

# **A Reusable Article on Reusable Components**

by  
*Jim Pellerin, PMP*

## ***Introduction***

Companies waste thousands of dollars, if not millions of dollars, every year on redundant development of various information components such as presentations, courses, marketing material, graphics, etc. These components are usually developed by different people in different departments and end up representing similar or common ideas. To help reduce this waste, an organization should implement a centralized component based development approach.

## **What is Component Based Development?**

Component based development is the development of components that individually perform one discrete or unique function. These components can be reused and combined to form other components that perform another discrete or unique function. The main objective of component based development is to



reduce time and costs related to the development of reusable learning components.

Component based Development has already been introduced as a software development approach which is used to reduce software development costs and to decrease the development time to develop software applications. However, this approach does not have to be limited to the software development field. In fact, the use of the component based development is being highlighted very well with all the work being done with the concepts of *learning objects* and *knowledge objects*.

There has been much discussion and confusion over what the term “object” actually means when discussing *learning*. I have seen the terms *knowledge object*, *learning object*, *instruction object* used interchangeably with no differentiation between them. I prefer to use the term *component* here to define a discrete and reusable entity and to avoid the use of the term *object*. Also, *object*, in the field of Object Oriented programming, has its own meaning with a much different set of characteristics.



## Benefits of a Component Based Development

There are several benefits of Component Based Development:

1. **Reuse** - Probably the most common benefit and the one that is easiest to implement is *reuse*. Significant costs savings, as well as reduced development time, can be realized through a proper reuse strategy. Components built for one project can be used in other components for other purposes.
2. **Central Management** - Components that are centrally located and linked to from several locations can be changed very easily. If a change is made to a centralized component that has been included in several presentations, or courses, that change will be reflected in all the presentations as soon as the change has been made to that one component in one place.
3. **Dynamic Construction** - Learners and organizations can *dynamically* create training modules that are specific to the individual learner. This will create a unique learning experience for the individual learner and will contribute to the overall motivation of the learner. The training will be more specific, and could eliminate unnecessary background information which may have already been learned. Also, training solutions can be created to satisfy the different types of learners.



4. **Course Assembly** – The different types of components will require specific skills to create. For example, Content Components could require subject matter experts and graphic artists to create. Instructional designers are required to create Instructional Components and Learning Components. Once these are created, courses could be assembled, as needed, by non-technical people without any instructional design expertise or learning background.
5. **Standardization** – Components created using this approach would have predefined standards for such things as layout, graphics type, metadata, etc.

There are some non-supporters of Component Based Development and Reusable Learning Objects or components. Some people argue that components designed for reuse lack the subject context alignment. Their position is that using components that were originally designed and developed for one project lack the necessary context required to satisfy a particular learning objective in another project. I believe that if Component Based Development is implemented properly, the educational values are not compromised.



## What is a component?

A component is a self contained entity that can be created, independent of other components, and can be reused when creating other components. To get maximum benefit from a component, a component should perform only one unique function.

Each component should contain a certain amount of descriptive information that allows it to be accessed easily so it can be used for each specific need. This descriptive data is commonly known as *metadata*, or data about data. It is essentially the attributes that are used to best describe and categorize the component for maximum access and reuse. For example, a Content Component may have metadata which describes: Name, Description, Category, Media Type (Graphic, Text, Video, or Audio), Size, Keywords, Format (gif, txt, doc, etc.), resolution, orientation, caption, etc. Using this metadata information helps the developers to use this component when creating other components.

Components can be combined and reused to support more efficient and effective learning. The more independent and unique a component is, the more reuse there can be.



## Reuse Approach

Organizations don't realize it but they can be much more efficient in the development and distribution of information and learning material. I have seen many organizations struggle with the redundant creation of duplicate information. (e.g. an organization chart is created by one department to be included in a presentation while another department creates the same or chart for an employee orientation course)

The concept of component based development is the development of components that are **discrete** and **reusable** which can be combined into other *components* which are also **discrete** and **reusable**.

