

# Designing A Knowledge Management System

*Erez Krispin*

Many types of knowledge management systems are being designed and implemented nowadays in organizations worldwide. It is certain to say that no one system is similar to the other in its generic design (as opposed, for example, to ERP systems). Nevertheless, most knowledge management systems share many important aspects and scopes.

In this article we will analyze those aspects and scopes. We will try to do the impossible mission of defining what knowledge is (without assuming that everyone will agree on this definition). We will present a model for representing knowledge in knowledge management systems as part of the K-CIRmodel which is a framework for representation of knowledge items and a methodology for implementing a knowledge management system. We will overview some popular types of knowledge management systems and finally highlight some architectural aspects of these systems.

## I. What is Knowledge?

In order to understand what Knowledge Management is, we should first agree on the definition of the term knowledge itself. Many definitions for knowledge exist in the market. Some are qualitative, some are methodological and some of them are even conflicting. In this article we will stick to the evolutionary definition that addresses knowledge when processed by a computational model.

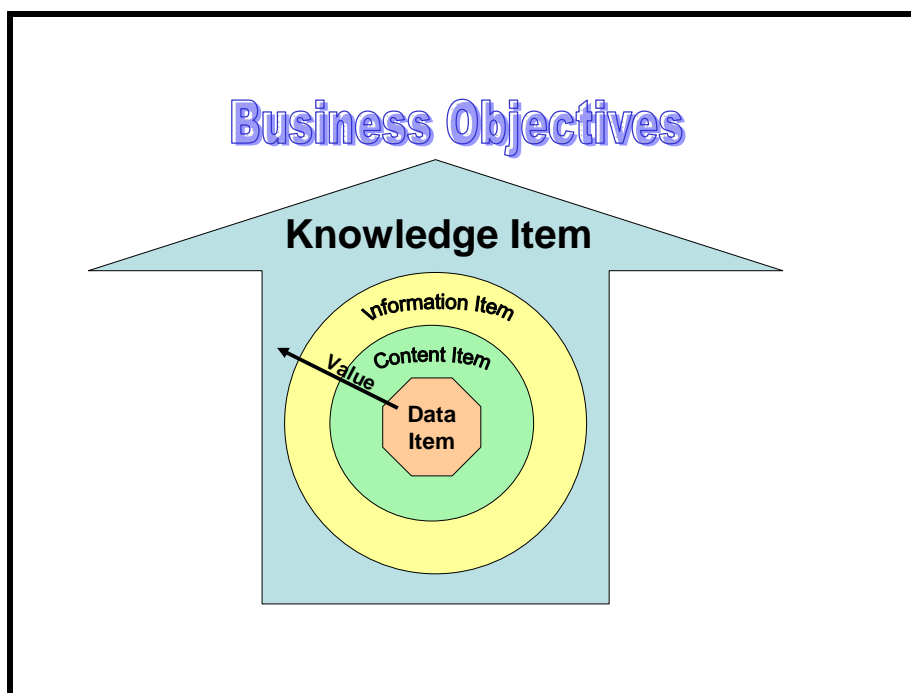
When we come to define the term knowledge, we should also regard three related terms – data, content, and information.

The term **data** usually refers to a discrete structured item that can reside in and fully managed by a database system. Examples for data are numeric fields, arrays, fixed length text strings, etc.

**Content** can be viewed as an evolution of data – any kind of item including unstructured, and not necessarily discrete. Examples of content are images, free text, streamed video clips and of course also data items. Although unstructured content can be also stored in a database, the database management system is limited in truly managing an unstructured content and therefore a Content Management System (CMS) is needed on top of the database. Content Management Systems (CMS) deal mostly with the manipulation of content items from the unstructured format point of view (smart storage and delivery, caching, etc.)

Data and content are mainly defined from the format point of view of the item and usually lightly relate to the understanding of its unique content and meaning. When such understanding is maintained, we say that the item is **information**, i.e. it informs us of something. For example, a numeric value data which is titled as Passport-ID field and being such, is manipulated by a certain information system can be regarded as information.

When we come to the term **Knowledge**, we try to dig in beyond the format and isolated meaning of a certain information item – we try to understand its nature and context within a broader scope. This advanced understanding is aimed at extracting from the information *an operational/actionable know-how value that will assist to the organization's decision making towards its business objectives.*

