group of professionals informally bound to one another through exposure to a common class of problems, common pursuit of solutions, and thereby themselves embodying a store of knowledge.</td>
</tr>
<tr>
<td><strong>Data</strong></td>
<td>(A) Set of discrete, objective facts about events. Data is transformed into information by adding value through context, categorization, calculations, corrections, and condensation. (B) Facts and figures, without context and interpretation.</td>
</tr>
<tr>
<td><strong>Data acquisition</strong></td>
<td>Accessing one or more general databases and transferring data into a problem-specific database.</td>
</tr>
</tbody>
</table>
| **Data mining** | (Also known as knowledge discovery in databases—KDD.) Extraction of implicit, previously unknown, and potentially useful information from databases. The process uses machine learning, statistical correlations, statistical analysis, and sophisticated search strategies to extract data in such a way that the information is easily comprehensible. Then the human can decide how to turn this information into knowledge. The source databases are usually already owned by the organization. Marketing departments frequently used data mining to learn more about customers and how to better market products and services. The skilled knowledge manager will help create database search strategies that enable successful data mining. However, in some ways data mining is the
antithesis of what a knowledge manager is trying to accomplish in an organization. A knowledge manager sets up systems to store and retrieve information on a timely basis; a data miner seeks information in databases that was previously underutilized.

**Domain**
A field or area requiring expertise (e.g., physics, design, or manufacturing). It is often used to refer to the area of knowledge that is the focus of a particular knowledge acquisition project. A knowledge base will usually represent the knowledge in a domain or subdomain.

**Experience**
Refers to what we have done and what has happened to us in the past.

**Explicit knowledge**
Formal or codified, comes in the form of books, documents, white papers, databases, and policy manuals.

**Human capital**
The capabilities of the individuals required to provide solutions to customers.

**Inference engine**
Rule-based algorithms that interact with a knowledge base to draw conclusions about a set of inputs.

**Information**
(A) A message, usually in the form of a document or an audible or visible communication...meant to change the way the receiver perceives something, to have an impact on his judgment and behavior...it is data that makes a difference. (B) Patterns in the data

**Intellectual capital**
Refers to the commercial value of trademarks, licenses, brand names, formulations, and patents.

**Intellectual property**
Refers to the intangible or intellectual nature of works or creations and the body of laws governing such property; there are six areas of intellectual property: patents, trademarks, industrial designs, confidential information, copyright, and integrated circuit topography protection.
Intelligence
An ability to learn and understand new knowledge or reason in new situations.

IS A
Relation that denotes what class an object is a member of. For example, “Car—is a—vehicle” and “chicken—is a—bird.” It can be thought of as being a shorthand for “is a type of.”

Knowledge
(A) A fluid mix of framed experience, values, contextual information, and expert insight that provides a framework for evaluating and incorporating new experiences and information. It originates and is applied in the minds of knowers. In organizations, it often becomes embedded not only in documents or repositories but also in organizational routines, processes, practices, and norms. Key concepts of knowledge are experience, truth, judgment, and rules of thumb. (B) Actionable information. (C) A defined body of information ... depending on the definition, the body of information might consist of facts, opinions, ideas, theories, principles, and models (or other frameworks)...also refers to a person’s state of being with respect to some body of information. These states include ignorance, awareness, familiarity, understanding, facility, and so on. (D) The integration of ideas, experience, intuition, skill, and lessons learned that has the potential to create value for a business, its employees, its products and services, its customers, and ultimately its shareholders by informing decisions and improving actions

Knowledge acquisition
Eliciting and formally coding tacit knowledge into facts and rules and entering them in a knowledge base
**Knowledge architect**

The staff member who oversees the definitions of knowledge and intellectual processes and then identifies the technological and human resources required creating, capturing, organizing, accessing, and using knowledge assets. Architecture is the technology and human infrastructure to support the organization’s KM initiatives. It includes physical (e.g., hardware and tools) and logical (e.g., knowledge policies) dimensions.

**Knowledge assets**

Also called intellectual capital, are the human, structural and recorded resources available to the organization. Assets reside within the minds of members, customers, and colleagues, and also include physical structures and recorded media.

Knowledge audit: The formal process to determination and evaluation of how and where information knowledge is used within the organization. The audit examines policies, forms, procedures, storage, and any other ways that knowledge is collected, stored, and cataloged.

Knowledge base: A database containing tacit knowledge in the form of formally coded facts and if-then-else decision rules.

**Knowledge-based economy**

An economy in which value is added to products primarily by increasing embedded knowledge content and in which the content value evolves to exceed the material value.

**Knowledge bridge**

The connection that a KM expert builds between the business processes and the technological-, sociological-, personal-, financial-, sales-, creative-, and customer-oriented functions of the organization.
Building a knowledge bridge is the “glue” to make the long-term connections between the functions that sometimes are in competition for resources.

Knowledge content: The meaning that underlies data, information, knowledge, or wisdom.