By **discrete** I mean that it satisfies one objective or performs one specific function. For example, a simple component could be a graphic of "a picture of car". The objective of this component would be "to show what a car looks like". A text component could be created that describes the "characteristics of a car". Another component could be a graphic of "a picture of a truck". The objective of this component would be "to show what a truck looks like". A text component could be created that describes the "characteristics of a truck". The graphic component and the text components of the car can be combined to create a new



component which “provides information about a car”. Similarly, the graphic component and the text components can be combined to create a new component which “provides information about a truck”. These two “information” components could also be combined and reused to form another component which “describes information about vehicles”. And so on.

Reusable components achieve maximum benefits when implemented at the enterprise level in their electronic form. Benefits can also be found when including reusable components in manuals, books, or other fixed presentation forms.

## **Level of Implementation**

To receive the most benefits, Component Based Development should be embraced and implemented for an entire enterprise. That being said, it does have some benefits when implemented by an individual developer.

At the enterprise level, a company should setup libraries, databases, development standards, publishing standards, templates, etc. The organization should have a clear strategy on how to develop components, how to store and manage these



components and how to link to and reuse the components. These standards would be used when creating and accessing the learning components. This would allow other developers in an organization to share learning components with other developers on other projects.

If there is no enterprise approach to component based development, the individual developer could still benefit by setting up their own libraries, databases, development standards, publishing standards, etc. This would allow the developer to reuse learning object on other projects.

Some organizations benefit by purchasing “components” from other organizations. One popular practice is the purchase of clip art for presentations or lessons. Articles can be found on the internet which can be used to support your course. There are also learning component factories which provide learning components that can be included as part of your course development.

## ***Component Types***


There are many components than can exist at many levels within the learning domain. I have classified these components as Content Components, Knowledge





Components, Instruction Components, Learning Components and Learning Products. Each of these categories of components describe a level of component that performs a specific function that can be combined together to form another type of component.

## 1. Content Component

 A content component is the simplest form of information (e.g. text, video, graphic, audio).

The objective of a Content Component is to represent information in some form such as text, video, audio, etc. These components have the greatest opportunity for reuse. Reuse can be accommodated by placing them in libraries that are centrally located and globally accessible. Examples of these libraries are the clip art library that comes with your word processor or presentation software. These libraries usually have hundreds of pictures, audio and videos. The reuse of these libraries is the simplest when it is only available to one person, the developer. An alternative approach would be to setup a corporate repository where all employees could store their own created content files. These files should be stored on common storage such as a shared network drive. The files should be indexed and

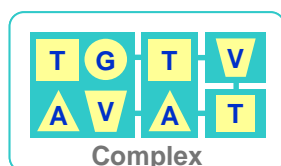


documented so that they can easily be located. Examples of public and commercial media libraries are: [www.clipart.com](http://www.clipart.com) or <http://images.google.com>

*Reuse* is very high for Content Components. Content components are used during the development and construction of other types of components. The quickest and easiest way for an enterprise to begin implementing these components is to setup standards and strategies for maintaining and cataloging Content Components. The commercially available content libraries should also be indexed and referenced and integrated with those created by the organization. These components can be equated to the Asset files of the SCORM model.

*Metadata:* Content Components could use the following metadata attributes: Name, Description, Category (Business, Education, etc.), Content Type (Graphic, Text, Video, Audio), Size, Keywords, Format (gif, txt, doc, etc.), resolution, orientation, caption, etc.

## 2. Knowledge Component



A knowledge component is a combination of one or more Content Components. The



objective of a Knowledge Component is to represent or describe some basic forms of knowledge or information such as a Fact, a Concept, a Procedure, a Structure, a Process or a Principle. Facts and Concepts are Simple Knowledge Components. A Fact represents one piece of knowledge. A Concept (type) classifies knowledge of the same type. Simple Knowledge Components are also components of Complex Knowledge Components. Procedures (steps), Structures (parts), and Processes (how things work) and Principles (analysis) are Complex Knowledge Components.