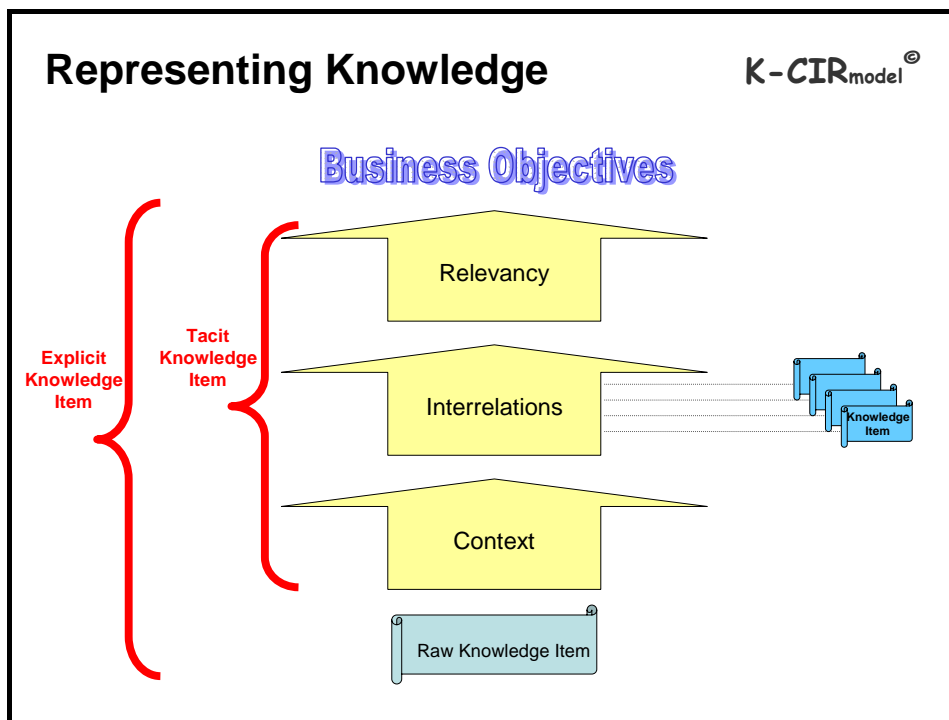


## II. Knowledge Representation

Given a raw information item that contains an actionable know-how, i.e. a knowledge item, we need a model according to which we will represent that knowledge item in our knowledge management system. K-CIRmodel is a such.

According to the K-CIRmodel, for each knowledge item, we should maintain, apart of the raw item, the following metadata to be attached to the raw item itself. This metadata is a scheme to describe the knowledge-nature of the item:

1. **Context:** information about the item’s identity, its context and classification. For example, a given streamed video clip knowledge item may be attached with metadata that indicates that it contains a speech of the company’s CEO, that the subject of the speech is the company’s annual results, etc.
2. **Interrelations:** based on that understanding, we have and maintain for such a piece of knowledge, its relations with other pieces of knowledge that share a certain theme under a certain context. For example, in that case of the video clip of the CEO, we will maintain relation of that item to all other items that relate to the CEO and/or to the company financial results, such as the CEO’s pictures, other speeches, biography text; the company’s underlying annual report document, etc. A set of items that interrelate and thus share a context is called *a category*. As part of the Interrelations criterion, we will also maintain the interrelations of the categories themselves. This type of interrelation is sometimes called Taxonomy (when having a tree structure), Ontology, Semantic Network, etc.
3. **Relevancy:** finally, and most importantly, we expect to have and maintain each item’s relations with the organization’s *business objectives*. After all, as an operation, we are not interested in any piece of know-how, we must focus on those know-hows that have some value to our decision making that is targeted towards our business objectives.



We recognize two types of knowledge items in an organization:

1. **Explicit Knowledge:** knowledge that information items on which it is based are already available in system.
2. **Tacit Knowledge:** knowledge which has no representation or that has partial representation in existing processed information items while it does exist in other forms in the organization, beyond the information systems (e.g. undocumented know-how accumulated in employees' minds, informal decisions, training materials, etc.) In case of a tacit knowledge, we naturally do not have in hand a raw knowledge item. Thus, representation of a tacit knowledge consists of the Context-Interrelations-Relevancy metadata only with some sort of a pointer to the source of that knowledge (Expert, Community, etc.)

### III. What is Knowledge Management?

Consequently, knowledge management is defined as a process that manages all aspects of knowledge items, as defined above, all long their lifecycle.

#### *Knowledge Lifecycle Management Tasks*

We categorize all knowledge lifecycle management tasks in several stages:

1. **Creation, Inception and Capturing:** aggregation of knowledge from external as well as internal sources, both through online and batch processes, manually and automatically. In addition, the ability to create new knowledge item information, in the case of tacit knowledge.
2. **Processing:** construction of knowledge metadata: classification, context, interrelations, relevancy through an optional workflow scenario. Keeping knowledge updated as to constant changes and events.
3. **Distribution, Reuse and Sharing:** posting & presentation, syndication, sharing and knowledge-based collaboration.