Knowledge creation: The process that results in new knowledge or organizes current knowledge in new ways to make techniques to use existing knowledge. Once knowledge is created, the organization has a knowledge flow

**Knowledge flow**
The way knowledge travels, grows, and is stored. Knowledge flows (A) up and down from management; (B) within circles of sharing (such as shared interests between staff performing similar or complementary roles); (C) through planning, investigation, and training; or (D) through common sources such as books, reports, databases, or knowledge bases

**Knowledge facilitators**
Help harness the wealth of knowledge in the organization.

Facilitators engender a sense of ownership by those involved, by helping them arrive at a jointly developed solution

**Know-how**
The technical expression of knowledge. Examples are the manual and mental skills of a master craftsman or tradesman

**Knowledge interrogators**
Aka corporate librarian and knowledge integrator. The person responsible for managing the content of organizational knowledge as well as its technology. They keep the database orderly, categorize and format documents and chucking the obsolete, and connect the
Knowledge management

- Make an organization’s knowledge stores more accessible and useful.
- A business activity with two primary aspects: (1) treating the knowledge component of business activities as an explicit concern of business reflected in strategy, policy, and practice at all levels of the organization, and (2) making a direct connection between an organization’s intellectual assets—both explicit [recorded] and tacit [personal know how]—and positive business results.
- Conscious strategy of getting the right knowledge to the right people at the right time and helping people share and put information into action in ways that strive to improve organizational performance.

Knowledge map (K-Map)

- A tangible representation or catalog of the concepts and relationships of knowledge. The catalog is a navigational aid that enables a user to find the desired concept and then retrieve relevant knowledge sources.

Knowledge source

- The person, document, nonprint source, or place that is the origin or prime cause of knowledge.

Knowledge object

- A physical object used to support knowledge synthesis (e.g., a plant, insect, or rock collection).

Knowledge owner

- The person or people responsible for knowledge, a knowledge domain, or set of documents. The knowledge owner is responsible for keeping the knowledge and information current, relevant, and complete. The knowledge owner usually acts at a local or
decentralized level. The knowledge owner may or may not be the author or creator of the specific content. The owner may be the expert in the subject area or a skilled editor.

Knowledge processes: Organizational context, human activities, content value, information systems, and IT that are used to add value to content by increasing the amount of underlying processing and depth and breadth of meaning.

<table>
<thead>
<tr>
<th>Knowledge product</th>
<th>Knowledge that has been adapted to the needs of specific users.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Knowledge use</td>
<td>The effective integration of knowledge by people or organizations. It is the result of understanding and application of knowledge and the knowledge gathering process. It is hard to define because it is the result and application of all the terms defined on this page.</td>
</tr>
<tr>
<td>Knowledge worker</td>
<td>A member of the organization who uses knowledge to be a more productive worker. These workers use all varieties of knowledge in the performance of their regular business activities. Everyone who uses any form of recorded knowledge could be considered a knowledge worker. Alternative job titles for person in charge of knowledge management: Director of knowledge mobilization, Director of global knowledge exchange, and Senior Vice President of Strategic Knowledge Capabilities.</td>
</tr>
<tr>
<td>Ladder</td>
<td>A hierarchical (tree-like) network diagram. A ladder can comprise a</td>
</tr>
</tbody>
</table>
Laddering

A knowledge acquisition technique that involves the construction, modification, and validation of ladders. It is a valuable method for acquiring knowledge of concepts.

Node

A term used in a ladder or diagram to refer to an element that is not a link (i.e., is a rectangular or other shaped element). Each node can represent a knowledge object in a knowledge base.

Relationship

An instantiated relation (i.e., a relation that connects two specific knowledge objects).

Sharing

The human behavior that describes the exchange of knowledge.

Sharing and learning are social activities and may occur in face-to-face meetings or via written or visual stimuli. At least two people are required for sharing. Sharing knowledge is a positive activity in an organization. Coveting knowledge is the opposite of sharing. State: A named condition or situation in the life of an object that lasts for some finite amount of time, during which the object satisfies some condition, performs some activities or actions, or waits for some event.

Swimlane

A pair of parallel vertical or horizontal lines on an activity diagram used to delineate a partition.

Tacit knowledge

Information and skills that are not easily communicated and
documented (e.g., expertise, gut feel). An expert may not even be aware they use certain tacit knowledge when performing particular tasks.

(A) Knowledge developed and internalized by the knower over a long period of time... incorporates so much accrued and embedded learning that its rules may be impossible to separate from how an individual acts. (B) Informal or uncodified... found in the heads of employees, the experience of customers, the memories of past vendors... highly experiential, difficult to document in any detail, ephemeral, and transitory.

Use case specification

Defines a sequence of actions performed by one or more actors and the system, which results in an observable result of value to one or more actors. A use case specification always has one main flow of events (also known as a basic course of action and will generally have at least one exceptional flow of events (or alternate course of action).

Use case diagram

A use case diagram is a behavior diagram that shows a set of use cases and actors and the relationships among them.

Value proposition

The logical link between action and payoff that KM must create to be effective. Customer intimacy, product-to-market excellence, and operational excellence are examples.