For example, a Fact Component, which describes a guitar's machine head, could be combined with another Fact Component describing the guitar's nut, to create a Structure Component describing the guitar's headstock. The headstock Structure Component could then be combined with a fretboard Fact Component, and a tuner Fact Component to create the guitar's neck Structure Component. The neck component and body component could be combined to create a guitar Structure Component which describes the parts of the guitar.

Reuse can be high for Knowledge Components. They are used during the development and construction of Instruction Components and Learning Components. Knowledge Components can also be used on their own, independently of Instructional Components and Learning Components, when the



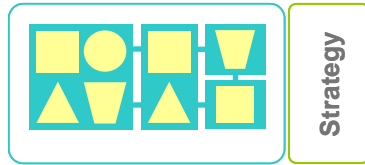
only objective is to inform and not promote learning. It could be argued that the activity of informing or providing information could benefit by incorporating instructional design strategies and, therefore, should always be enhanced through the creation of Instruction and Learning Components. The recipient of the information would be better engaged so they actually learn the information by applying instructional strategies to the information.

An example of a Knowledge Component would be a web page used to provide information about a classical guitar. It could contain a picture of a guitar, the text describing the guitar and an audio clip to demonstrate what the guitar sounds like. The picture, text and audio clip were all Content Components previously created and reused in this Knowledge Component.

*Metadata:* Knowledge Components could use the following metadata attributes: Name, Description, Category (Business, Education, etc.), Knowledge Type (Fact, Concept, Procedure, Process, Principle, or Structure), Size, Keywords, Format (html, doc, ppt, etc.), etc.

### **3. Instructional Component**





An Instructional Component is used to achieve one instructional objective. An

Instructional Component includes an instructional strategy that accomplishes one instructional objective such as Tell, Show, Do and Ask. These instructional objectives (Tell, Show, Do and Ask) were first introduced by Dr. David Merrill in his instructional design work. Several Instructional Components are combined together to form one Learning Component.



“Tell” describes the information or rule of the topic to be learned. It is the basic building block of any Learning Component. It is the topic of

information and describes what it is that is being taught.

“Show” provides examples or non examples of the same topic of information just described. There could be none, one or more “Show” instructional components depending on the type of knowledge and the complexity of the information being presented.

“Do” provides practice for the learner. This practice must be aligned with the Learning Objective of the Learning Component. For example, if we are teaching



classes of automobiles, then we would provide examples of the various classes of automobiles. There could be zero, one, or more “Do” instructional components depending on the complexity of the information being presented.

“Ask” provides an assessment activity which checks the learners newly acquired knowledge. Multiple “Ask” instructional components may be used to ensure adequate assessment has occurred. “Ask” components can be used as pre-tests and post-tests.

These Instructional Components can be combined into Learning Components. Each Learning Component is used to satisfy one learning goal. Let’s suppose we want to teach the difference between a classical, acoustic and electric guitars. First we would create three Instructional Components by combining a Knowledge Component with additional information to describe the various characteristics of each guitar (*Tell*). We could create a web page for each piece of information, that is, each type of guitar. The *objective* of these Instructional Components is to describe each type of guitar and the characteristics of each type of guitar. This may also include prerequisite information if this is the first time a learner is learning about a guitar (the guitar parts described earlier). We would also create three more Instructional Components to “*Show*” examples of each type of guitar, highlighting the characteristics which determine their uniqueness. Characteristics