Moreover and above all, we demand from such a management system to manage all three aspects listed above that make an item a knowledge, i.e. context, relations, relevancy, and to maintain it all along the tasks mentioned above:

1. **Context Management:** maintain categorization, meaningful coded description and understanding of the item. Example for such functions are a taxonomy, indexing and automatic summarization and keyword extraction for free text items.
2. **Interrelations Management:** graph or other network representation of interrelations between the various knowledge items in various contextual aspects and categories.
3. **Relevancy Management:** maintain relation of a knowledge item with the business objectives of the organization. This relation can be represented either by business rules or taxonomies.

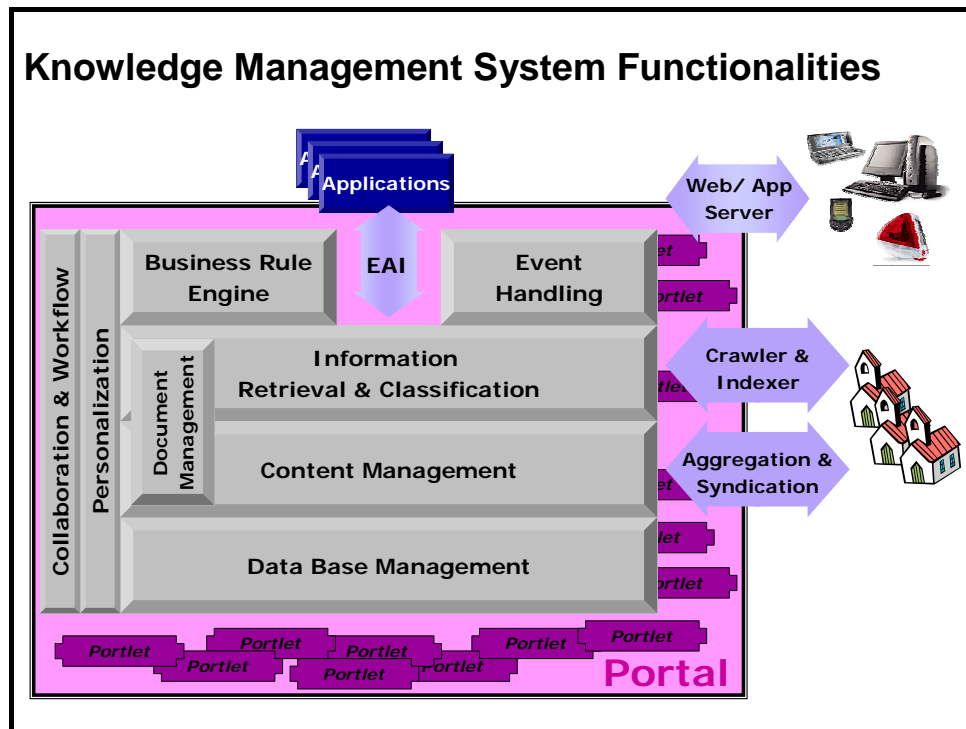
#### IV. Knowledge Management System Types

Knowledge Management systems vary considerably in essence, purpose and scope. Most Knowledge Management systems can be classified into the one or more of the following categories:

- Enterprise Portal
- Knowledge Sharing and Knowledge-based Collaboration
- Document Management
- Best Practices Management
- Competitive Intelligence
- Organizational Knowledge Classification and Retrieval
- Customer Support Management (CSR systems) and Self-Service
- Tacit Knowledge Handling
- Enterprise Expertise Management

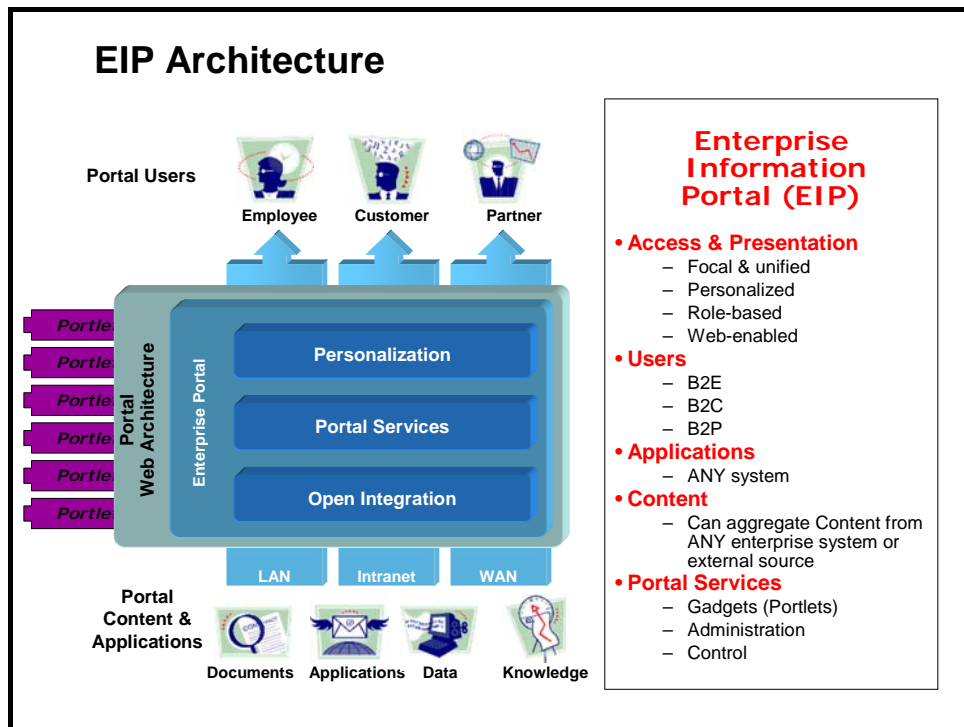
#### VI. Knowledge Management System Architecture

The following diagram depicts the essential functionalities required by a knowledge management system in order to fulfill the above tasks:



These are not necessarily complete commercial products but rather than that functional and technological islands that can sometimes span more than one commercial product. These functional and technological islands should be integrated and sometimes developed as part of the Knowledge Management project.

- Portal:** Many Knowledge Management functionalities are encapsulated nowadays within an Enterprise Information Portal (EIP) framework. Such an EIP framework enhances the knowledge handling process to a more efficient and friendly manner, automating many of the tasks that previously required development and providing ready-made services (Portal Services). Such a framework reduces the need for development to customization-only using ready-made flexible components such as plug-ins and portlets. The EIP functions as an integration center and presentation focal point for all the disparate components involved in the knowledge management process. Such an integration is obtained through OEM products already included within the EIP framework (such as an advanced Search Engine) or through open APIs and interfaces, both generic (e.g. Web Services) and specific (e.g. a tailored interface to a specific commercial product).



- **Information Retrieval & Classification:** This capability is obtained by an advanced search engine technology. Functionalities include indexing, classification, taxonomy, various search options, etc.
- **Content Management System:** since a knowledge item is also content or/and a data, as defined above, Knowledge Management Systems must include, in addition to their knowledge management capabilities, a content management functionality, either integrated or by using an external content management package. These capabilities should include an underlying *information model* for the arrangement of the content in a meaningful adaptive form. Sometimes this functionality is addressed as *Document Management* where a basic content item is actually a document file.
- **Business Rule Engine:** Supports interrelations between knowledge items and their linkage to business objectives.
- **Event Handling:** Event tracing and launching in order to keep knowledge updated as to changes and events.
- **EAI Capabilities:** Some of the information comprising a knowledge item resides in the information systems. In order to extract such information, an EAI capability (e.g. Web Services) is required.

- **Collaboration:** Knowledge-based collaboration and sharing among individuals, communities, work teams and interest groups. Collaboration is especially important for handling tacit knowledge and in general for sharing knowledge by individuals in disparate places at the organization. Recently, **Social Network** approach has been introduced to the market as part of the collaboration tools. This category includes **Expertise profiling** and **expert location** that aim at automatic identification of and connecting to formal and informal experts. Also within collaboration, **e-Learning** functionality is sometimes included to delegate and share knowledge through structured virtual classrooms and other e-Learning facilities.
- **Workflow:** workflow mechanism assures a methodological life cycle handling of knowledge items, including approval routings.
- **Personalization:** to assure the right knowledge item to be delivered to the right person, at the right timing.

## X. Knowledge Management - Commercial Products

Given the above required functionalities and technologies, we should map them into executable components. A typical Knowledge Management System is combined of the following types of executable components and efforts:

- Commercial software products.
- Customization and integration (through parameters, menus, administration interfaces, and APIs)
- Development

Given a system requirements and design, approach should be to favor elements in the order listed above (i.e. ready-made building blocks implemented by commercial software products, then customization and integration, and only when necessary – development).

Knowledge Management popular commercial products include:

- Data Base Management Servers
- Web / Application Servers
- Enterprise Information Portals (EIP)
- Search, Classification & Retrieval Engines
- Document Management Systems
- Competitive Intelligence packages
- Content Management Systems
- Integration Technologies (Web Services / EAI)

- Business Rule Engines
- Collaboration Tools
- Expertise Management Tools (Expertise Profiling, Expert Location)
- Personalization Engines
- Knowledge Visualization Technologies
- CSR Products
- Tacit Knowledge Management Systems

## XI. Technological Issues to Watch

### Technological Issues to Watch

- **Multi-platform & Portability**
- **Compliance with standards**
- **Performance**
- **Future Growth**
  - Scalability
  - From B2E to B2P & B2C
- **Extensibility**
  - APIs
  - Components & Solution Alliances
  - Friendly interfaces & administration
- **Open Security Mechanism**
  - SSO, Roles, Zoning
- **Distributed Architecture & Fault Tolerance**
- **ROI Calculation**
- **Strategic in the Vendor's offerings**