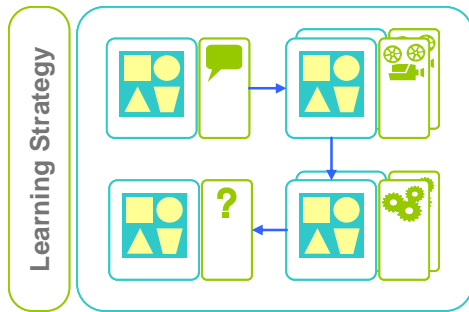


such as size, body type, strings, etc. should be used to ensure the uniqueness of each *type* of guitar. This could be a web page for each of the guitar types Next we create another Instructional Component to practice (“*Do*”) differentiating between the three different types of guitars. Ideally, we use Knowledge Components that were not used in the previous “*Show*” activities as examples. This could be a web page for each example. And finally, we create a Instructional Component to assess what has been taught. All of he Instructional Components can be combined to create a Learning Component with the objective to teach the differences between a classical, acoustic and electric guitar. It should be easy to see that, because Instructional Components are componentized properly, additional guitar types, such as bass, jazz or flamenco guitars could be added very easily.

*Metadata:* Instructional Components could use the following metadata attributes: Name, Description, Category (Business, Education, etc.), Instruction Strategy (Tell, Show, Do, or Ask), Size, Keywords, Format (html, doc, ppt, etc.), etc.

#### **4. Learning Components**





A Learning Component is used to achieve one Learning Objective. It can contain several Instruction Components.

Learning Components are created by combining several Instructional Components with the appropriate Learning Strategies. Learning Strategies exist in many forms. Examples of learning strategies include Sequencing, Job Tasks, Motivations, Objectives, Introductions, Summaries, Pre-Tests, etc.

Let's look at our example to teach the differences of guitars. We have already acquired or developed the different Content Components that portray the different types of guitars. We then combined these Content Components to create the Knowledge Components that describe a guitar (Fact) and the different parts (Structure) of the guitar and the different types (Concept) of guitars. These Knowledge Components were combined with Instructional Strategies to create Instructional Components that describes the guitar (Tell); components that show examples of the different types of guitars (Show); components that provide yet more examples for practice (Do); and finally components that assess the learning achieved (Ask). These Instruction components can be combined with Learning





Strategies to form a Learning Component with an objective to “describe the difference between guitars”.

The following examples of Learning Strategies could be included:

1. Test for the pre-requisite level of knowledge (pre-test). If pre-requisite knowledge is not present a Learning Component that teaches “the parts of the guitar” is provided.
2. Sequence the different Instructional Components appropriately. Use a Tell → Show → Do → Ask sequence. Allow the selection of additional Show, Do Instructional Components, if required. Allow review of information (Tell) if required. The flow could be sequential as a default sequence but could also be dynamic based on learner behavior and learner responses.
3. Motivation should be provided that describes why this Learning Component is appropriate for a specific job task and what the benefits to the learner are.
4. Objectives should describe to the learner what they should expect to achieve at the end of the Learning Component. For example, “at the end of this lesson, the learner will be able to distinguish between a classical and electric guitar”.
5. Introductions should be provided that will introduce the learner to what the focus of this Learning Component is.



6. A Summary should be provided which summarizes the Learning component.

A Learning Component is probably the closest to what the Learning Industry is referring to as Learning Objects and also to the SCORM Content Package. There are learning object factories that specialize in creating reusable learning objects for sale. Examples are:

Merlot - <http://www.merlot.org>

SLOPE – <http://slope.senecac.on.ca>

AEShareNet - <http://www.aesharenet.com.au/>

When Learning Components are componentized properly they can be combined to create Learning Products with higher level objectives. Reuse of Learning Components can be highest during development phases. In its simplest form, a single Learning Component can also be a Learning Product (e.g. a module). Learning Component could be reused during the delivery of the instruction when the learner combines various Learning Components to design their own Learning Product.



*Metadata:* Learning Components could use the following metadata attributes: Name, Description, Category (Business, Education, etc.), Learning Objectives, Size, Keywords, etc.

## **5. Learning Products**

A learning product is simply the packaging and delivery of learning. A learning product could be a module, a lesson, a course or a curriculum. There is some degree of reusability at this level. For example a course on guitars could include a lesson on the Parts of a guitar, History of the Guitar and the types of Guitars. Another course on the history of music might have a lesson on the History of the Piano, History of the Guitar, History of Drums, etc. The History of the Guitar is the same lesson but is reused in each course.

Reuse can be present at the time of development where a module can be part of one or more lessons, a lesson can be part of one of more courses. Learning Products can also be reused by learners when creating their own learning experience and dynamically selecting the Learning Products they wish to use. This allows instruction to be individually tailored to meet the needs of each learner based on their learning styles, interest and goals.



The following is an example of how the different types of components can be combined to form a Learning Product.

Course (Learning Product)

Course Introduction (Learning Strategy)

Course Pre-Test (*Reusable Instructional Component*)

Module 1 (*Reusable Learning Product*)

Module Introduction (Learning Strategy)

Module Pre-Test (*Reusable Instructional Component*)

Lesson 1 (*Reusable Learning Component*)

Lesson 2 (*Reusable Learning Component*)

Module Test (*Reusable Instructional Component*)

Module Summary (Learning Strategy)

Module 2 (*Reusable Learning Product*)

.....

Course Test (*Reusable Instructional Component*)

Course Summary (Learning Strategy)

---

Reusable Learning Component LCx



Introduction (Learning Strategy)

Pre-Test (*Reusable Instruction Component ICy*) **Ask**

Rule (*Reusable Instruction Component ICx*) **Tell**

Example 1 (*Reusable Instruction Component ICz*) **Show**

Practice 1 (*Reusable Instruction Component ICw*) **Do**

Test (*Reusable Instruction Component ICv*) **Ask**

Summary (Learning Strategy)

---

Instruction Component ICx – Tell Guitar Information

Tell Information – (*Reusable Knowledge Component KCx*)

Tell Information – (*Reusable Knowledge Component KCy*)

---

Knowledge Component KCx – Provide Guitar Information

Text – Description of Guitar (*Reusable Content Component CC1*)

Graphic – Picture of Guitar (*Reusable Content Component CC2*)

The Learning Product component is the equivalent of creating a SCORM Package which includes all the run time environment controls which is used to direct the



Learning Management Systems (LMS) for such things as assessment and sequencing.

## ***Uses of Learning Components***

Instruction is a structured approach to improving a student's performance in a specific subject. Instruction does not have to occur in a classroom environment or in a course? Instruction can occur when a sales person is trying to educate a potential customer on the benefits of a product. The desired result is that the customer has enough knowledge to make an informed decision to purchase the product.

Another form of instruction could be a presentation to a company's employees, educating the employees of the company's financial situation. The desired outcome is to motivate the employee to perform relative to the financial situation (cut expenses, invest in new products, etc.). Another form of instruction could be an article or book which could change the behavior of the reader (e.g. diet book to help the reader lose weight).

While there are opportunities to implement good instructional design strategies, the creators of these products seldom have the knowledge or the desire to create



learning products as part of the presentation. A component based development approach would allow organizations to implement instructionally rich products that would extend outside of courses and could be used in presentations, sales material and much more. People without any instructional design experience could assemble the already created components which contain the instructional design strategies and learning strategies embedded, thus producing a much richer presentation.

There are many commercially available tools which could be used in your Component Based Development approach. Some of these are Netg, Authorware, Toolbook, etc. Learning components can be managed and delivered using SCORM compliant Learning Management Systems (LMS).

### ***IMS / SCORM / XML***

Components and component based development can be implemented through the use of a SCORM based development and delivery approach, using SCORM development tools and a SCORM compliant LMS to develop and deliver Shareable Content Objects (SCO). The model presented in this article represents specific components that are more granular but the concepts are very similar and



SCORM can be adopted very easily. SCORM uses Assets, SCOs, and Content Organization. This article also recommends a particular Instructional Design strategy while SCORM is instructional design neutral and refers only to “activities”.

The Shareable Content Object Reference Model (SCORM) is a common Web-based content creation and content delivery approach for learning content.

SCORM is a collection of specifications and standards adapted from multiple sources to provide a comprehensive suite of e-learning capabilities that enable interoperability, accessibility and reusability of Web-based learning content.

The main purpose of SCORM is to provide standards for the production, packaging and delivery of reusable, sharable learning objects. SCORM targets the web as the delivery medium. It uses the IMS content packaging standard to define how learning objects should be created, packaged and reused to maximize interoperability so they can be delivered on different delivery engines (Learning Management Systems). SCORM also incorporates specific standards for the assessment and sequencing of the learning content. It does not dictate any particular instructional design strategy.

SCORM incorporates various standards such as IEEE LOM (Learning Object Metadata), AICC Content Structure and the IMS Content Packaging and





Sequencing standards. XML is used to describe the instructions that are to be used by the LMS.

## ***Summary***

Component based development is used to create components which can be used in various learning activities and information sharing activities. The main benefit of component based development is time and cost savings.

Components must be developed with reuse in mind. Start building your component library now with enterprise specific components and by integrating commercially available components.

## **Author Contact**

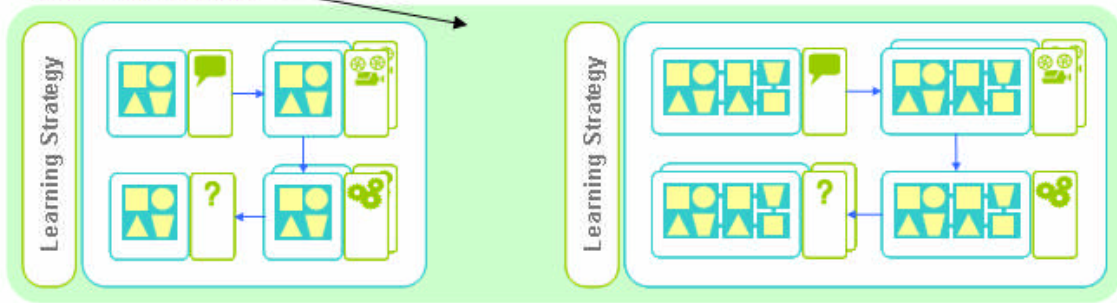
Jim Pellerin has over 25 years experience as a Project Manager, IT Consultant and Learning consultant. He is president of the Ivory Owl Learning Company. His main interest is in instructional design and component based development for learning products. Reach Jim by email at [pellerin@ivoryowl.com](mailto:pellerin@ivoryowl.com)





# COMPONENT BASED LEARNING

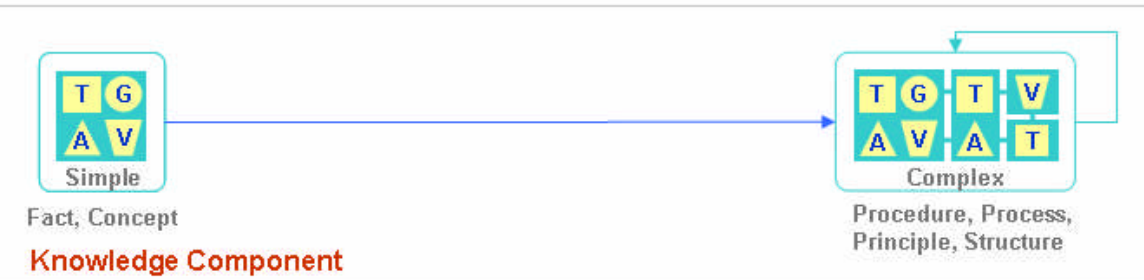
Learning Product



Learning Component



Instruction Component



Knowledge Component

Content Component

